OS-CS311 Project Phase 3

Literature Review

Author	Problems Addressed	Methodology	Key Contributions
Wu et al. (2018)	Multiclass DDoS traffic classification	CNN based IDS with flow features	CNN model showed high performance with fewer parameters
DeepDefense (2019)	Combining CNNs and RNNs for DDoS detection	Traffic feature arrays within time windows	High accuracy but very resource intensive
Ghanbari & Kinsner (2020)	Improving CNN sensitivity in DDoS detection	Feature Extraction using wavelet transformation	87.35% Accuracy - lacks real time performance metrics
Zhang et al. (2021)	Low-rate DDoS attack detection	Hybrid decision tree and deep learning model	Use of historical traffic data integration which enhanced pattern detection
Shahid et al. (2022)	Real-time DDoS detection in IoT networks	Federated learning models	Applicable on IoT systems with scalability, low latency and less False Positives
IEEE Study (2023)	Efficient DDoS detection in constrained systems	Lightweight CNN for resource constrained systems	High accuracy with minimal compilations overhead

Result Analysis

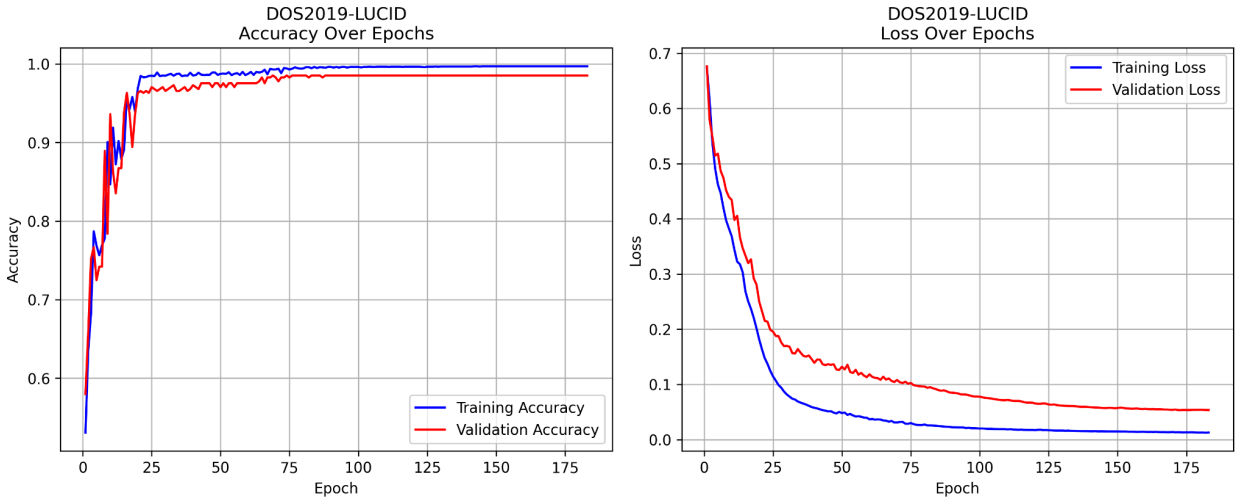
The paper used grid-search-cv to loop across multiple hyperparameters to find the best fitting one. The hyper-parametres that were used in training were:

- Learning rate
- Batch Size
- Kernel Size
- Regularization
- Dropout

The best model chosen after training had the following hyperparameters:

- Learning Rate: **0.1**
- Batch Size: **2048**
- Kernel Size: **64**
- Regularization: **None**
- Dropout: **None**

