|  |
| --- |
|  |
| MATH9001 – Research Methods |
| **Comparative Analysis of Machine Learning and Neural Networks in Intrusion Detection Systems to Detect Different Types of Cyber Attacks on Simulated Batch Streaming Data** |
|  |
| **Brian Higgins – R00239570** |
|  |

|  |
| --- |
|  |

|  |
| --- |
| I hereby certify that this material which I now submit for assessment is entirely my own work and has not been taken from the work of others, save to the extent, that such work has been cited and acknowledged within the text of the work.  I understand that my project documentation may be stored in the library at MTU, and may be referenced by others in the future. |

Contents

[1 Abstract 2](#_Toc134635308)

[2 Chapter 1 – Introduction 3](#_Toc134635309)

[2.1 Background and Motivation 3](#_Toc134635310)

[2.2 Research Question 4](#_Toc134635311)

[2.3 Objective and Scope 4](#_Toc134635312)

[3 Chapter 2 – Literature Review 5](#_Toc134635313)

[3.1 Intrusion Detection Systems 5](#_Toc134635314)

[3.2 Machine Learning Techniques with IDS Systems 5](#_Toc134635315)

[3.3 Neural Networks with IDS Systems 5](#_Toc134635316)

[3.4 CSE-CIC-IDS20218 dataset 6](#_Toc134635317)

[3.5 Imbalanced Data 6](#_Toc134635318)

[4 Chapter 3 – Proposed Dataset and Methodology 7](#_Toc134635319)

[4.1 Dataset 7](#_Toc134635320)

[4.2 Methodology 8](#_Toc134635321)

[4.3 Risks and Challenges 9](#_Toc134635322)

[5 Chapter 4 – Weekly Research Plan 10](#_Toc134635323)

[5.1 Grant Project Plan 10](#_Toc134635324)

[5 References 12](#_Toc134635325)

# 1 Abstract

Over the last number of years, Cyberattacks have become increasingly sophisticated which is posing an increased risk to organisations. Intrusion Detection Systems (IDS) need to improve to play a curial role in preventing and safeguarding computer systems. Using Artificial Intelligence (AI) methods such as Machine Learning (ML) and Neural Networks (NN) IDS systems can be trained to recognise attacks in real time. This research aims to conduct a comparative analysis of ML and NN techniques for detecting and classifying different types of network attacks using a dataset from the Canadian Institute of Cyber Security. To simulate a real-world data flow, Apache Spark will be used to stream the data in batches and using Apache’s Machine Learning Module, MLib, which has specially adapted ML and NN models to analyse the data as it is streamed into the models. Using this approach allows the researchers to investigate the effectiveness of different ML and NN models which will help to contribute to the development of more reliable and effective intrusion detection systems. The results of this research will provide valuable information on which of the models are best suited to enhance network security and allow faster detection of different types of cybersecurity attacks in real-time.

*Clinical Relevance* — This research will offer to enhance Intrusion Detection Systems by comparing the performance of various ML and NN models in detecting and classifying network attacks in real-time.

*Keywords* —Intrusion Detection Systems, Artificial Intelligence, Machine Learning, Neural Networks, Random Forest, Gradient Boosting, K-Means, Convolutional Neural Networks, Long Short-Term Memory

(252 words)

# Chapter 1 – Introduction

## Background and Motivation

Cybersecurity attacks have been increasing over the last number of years [1] and this has included two high-profile attacks on the HSE in 2021 [2] and Munster Technology University in 2023 [3] emphasizing the urgent need for effective IDS systems to be able to identify and prevent these attacks. These attacks, while damaging for an organisation’s reputation [4] can also be detrimental to a person’s privacy [5] by releasing unauthorised personal information such as medical records, student records, financial records or private correspondences online.

