Comparative Study of Robustness in Drowsiness Detection by Adversarial Attacks and Adversarial Retraining

Moin Khan
New York University
mk8793@nyu.edu

Ishan Miglani
New York University
im2410@nyu.edu

Priyanshi Singh
New York University
ps4609@nyu.edu

Siddharth Singh
New York University
ss16915@nyu.edu

Abstract

This project presents a comparative study on the robustness of drowsiness detection models against adversarial attacks. Building upon the work of Gwak et al.[1], our team introduced a custom CNN model, demonstrating superior resilience compared to a base model (Mobilenet v2). We employed various adversarial attacks, including FGSM, PGD, Auto-PGD, DDN, Deepfool, Patch, Carlini Wagner, and Boundary followed by an adversarial retraining process.[9] The results highlight the enhanced safety, accuracy, and resilience achieved through our approach.

GitHub repository: https://github.com/singh-priyanshi/ML-for-Cybersec-Final-Project

1 Introduction

In the realm of autonomous driving and machine learning for cybersecurity, the detection of driver drowsiness stands out as a critical component ensuring both safety and reliability. The foundation of our research is built upon the work of Gwak et al., where their ensemble machine learning algorithms laid the groundwork for drowsiness detection. However, their approach faced limitations in terms of robustness, especially under adversarial conditions.

The pressing needs in the current landscape include the inadequacies of existing drowsiness detection systems, which are prone to errors under varying conditions. The safety risks associated with drowsiness while driving or operating machinery underscore the urgency of enhancing the reliability of these systems. Moreover, the vulnerability of current models to adversarial attacks adds an additional layer of concern, pointing towards a significant research gap in the resilience of these algorithms against manipulations.

1.1 Motivation

The motivation for our research is rooted in envisioning a future where traditional drowsiness detection systems may falter under manipulations, potentially leading to safety hazards in autonomous driving scenarios. The advent of adversarial attacks, subtle manipulations designed to mislead machine learning models, poses a new challenge that demands attention. Picture a scenario where a malicious actor can deceive a drowsiness detection system, leading to misclassification of a vigilant driver as fatigued, thus compromising the safety of the entire system.

Our research aims to bridge this gap by presenting a novel approach: a custom-built Convolutional Neural Network (CNN) model designed specifically for drowsiness detection. Unlike the ensemble methods used by Gwak et al.[1], our custom model is introduced with the intention of providing superior robustness. Through a comprehensive comparative study involving various adversarial

attacks, we seek to demonstrate that our custom model exhibits enhanced resilience and reliability compared to the base model.

1.2 Objectives

1. Model Robustness: Develop a custom CNN model with a focus on robustness against adversarial attacks, contrasting with the limitations observed in existing models.

2. Adversarial Attacks: Implement a variety of adversarial attack techniques, including Fast Gradient Sign Method (FGSM), Projected Gradient Descent (PGD), Auto-Projected Gradient Descent (Auto-PGD), Decoupling Direction and Norm (DDN), Deepfool, Patch, Carlini Wagner, and Boundary attacks, to thoroughly test the vulnerability of both the custom CNN and a widely-used alternative, Mobilenet v2.

3. Retraining Process: Enhance the models using data obtained from the adversarial attacks to fortify their resilience against future manipulations.

4. Evaluation Criteria: Assess the models based on accuracy, reliability, and resistance to adversarial attacks, providing a comprehensive understanding of their performance.

1.3 Significance

The significance of our research lies in its potential to contribute to safer autonomous driving and improve the dependability of drowsiness detection systems. By addressing the current limitations and vulnerabilities, our work aims to pave the way for more secure and robust machine learning applications in critical domains. Moreover, the insights gained from our comparative study could extend to broader applications in ML security, influencing the development of more resilient models across various sectors.

In the subsequent sections, we will delve into our approach, the various adversarial attack techniques employed, the benefits of our research, the competitive landscape, and the contributions of each team member, culminating in a comprehensive presentation of our findings and the broader impact of our work in the field of machine learning for cybersecurity.