The accuracy and losses after multiple epochs mapped the following graphs

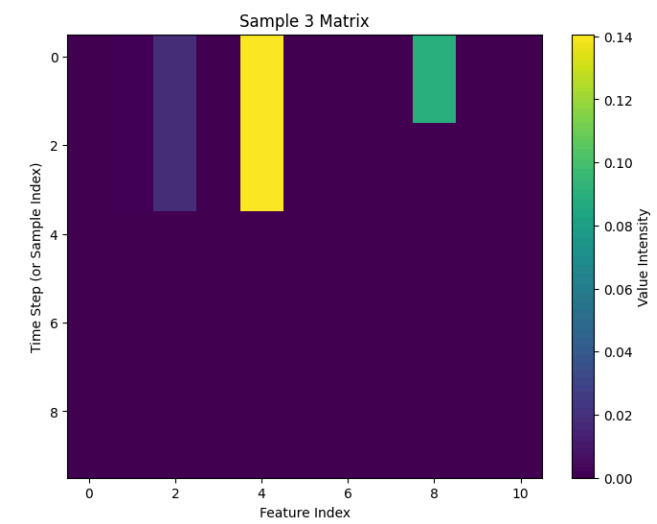


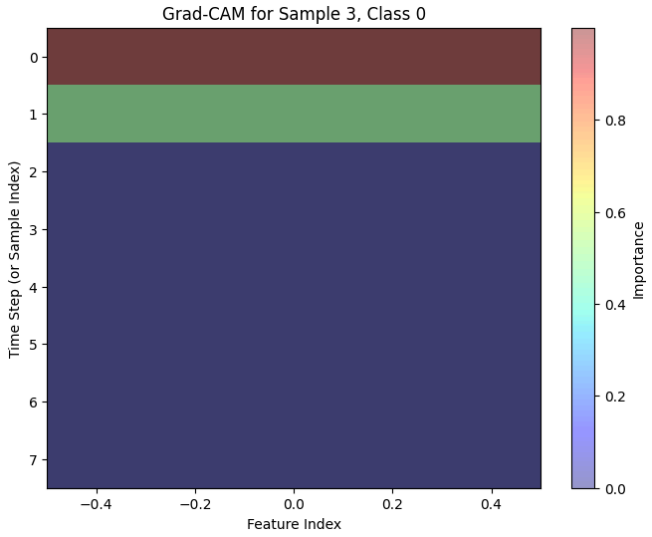
The graphs show an excellent result when it comes to the bias-variance tradeoff. This is because the training accuracy is **98.5%** showing a good bias while our validation accuracy is not far off with an accuracy between **96 - 97%.** With minimal differences between both training and validation accuracies and losses, this shows a good variance of the model.

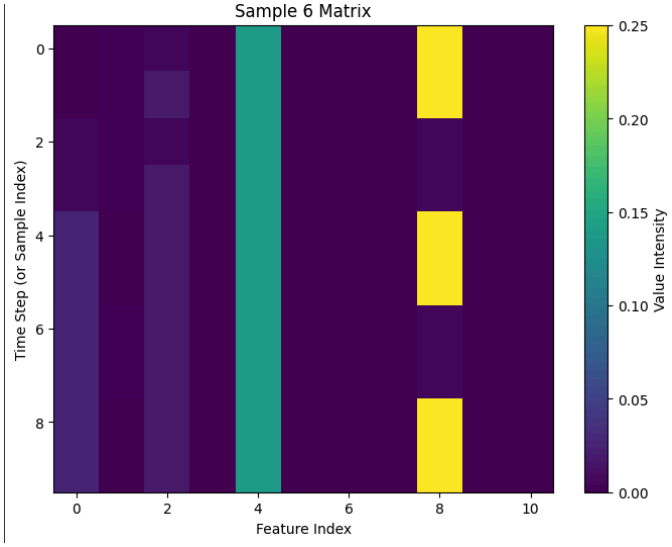
The F1 score of our model comes out to be **0.982.**

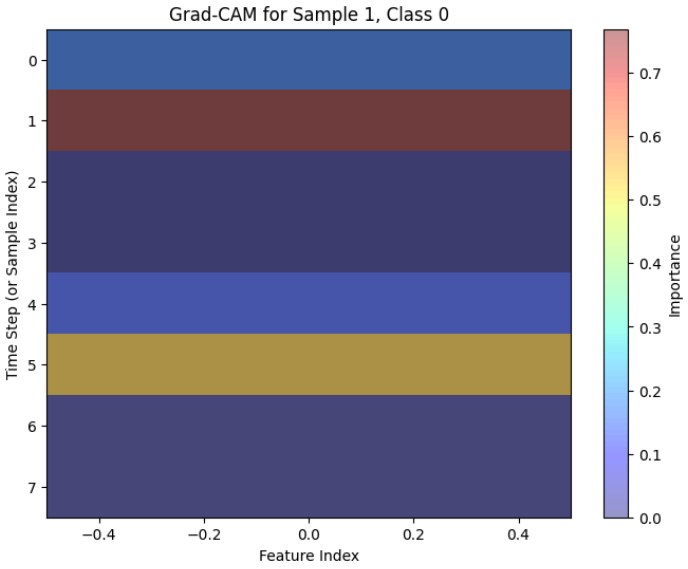
Our novelty deals with overcoming the lack of explainability of our CNN. CNN being a black-box, lacks the simplicity for others to understand how LUCID actually detects DDoS attacks. For LUCID to be reliable enough to tackle such a security threat we have used visualization maps to illustrate how the CNN functions in understanding important features.

We use Gradient-Weighted Class Activation Maps (G-CAM) to show how CNN emphasises certain features over others. G-CAM uses gradients to determine this importance. By introducing this into our codebase we gained the following results.









Future Work

Multi-class DDoS Detection

Future work can focus on extending the binary classification to multi-class models capable of identifying specific DDoS attack types. By developing a clearer classification of different attack types, we can improve the model's applicability to diverse attack scenarios.

Domain Adaptation and Transfer Learning

To make our system more adaptable across different networks, domain adaptation techniques can be explored. This involve using unsupervised or semi-supervised learning strategies to better align our model with new datasets. Transfer Learning technique can improve performance for large scale datasets.

Real-time Performance Optimization

While our current project allows for predictions on live traffic, there remains room for improvement. Research can be done on lightweight CNN architectures like MobileNet or explore model quantization and pruning techniques which are especially useful in resource-constrained environments.