Artificial Intelligence (AI) techniques such as Machine Learning (ML) and Neural Networks (NN) have shown huge promise in detecting cybersecurity attacks but still have many challenges [6]. ML and NN can learn from historical data to identify patterns and irregular patterns in network traffic to enable the detection of various attacks such as Man-in-the-Middle, Denial-of-Service, Malware Detection and Distributed Denial-of-Service attacks for example. Previous research has focused on performing modelling on full datasets as a single event in order to detect what type of attack is happening. This approach does not accurately reflect a real-world situation where systems need to continuously monitor network traffic and logs to detect if an attack is happening and what kind of attack as different countermeasures are needed for different types of attacks. The HSE hackers were reportedly in the computer system for 8 weeks [7] before they were found. So this approach of having all the data upfront is unrealistic. IDS systems need to monitor data in real-time and then raise the alarm when they notice an attack. This research is a novel approach from past research as it aims to classify attacks in real-time and evaluate the performance of different ML and NN models. As there are different types of attacks it will also look to see if different ML and NN models are better at detecting each type of threat.

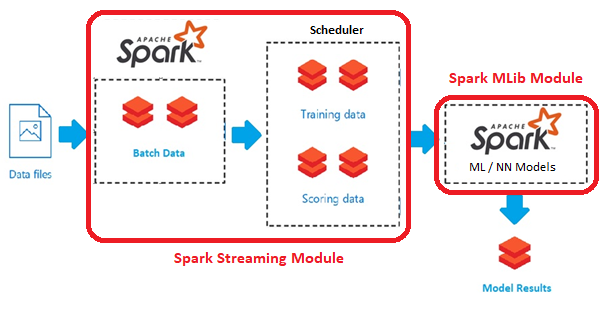


Figure 2.1: Project Architecture

Figure 2.1 shows an overview of the project architecture and how Apache Spark will be used. The research will use two of Sparks Modules; Spark’s Streaming module [8] and Spark’s Machine Learning module, MLib, [9] . The Spark Streaming Module will be used to load in the data from the Canadian Institute of Cybersecurity and then batch the data. A scheduler will then feed in the batched data to Spark’s MLib module which has a number of ML and NN models that will be used to classify the different types of attacks. The reason to use Apache’s system, is because it speeds up the set up of the research. A new system could be programmed with pure python but then some of the ML and NN models need to be adjusted to work with batched streaming data. Using Apache Spark quickens the set up of the project so the research can focus on comparing different ML and NN models.

## Research Question

The research question for this project is: Compare ML and NN models in Apache’s Machine Learning Module, MLib, to classify several cybersecurity attacks when the dataset is streamed in batches. Which of these models performs best at classifying the various types of attacks in order to improve IDS systems.

## Objective and Scope

The primary objective of this research is to compare the effectiveness of ML and NN models when used in the Apache Spark ecosystem for classification of different types of cyber security attacks in a simulated real-time environment. The focus is to distinguishing between 6 different types of attacks from the dataset from the Canadian Institute of Cybersecurity (CIS) which closely mirrors a real-world network environment with multiple attacks.

This research will be using Apache Sparks Streaming module in order to segment the dataset into batches and then stream this into different ML and NN models. Previous research has treated the dataset as one singular entity, but this is unrealistic as models need to learn as data is streamed in, simulating a real-world environment. This approach is where the new novel approach comes from as it is unknow which of the models will perform best or how this streaming approach will affect the results from the different models. This approach is extremely important as IDS systems need to be able to learn in real-time. Models will need to train as data is streamed in and so the length of time of training becomes more important and can also have an effect on the performance of a model. A model that performs better but falls behind on training will cause a backup on the scheduler and will be less effective overall.

Another objective is that each attack will have different characteristics and so distinct countermeasures are needed to react to the cybersecurity threat. Different models may provide different results for different types of attacks so the use of different models is essential not just for the overall results of the classification but to investigate which models perform best for each type of attack.

(822 words)

# 3 Chapter 2 – Literature Review

In this section there is an overview of existing literature in the areas relevant to this research, focusing on Intrusion Detection Systems, Machine Learning, Neural Networks, the Canadian Institute of Cybersecurity dataset, CSE-CIC-IDS20218 and the imbalanced nature of the data.