2 Literature Survey

The study by Gwak, Hirao, and Shino focuses on early detection of driver drowsiness, a significant factor in traffic accidents. [1] It explores the classification of driver alertness, especially the slightly drowsy state, using a combination of vehicle-based, behavioral, and physiological indicators. The research utilized a driving simulator and driver monitoring system to measure various factors, including physiological signals from EEG and ECG. Machine learning algorithms were applied to identify driver states and construct a dataset. The findings indicated that an ensemble algorithm achieved 82.4% accuracy in classifying alert and slightly drowsy states and 95.4% for alert and moderately drowsy states. When physiological indicators were excluded, the random forest algorithm achieved 78.7% accuracy for alert vs. slightly drowsy states and 89.8% for alert vs. moderately drowsy states. This research underlines the potential for implementing a driver drowsiness detection system using non-contact sensors. [1]

The research by Mehta et al.[2] introduces a real-time driver drowsiness detection system leveraging the Eye Aspect Ratio (EAR) and Eye Closure Ratio (ECR). This system, designed as an Android application, utilizes a video-based approach to continuously monitor the driver's facial features. By analyzing facial landmarks, particularly around the eyes, the system computes EAR and ECR for drowsiness detection. The use of machine learning, specifically the Random Forest classifier, demonstrated an 84% accuracy in identifying drowsy states. This study highlights the feasibility of employing smartphone-based, non-intrusive methods for enhancing road safety through drowsiness detection.[2]

Our research enhances driver drowsiness detection by combining the insights of Gwak, Hirao, Shino, and Mehta et al. with a focus on robustness. Acknowledging the strengths of previous studies, we introduce a Custom CNN model, surpassing Mobilenet v2 in resilience against adversarial

attacks. This approach not only improves detection accuracy but also ensures reliability under diverse adversarial conditions, setting a new standard in the field.

3 Dataset

In this study, we leveraged the Closed Eyes in the Wild (CEW) dataset[4], a comprehensive collection comprising 1452 subjects. Within this dataset, 726 individuals exhibit closed eyes, while 726 individuals have open eyes. The CEW dataset[4] provides a diverse set of scenarios, enriching our understanding of drowsiness detection. Our focus on the CEW dataset[4] ensures a robust exploration of drowsiness detection across varied subjects and conditions. Figure 1 illustrates examples from the CEW dataset.[4]

Figure 1: Sample Images from CEW DataSet [4]

4 Methodology

4.1 Models

Mobilenet v2, used as a baseline for comparison, has 3,360,322 parameters. Despite its widespread application, this model shows certain weaknesses when faced with adversarial attacks. This comparison underlines the necessity of developing more resilient neural network models. The study presumably delves into the specific aspects where Mobilenet v2 falls short and demonstrates how the custom CNN model overcomes these limitations, providing a detailed analysis of both models' performance and resistance to adversarial attacks.

The custom CNN model, featuring 5,561,922 parameters, is meticulously crafted to mitigate vulnerabilities observed in the Mobilenet v2 model when subjected to adversarial attacks. This focal point of our research is tailored for heightened resilience in the face of such threats. Leveraging advanced attributes or structures, the model is fortified against adversarial concerns, integrating convolutional and max pooling layers, followed by fully connected layers. The input shape is dictated by the 4D tensor format of the training data, while subsequent layers strategically diminish filter numbers and feature map dimensions. This CNN, with a softmax activation output layer, promises enhanced security, and its training involves categorical crossentropy loss, accuracy metrics, and the Adam optimizer, making it adaptable for image classification datasets.

4.2 Adversarial Attack Techniques

In the cybersecurity domain, machine learning faces unique challenges due to adversarial attack techniques. These attacks involve deliberate manipulation of input data to exploit vulnerabilities in learning algorithms, leading to incorrect model outputs. In cybersecurity, these techniques are particularly concerning as they can bypass security systems designed to detect malicious activities. The field thus focuses on strengthening machine learning models against such attacks, ensuring they remain effective and reliable in identifying and mitigating cyber threats. This ongoing effort is crucial for maintaining robust cyber defenses in an increasingly digital world.[9]

Model Specification	Values
Number of Convolutional Layers	5
Number of Filters in Convolutional Layers	512, 512, 256, 256, 256
Kernel Size in Convolutional Layers	(3, 3)
Number of Max Pooling Layers	5
Pooling Size in Max Pooling Layers	(2, 2)
Number of Dense Layers	3
Number of Units in Dense Layers	128, 64, 2
Loss Function	Categorical Crossentropy (SoftMax)
Optimizer	Adam
Learning Rate	0.001 (default of Adam)
Number of Parameters	4,873,924

Table 1: Custom CNN Model Configuration Details

1. Fast Gradient Sign Method Attack :
FGSM is a technique in machine learning for cybersecurity, designed to test the resilience of neural networks. By applying small but intentional disturbances to input data, FGSM effectively highlights vulnerabilities in a model's ability to process and categorize information accurately. This method is pivotal in cybersecurity as it allows researchers and developers to anticipate potential weak points in ML defenses, ensuring that machine learning models can withstand and adapt to sophisticated cyber attacks. FGSM is a cornerstone in the ongoing effort to enhance the security and reliability of ML systems in cyber defense.