## 3.1 Intrusion Detection Systems

IDS systems [12] are a critical component of an organisation’s security system as they can be critical in helping to identify and prevent unauthorised access but their success has not been proven [13]. IDS systems can be classified into two types, Host-Based and Network-Based [14]. Host-based IDS systems focus on monitoring activities on a single machine and Network-Based looks at analysing network traffic across a network. This can include a single machines data In the form of Windows logs (copy, paste, transfer of data instances) and network logs (traffic upload, download, IP addresses and ports). This research is looking at a Network-Based approach. Machine Learning and Neural Networks can be used to improve the performance of IDS systems and this research will explore which models work best for the different types of attacks on batch streaming data using network traffic data.

## 3.2 Machine Learning Techniques with IDS Systems

Machine Learning models have shown promise in helping to improve the detection rate in IDS systems [15]. Different ML models have been applied to help identify irregular patterns or in detecting anomalous behaviour in networks such as Random Forest [16], Gradient Boosting [17], Support Vector Machines [18] and K-nearest Neighbour [19]. The models are also designed to work with large datasets and reduce the amount of false positives, i.e. situations where the models detect an attack but there is none. They used some large security datasets such as KDD99 [20] and NSL-KDD [21]. This research is taking a different novel approach in that it will not be training on the full data as one block but instead is looking at simulating a real-world environment by batch streaming data into ML and NN models and comparing which models perform best. This is a unique improvement on the previous research.

## 3.3 Neural Networks with IDS Systems

Neural Networks, a subset of AI, have also been used in Intrusion Detection Systems. Convolutional Neural Networks (CNN) [22], Long Short Term Memory (LSTM) networks [23] and Autoencoders [24] are some of the various NN used with IDS systems. These studies have looked at some advanced AI models which offer alternative approaches to ML models to address the challenges presented when looking at the large volumes of data, the level of imbalance and other challenges of network traffic data. These studies also offer different approaches with some methods such as Principal Component Analysis (PCA) [25] and Mutual Information (MI) [26] which have been used to reduce the large amount of dimensionality of network traffic. These studies have used these techniques to improve accuracy as network traffic can have hundreds of features.

Using NNs on streaming data does offer additional challenges due to the high amount of incoming data, lack of meaningful labels, computational complexity and training times. For this research the data is loaded in batches so there is a limited amount of data to start with. This is a novel approach for this research as it will be interesting to see how the NN’s will perform given there need to train with large amounts of data and this researches batch nature which at first means there will be limited data available. The ML models may perform better than the NN models, but it will be interesting to find out.

## CSE-CIC-IDS20218 dataset

Previous research on the CSE-CIC-IDS2018 [27] dataset has explored many ML models such as Random Forest, XG Boast, Support Vector Machines (SVM), and K-Nearest Neighbour (KNN) [28], [29] and had very good results. Different NN models such as Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM) and Autoencoders have also been used [30], [31], [32]. This project will also look at performing some of the above models but looking to classify each type of attack into one of six types when the data is batch streamed in . This streaming approach means some of the previous models are not designed for this approach or setting them up would be beyond the scope of this research. Apache Spark was chosen because it allows this streaming approach to be set up relatively quickly. This approach was discussed with the Distributed Data Management lecturer in order to make sure it is feasible.

Previous research had very high results with some previous studies having accuracy scores between 96% [30] and 100% [33]. Some papers also had recall scores of 100% [33], [34] and precision scores of 100% [33], [34]. This level performance could indicate some overfitting perhaps due to the imbalanced nature of the data, this is something that needs to be considered when this project is performed, something that was skipped in the mentioned research papers as they did not attempt to adapt for overfitting.

## Imbalanced Data

One of the biggest challenges with an IDS system is that some organisations may not be attacked for a very long time while other organisations such as banking, finance and e-commerce would have higher instances of cybersecurity attacks. Either way data imbalance [35] is very common in IDS systems. Various techniques have been developed to deal with the imbalance nature of this data with resampling methods [36] such as Random Undersampling (RU), Random Oversampling (RO) and Synthetic Minority Oversampling Technique (RU-SMOTE). Later when starting the modelling these techniques will be bore in mind.

(918 words)

# 4 Chapter 3 – Dataset and Methodology

## 4.1 Dataset

The dataset used in this project is CSE-CIC-IDS20218 from the Canadian Institute for Cyber Security at the University of New Brunswick [27]. It is hard to find real-world datasets of network traffic and so this dataset was designed to simulate normal traffic in a network environment which then simulates 6 different types of cybersecurity attacks. These malicious types of attacks include Distributed Denial of Service (DDoS), Denial of Service (DoS), Brute Force, Botnet, Infiltration and Web attacks. These attacks were designed to replicate real-world cyber-attacks.

This dataset consists of several days of network traffic with over 16 million data points and over 80 features so offers a rich and diverse collection of traffic for research. The data is labelled with each data point having either a benign label or one of several classes of attacks. The data is made up of different departments with a combination of single machines, servers and switches, Figure 4.1 shows the network architecture of the organisation. The data points include features such as source and destination IP addresses, port numbers, protocols, packet length and others. Table 4.2 shows the distribution of the data and the percentage of data for each class of attack. The data is massively imbalanced which is normal for data of network traffic and cyber-attacks.

Diagram

Description automatically generated

Figure 4.1: CSE-CIC-IDS2018 Diagram

|  |  |
| --- | --- |
| **Traffic Type** | **Distribution** |
| Benign | 83.1% |
| DDoS | 7.7% |
| DoS | 4.0% |
| Brute Force | 2.4% |
| Botnet | 1.8% |
| Infiltration | 0.997% |
| Web Attack | 0.006% |

Table 4.2: Traffic Type Distribution

## Methodology

This research uses Apache Spark’s MLib module, which works with large-scale data and supports batch and streaming data so that the research will be able to simulate a streaming situation by batching the data into smaller chunks and feeding this into the MLib ML and NN models. The research further breaks down the methodology:

1. **Data Pre-processing**

Before using the machine learning models, the data needs will to be pre-processed to fix missing data, change object types and create new features for example. Since this is a simulated streaming scenario this does not need to be done live on the streaming data so this will reduce the complexity of running the models.

1. **Batch Streaming Setup**

Once Pre-processing has been done, data will be divided into batches to simulate real-time data flows and a scheduler will be used to stream the data into the models.

1. **Model Selection and Implementation**

Using the literature review, models were chosen that had previously performed well on the full dataset. The performance of these models when applied to streaming data remains to be determined.

* **Random Forest:** Ensemble learning method that creates a multitude of Decision Trees (DT) which has had good performance from previous research [16].
* **Streaming K-means:** A type of unsupervised ML model that has had some good results in looking for anomalies in network data in previous research. [37].
* **Gradient Boosting Trees:** Another ensembling method of DT’s that has also worked well in previous research [17].
* **Convolutional Neural Network:** A deep learning model which takes an input and sets the importance by iterating through the network changing weights. CNNs have also had good results in the previous research [31].
* **Long Short-Term Memory:** A type of NN that can learn and remember over sequences of data. This model is likely not to do well at first but could offer some of the best results over time as seen in previous research [23].

1. **Hypertuning of models**

Examine the top performing models and tune hyperparameters to explore potential improvements.

1. **Performance Evaluation**

As this is a real-world scenario, analyse the results in real-time for each type of attack. Compare models in real-time and compare the different models to each other.

1. **Analysis and Interpretation of results**

Analysis the performance of different models, both ML and NN’s to notice any strength or weakness. Investigate different metrics to offer a complete comparison.

1. **Discussion and Conclusion**

Discuss the implications of the findings from the point of improving IDS systems. Which models performed the best? Did certain models do better for certain types of attacks? Were there issues with the time needed to train models for ML V’s NN? Highlight any limitations of the research.