The Fast Gradient Sign Method (FGSM) operates using the formula:

$$Perturbed\ Input = Original\ Input + \epsilon.sign(\nabla_x J(\theta, x, y))$$

In this formula, ϵ represents a small value determining the magnitude of the perturbation, J is the loss function of the model, θ denotes model parameters, x is the input to the model, y is the target output, and ∇_x signifies the gradient of the loss concerning the input. FGSM thus creates perturbations that maximize the loss, leading to model misclassification.

2. Projected Gradient Descent Attack :
PGD is a refined adversarial attack strategy in machine learning, particularly relevant in cybersecurity. Unlike simpler methods, PGD iteratively adjusts its perturbations, staying within a defined boundary to ensure subtlety. This makes it a more sophisticated and realistic test for machine learning models, simulating advanced cyber threats. By repeatedly applying these small, calculated adjustments, PGD exposes potential weaknesses in ML systems, helping to fortify them against complex, real-world cyber attacks. Its iterative nature makes it one of the most effective techniques for evaluating model robustness.

Projected Gradient Descent (PGD) operates using the iterative formula:

$$Input_{t+1} = Clip_{Input+\epsilon}(Input_t + \alpha \cdot sign(\nabla_x J(\theta, Input_t, y)))$$

Here, Input$_t$ is the input in iteration t, ϵ defines the maximum perturbation magnitude, α is the step size, J is the loss function, θ represents the model parameters, y is the target output, and Clip ensures that the perturbed input stays within the allowable range around the original input. This process iteratively refines the attack for maximum effectiveness.

3. Auto-Projected Gradient Descent Attack :
Auto-PGD in machine learning for cybersecurity is an advanced variant of the PGD attack, characterized by its automated tuning of hyperparameters. This automation allows for a more efficient and effective discovery of vulnerabilities in ML systems. In cybersecurity, Auto-PGD represents a significant step forward in simulating sophisticated cyber threats, testing the resilience of machine learning models against intricate and adaptive adversarial attacks. Its capability to self-adjust makes it a formidable tool for uncovering potential security breaches in ML defenses.

The basic working formula for Auto-Projected Gradient Descent (Auto-PGD) is similar to PGD but includes an additional component for automatic hyperparameter optimization. This typically involves

adjusting parameters like the step size and the perturbation limits based on the performance feedback of each iteration, optimizing the effectiveness of the attack. The exact mathematical representation of Auto-PGD can vary depending on the specific implementation and the adaptive algorithm used for tuning these parameters.

4. Decoupling Direction and Norm Attack :
DDN in machine learning for cybersecurity involves separating the direction of the gradient from its magnitude when crafting adversarial attacks. This method enhances the sophistication of attacks by fine-tuning their effectiveness, allowing for more precise exploration of vulnerabilities in ML systems. It's particularly important in cybersecurity, as it contributes to developing more robust defense mechanisms by understanding how different aspects of adversarial perturbations impact model performance and security.

It's a conceptual approach where the direction of the gradient (indicating the direction of maximum increase of the loss function) is decoupled from its magnitude (or norm). This allows for more nuanced manipulations in adversarial attacks. The exact implementation can vary, but it typically involves separately adjusting the gradient direction and its magnitude to find the most effective perturbations.

5. Deepfool Attack :
DeepFool is an adversarial attack technique in machine learning, particularly relevant in cybersecurity. It is designed to efficiently compute minimal perturbations that cause a machine learning model to misclassify input data. The strength of DeepFool lies in its ability to find the shortest path to cross model decision boundaries, providing insights into the model's vulnerabilities. This method helps cybersecurity professionals understand and improve the robustness of ML systems against subtle and efficient adversarial manipulations.

DeepFool operates by iteratively perturbing an input image to cross the decision boundary of a classifier. Its basic working principle can be summarized as:

$$Perturbed\ Input = Original\ Input + \sum \delta$$

where δ represents small perturbations calculated in each iteration to move the input closer to the classifier's decision boundary. The goal is to find the minimal change required to alter the model's output, providing a measure of the model's robustness.