## Risks and Challenges

1. **Setting up the Apache batch steaming system:** A core concept of this research and its novel approach is that it will look at the data in a streaming format which has not been looked at in previous research for this dataset. This does offer challenges but it is an important aspect of this project. Apache Spark was chosen due to its built-in Streaming and ML modules that will speed up the setup process which offers a significant advantage over starting from scratch.
2. **Apache Machine Learning Models:** Not all models are designed to work with streaming data, some of the previous research was completed on full datasets and the same models that were used will not work well when data is streamed in with batches.
3. **Computational Complexity:** 
   1. **Data Size:** Processing large amounts of data can be computationally expensive, especially for complex models. An advantage of using Apache’s services is that you get one cluster for free use and for later models, other clusters can be purchased to speed up computational processing.
   2. **Feeding in batch data:** The timing of schedular in feeding in the simulated batch data is important to ensure that models have enough time to train. This needs to be explored later when model training starts.
4. **Imbalanced Data:** As seen in Table 4.2 and in previous research on this dataset the data is highly imbalanced. This is a realistic real-world aspect of network data and the research will look at techniques to address this issue.
5. **Data Quality:** Pre-processing data is essential to maintain data quality and model performance. This research has the advantage of exploring the data beforehand. A real-world dataset would not have this advantage.
6. **Overfitting:** As mentioned very high accuracy scores have been found in previous models so for this research this needs to be bear in mind. Model Evaluation techniques such as cross-validation or regularization might need to be considered.

# Chapter 4 – Weekly Research Plan

## Grant Project Plan

Outlined below in Figure 5.1 is a comprehensive plan detailing the steps to take for each section. An objective of each section is to produce an initial draft, which should help in the process of compiling the final report.

Chart, treemap chart

Description automatically generated

Figure 5.1: Grant Project Plan

1. **Examination of Literature Review and Methodology**

* Review Previous Research on the Dataset, Cybersecurity, Apache Spark and Apache Machine Learning Models
* Identify gaps in existing research and areas of potential improvements
  + Use the Research Paper checklist created from STAT9004 – Statistical Data Analysis with Catherine Palmer as a guide when reviewing research papers
* Choose the Machine Learning and Neural Networks models to be used for the research
* Investigate the Apache Machine Learning Module and Streaming Module

1. **Writing of Literature Review and Methodology**

* Written Report First Draft: Summarise the findings of the Literature Review
* Written Report First Draft: Define the research problem and objectives clearly
* Written Report First Draft: Describe the chosen methodology to be used

1. **Gathering of Primary Data**

* Completed. Using a dataset from the Canadian Institute of Cybersecurity (CIS)
* Data Pre-processing. Clean data, handle missing data, perform feature engineering. The data is in good condition already as the CIS has already performed data pre-processing on it
* Written Report Frist Draft: Describe data and pre-processing steps taken

1. **Analysis Data/ Apache Spark Setup**

* Perform Exploratory Data Analysis (EDA) on the data
* Set up Apache spark to allow the Data to be fed into models in batches – Discussed with lecturer Ignacio Castineiras
* Configure Apache Spark to include single and multi-cluster (free and paid) set-up and batch streaming of the data
* Written Report First Draft: Write up and explain Apache Spark Streaming Set Up
* Written Report First Draft: EDA description of data

1. **Machine Learning / Neural Network Modelling**

* Explore Apache Sparks’ Machine Learning Module developed to work with batch data
* Implement and train the chosen ML and NN models using Apache Spark
* Evaluate model performance with appropriate metrics
* Perform hyperparameter tunning on best performing models
* Write Report First Draft: Explain Apache Spark’s MLib Module
* Write Report First Draft: Write up results from models and compare their results

1. **Review and rewrite the literature**

* Update literature review with any new findings from throughout the project

1. **Finish write up**

* Write up each section using first drafts
* Additional:
  + Discuss the implications of the findings for the field of IDS and Cybersecurity
  + Make recommendations for future research and areas of improvements
* Review the report using the STAT9004 checklist
* Finalise the Final Report

# 5 References

Reference style is IEEE.

“[1] “The Latest Cyber Crime Statistics (updated May 2023) | AAG IT Support.” https://aag-it.com/the-latest-cyber-crime-statistics/ (accessed May 09, 2023).

[2] “How the HSE cyber attack changed the face of online crime globally,” *The Irish Times*. https://www.irishtimes.com/technology/data-security/2023/03/11/how-the-hse-cyber-attack-changed-the-face-of-online-crime-globally/ (accessed Mar. 17, 2023).

[3] “Hackers threaten to publish ‘confidential’ MTU data unless ransom is paid, High Court told,” *The Irish Times*. https://www.irishtimes.com/ireland/education/2023/02/11/hackers-threaten-to-publish-confidential-mtu-data-unless-ransom-is-paid-high-court-told/ (accessed Mar. 17, 2023).

[4] “How reputational damage from a data breach affects consumer perception | Imprivata.” https://www.imprivata.com/blog/reputation-risks-how-cyberattacks-affect-consumer-perception (accessed May 09, 2023).

[5] S. Romanosky, R. Telang, and A. Acquisti, “Do Data Breach Disclosure Laws Reduce Identity Theft?,” *J. Policy Anal. Manage.*, vol. 30, no. 2, pp. 256–286, 2011.

[6] M. Pawlicki, R. Kozik, and M. Choraś, “A survey on neural networks for (cyber-) security and (cyber-) security of neural networks,” *Neurocomputing*, vol. 500, pp. 1075–1087, Aug. 2022, doi: 10.1016/j.neucom.2022.06.002.

[7] T. McNally, “HSE hackers were in health service’s computer system for eight weeks before cyber attack,” *TheJournal.ie*, Dec. 10, 2021. https://www.thejournal.ie/hse-hack-report-5626054-Dec2021/ (accessed May 09, 2023).

[8] “pyspark.streaming module — PySpark 2.1.1 documentation.” https://spark.apache.org/docs/2.1.1/api/python/pyspark.streaming.html (accessed May 11, 2023).

[9] “MLlib: Main Guide - Spark 3.4.0 Documentation.” https://spark.apache.org/docs/latest/ml-guide.html (accessed May 11, 2023).

[10] “Applications | Research | Canadian Institute for Cybersecurity | UNB.” https://www.unb.ca/cic/research/applications.html (accessed May 11, 2023).

[11] J. L. Leevy and T. M. Khoshgoftaar, “A survey and analysis of intrusion detection models based on CSE-CIC-IDS2018 Big Data,” *J. Big Data*, vol. 7, no. 1, p. 104, Nov. 2020, doi: 10.1186/s40537-020-00382-x.

[12] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, “Survey of intrusion detection systems: techniques, datasets and challenges,” *Cybersecurity*, vol. 2, no. 1, p. 20, Jul. 2019, doi: 10.1186/s42400-019-0038-7.

[13] A. C. Solutions, “Why Next Generation IDS Systems are Flawed.” https://blog.ariacybersecurity.com/blog/why-next-generation-ids-systems-are-flawed (accessed May 09, 2023).

[14] Amrit Pal Singh, “Analysis of Host-Based and Network-Based Intrusion Detection System”.

[15] S. M. Othman, F. M. Ba-Alwi, N. T. Alsohybe, and A. Y. Al-Hashida, “Intrusion detection model using machine learning algorithm on Big Data environment,” *J. Big Data*, vol. 5, no. 1, p. 34, Sep. 2018, doi: 10.1186/s40537-018-0145-4.

[16] N. Farnaaz and M. A. Jabbar, “Random Forest Modeling for Network Intrusion Detection System,” *Procedia Comput. Sci.*, vol. 89, pp. 213–217, Jan. 2016, doi: 10.1016/j.procs.2016.06.047.

[17] P. Verma, S. Anwar, S. Khan, and S. B. Mane, “Network Intrusion Detection Using Clustering and Gradient Boosting,” in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Jul. 2018, pp. 1–7. doi: 10.1109/ICCCNT.2018.8494186.