6. Patch Attack :
The Patch Attack in machine learning, crucial in cybersecurity, involves embedding a specially crafted patch into an image to deceive neural networks. Unlike other adversarial attacks that modify the entire image, Patch Attack strategically places a visible but inconspicuous patch to cause misclassification. This technique challenges the robustness of ML models by demonstrating how easily they can be fooled with minimal and localized alterations, highlighting the need for advanced detection and defense mechanisms in cybersecurity applications [8].

It involves creating and superimposing a patch onto an image. The patch is designed to maximize the error in the neural network's prediction. The effectiveness of the patch is typically evaluated by how well it can induce misclassification when added to various images. This technique showcases the need for robustness in neural networks against localized, visible perturbations.

7. Carlini Wagner Attack :
The Carlini-Wagner attack in machine learning is a sophisticated adversarial technique crucial in cybersecurity. It effectively creates subtle yet potent alterations in data to mislead ML models. This method highlights the need for more resilient defenses in ML systems.

The core of the Carlini-Wagner attack is an optimization process aiming to find the smallest possible perturbation that leads to misclassification. It is represented by the formula:

$$minimize \|\delta\|_2 + c \cdot f(x + \delta)$$

Here, δ is the perturbation, $\|\delta\|_2$ is its L2 norm, f is the model's output function, and c is a balancing constant. This formula aims for minimal yet effective changes to the input data, ensuring the attack's efficacy and stealth.

8. Boundary Attack :
In the field of cybersecurity, the Boundary Attack is a technique in machine learning that incrementally

alters input data to test the resilience of models. This method is effective in uncovering vulnerabilities, as it makes minimal, hard-to-detect changes to inputs, thereby challenging the robustness of ML systems.

The Boundary Attack operates by starting with a sample from the target class and iteratively adjusting it towards a source instance, ensuring the model's classification remains unchanged. This iterative process aims to find the least amount of change needed to cross the model's decision boundary, highlighting the model's sensitivity to input variations.

5 Result

In our study, we systematically assessed the robustness of Mobilenet v2 and Custom CNN drowsiness detection models against various adversarial attacks, including FGSM, PGD, Auto-PGD, DDN, DeepFool, C&W, and Boundary Attack. Initial accuracies exhibited variability, as Mobilenet v2 dropped to 51.4% post-FGSM, while Custom CNN maintained 96.2%. Through retraining, both models demonstrated substantial improvement, with Mobilenet v2 reaching 99.7% and Custom CNN achieving 94.2% accuracy after PGD. Auto-PGD enhanced Mobilenet v2 to 1.0 and Custom CNN to 0.97. DDN and DeepFool attacks showcased increased post-retraining accuracies for both models. Similarly, C&W attacks on adversarial data resulted in improvements, with Custom CNN showing an increase from 0.89 before retraining to 0.98 after retraining, and Mobilenet v2 improving from 0.51 to 0.99. Additionally, Boundary Attack on adversarial data led to notable enhancements, with Custom CNN improving from 0.05 before retraining to 0.90 after retraining, and Mobilenet v2 increasing from 0.07 to 1.0. Throughout these experiments, the retraining strategy, utilizing 25% adversarial and 100% clean training data, consistently proved effective in enhancing the robustness of both models against diverse adversarial attacks.

Table 2: Parameter settings of Mobilenet V2 and Custom CNN

Parameters	**Mobilenet V2**	**Custom CNN**
Epsilon	0.01	0.01
Batch Size	32	32
Retraining Data	25% Adv + 100% Clean Training	25% Adv + 100% Clean Training

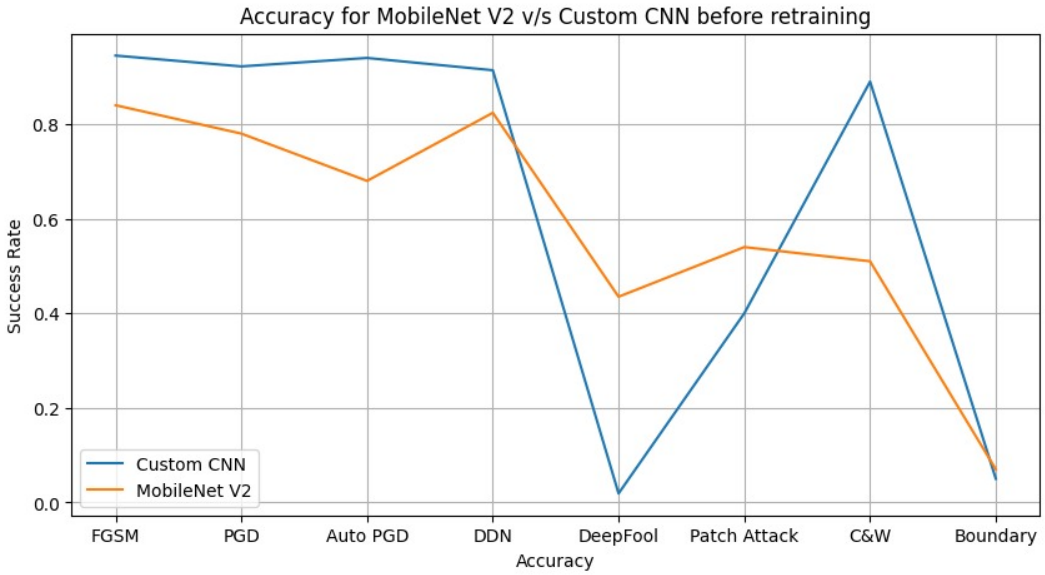

Figure 2: Accuracy comparison of models before re-training on adversarial data

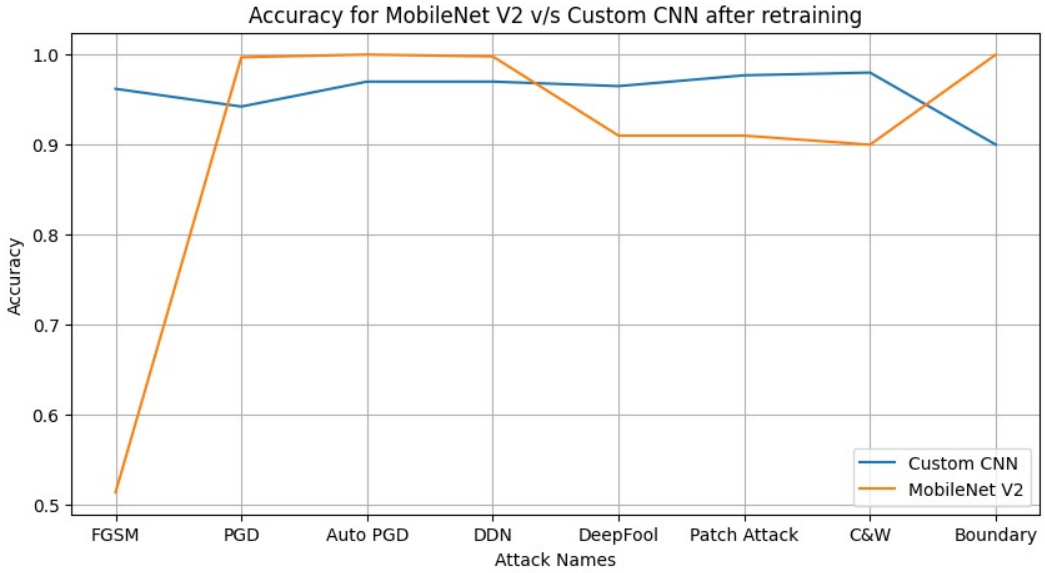

Figure 3: Accuracy comparison of models after re-training on adversarial data

Attack	Mobilenet V2 (Before)	Mobilenet V2 (After)	Custom CNN (Before)	Custom CNN (After)
FGSM	0.84	0.514	0.945	0.962
PGD	0.78	0.997	0.9221	0.9423
Auto PGD	0.68	1.0	0.94	0.97
DDN	0.824	0.9979	0.914	0.97
DeepFool	0.435	0.91	0.019	0.965
Patch Attack	0.54	0.91	0.4	0.977
C&W Attack	0.51	0.99	0.89	0.98
Boundary Attack	0.07	1.0	0.05	0.90

Table 3: Accuracy Before and After Retraining for Mobilenet V2 and Custom CNN.

6 Limitations

While the presentation adeptly outlines the problem statement and proposes an innovative solution, it is crucial to acknowledge and discuss the limitations inherent in the study. The paper highlights the enhanced robustness of the custom CNN model through empirical evaluations, particularly in the context of adversarial attacks. However, it's essential to recognize the following limitations:

- **Contextualizing Data Distribution:** In our study, the data distribution was impacted by two key decisions: the division into training and testing sets and the integration of a 25% adversarial training set with the original training dataset. This latter approach was driven by computational constraints; under different circumstances, a larger adversarial set could potentially yield better model performance. However, this strategy might not fully reflect the diverse data characteristics found in real-world scenarios, which could influence the general applicability and effectiveness of our findings.
- **Addressing Robustness Claims:** Our claim of enhanced robustness was based on the model's performance under the conditions defined by our chosen benchmark. However, this claim must be critically examined in the context of data distribution assumptions. The robustness of AI models is often challenged when faced with data significantly different from the training set. For instance, our model, trained on specific image types, may not maintain the same level of robustness when exposed to varied or more complex images, such as full-face pictures in contrast to only closed or open eyes. This highlights a need for further validation and testing across a broader spectrum of data types.

- **Real-world Transferability:** The proposed model's effectiveness in dynamic real-world scenarios is not validated, and its response to variations in practical adversarial attacks needs exploration.
- **Drowsiness State Generalization:** The study focuses on adversarial attacks, but the model's generalization to diverse drowsiness states across individuals and contexts should be scrutinized.
- **Evaluation Metrics and Benchmarks:** While relevant metrics are presented, additional metrics and comparisons with state-of-the-art models would provide a more comprehensive assessment.
- **Impact of Retraining:** The impact of retraining on the model's performance, potential overfitting concerns, and computational resources required for frequent retraining should be considered.

7 Conclusion

In conclusion, our study on drowsiness detection's robustness through adversarial attacks and retraining yielded profound insights. Building upon Gwak et al.'s work, we improved ensemble algorithms using a custom CNN, addressing existing system limitations. Initiating with needs identification, safety risks, and vulnerabilities assessment, we meticulously compared our Custom CNN (5,561,922 parameters) with Mobilenet v2. Employing FGSM, PGD, Auto-PGD, DDN, Deepfool, Patch, Carlini Wagner, and Boundary Attack, our Custom CNN demonstrated superior post-retraining accuracy, enhancing safety and resilience against adversarial attacks. Beyond accuracy, our approach contributes to ML security insights, applicable in diverse sectors. We are grateful for our instructors' and TAs' unwavering support, our research ensures impactful advancements in robust drowsiness detection.

References

[1] J. Gwak, A. Hirao, and M. Shino, "An investigation of early detection of driver drowsiness using ensemble machine learning based on hybrid sensing," *Applied Sciences*, 10(8), 2890. Retrieved from https://www.mdpi.com/2076-3417/10/8/2890

[2] S. Mehta, S. Dadhich, S. Gumber, and A.J. Bhatt, "Real-time driver drowsiness detection system using eye aspect ratio and eye closure ratio," *SSRN Electronic Journal*. Retrieved from https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3356401

[3] S. Sathasivam, A. K. Mahamad, S. Saon, A. Sidek, M. M. Som, and H. A. Ameen, "Drowsiness detection system using eye aspect ratio technique," In *2020 IEEE Student Conference on Research and Development (SCOReD)*. Retrieved from https://doi.org/10.1109/scored50371.2020.9251035

[4] X. Tan, Y. Lu, J. Yang, and L. Jin, "Closed eye in the wild (cew) database and benchmarking," *arXiv preprint arXiv:1705.05679*. Retrieved from http://parnec.nuaa.edu.cn/_upload/tpl/02/db/731/template731/pages/xtan/ClosedEyeDatabases.html

[5] M. Ozdag, "Adversarial Attacks and Defenses Against Deep Neural Networks: A Survey." Retrieved from https://www.sciencedirect.com/science/article/pii/S1877050918319884

[6] Driver drowsiness detection. (2023). In *Wikipedia, The Free Encyclopedia*. Retrieved from https://en.wikipedia.org/wiki/Driver_drowsiness_detection

[7] Brightspace Course Content. (2023). *Machine Learning for Cybersecurity*. Retrieved from https://brightspace.nyu.edu/d2l/home/316313

[8] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1712.04248. Retrieved from https://arxiv.org/abs/1712.04248

[9] Adversarial Robustness Toolbox. (n.d.). Retrieved from https://adversarial-robustness-toolbox.readthedocs.io/en/latest/