[18] D. S. Kim and J. S. Park, “Network-Based Intrusion Detection with Support Vector Machines,” in *Information Networking*, H.-K. Kahng, Ed., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2003, pp. 747–756. doi: 10.1007/978-3-540-45235-5\_73.

[19] Y. Liao and V. R. Vemuri, “Use of K-Nearest Neighbor classifier for intrusion detection11An earlier version of this paper is to appear in the Proceedings of the 11th USENIX Security Symposium, San Francisco, CA, August 2002,” *Comput. Secur.*, vol. 21, no. 5, pp. 439–448, Oct. 2002, doi: 10.1016/S0167-4048(02)00514-X.

[20] “UCI Machine Learning Repository: KDD Cup 1999 Data Data Set.” https://archive.ics.uci.edu/ml/datasets/kdd+cup+1999+data (accessed May 10, 2023).

[21] “NSL-KDD | Datasets | Research | Canadian Institute for Cybersecurity | UNB.” https://www.unb.ca/cic/datasets/nsl.html (accessed May 10, 2023).

[22] “(PDF) A Survey of CNN-Based Network Intrusion Detection,” *ResearchGate*, doi: 10.3390/app12168162.

[23] F. Laghrissi, S. Douzi, K. Douzi, and B. Hssina, “Intrusion detection systems using long short-term memory (LSTM),” *J. Big Data*, vol. 8, no. 1, p. 65, May 2021, doi: 10.1186/s40537-021-00448-4.

[24] A. Singh and J. Jang-Jaccard, “Autoencoder-based Unsupervised Intrusion Detection using Multi-Scale Convolutional Recurrent Networks.” arXiv, Apr. 07, 2022. Accessed: May 09, 2023. [Online]. Available: http://arxiv.org/abs/2204.03779

[25] “Intrusion Detection Using Principal Component Analysis and Support Vector Machines | IEEE Conference Publication | IEEE Xplore.” https://ieeexplore.ieee.org/document/9264568 (accessed May 10, 2023).

[26] F. Amiri, M. Rezaei Yousefi, C. Lucas, A. Shakery, and N. Yazdani, “Mutual information-based feature selection for intrusion detection systems,” *J Netw. Comput. Appl.*, vol. 34, pp. 1184–1199, Jul. 2011, doi: 10.1016/j.jnca.2011.01.002.

[27] “IDS 2018 | Datasets | Research | Canadian Institute for Cybersecurity | UNB.” https://www.unb.ca/cic/datasets/ids-2018.html (accessed Mar. 17, 2023).

[28] “cic-ids 2018 ML classification day9 (Unbalanced) | Kaggle.” https://www.kaggle.com/code/kareemayman10/cic-ids-2018-ml-classification-day9-unbalanced (accessed Mar. 17, 2023).

[29] “Method of analyzing computer traffic based on recurrent neural networks - IOPscience.” https://iopscience.iop.org/article/10.1088/1742-6596/1353/1/012133/meta (accessed Mar. 17, 2023).

[30] L. D’hooge, T. Wauters, B. Volckaert, and F. De Turck, “Inter-dataset generalization strength of supervised machine learning methods for intrusion detection,” *J. Inf. Secur. Appl.*, vol. 54, p. 102564, Oct. 2020, doi: 10.1016/j.jisa.2020.102564.

[31] J. Kim, J. Kim, H. Kim, M. Shim, and E. Choi, “CNN-Based Network Intrusion Detection against Denial-of-Service Attacks,” *Electronics*, vol. 9, no. 6, Art. no. 6, Jun. 2020, doi: 10.3390/electronics9060916.

[32] R. Atefinia and M. Ahmadi, “Network intrusion detection using multi-architectural modular deep neural network,” *J. Supercomput.*, vol. 77, no. 4, pp. 3571–3593, Apr. 2021, doi: 10.1007/s11227-020-03410-y.

[33] V. Kanimozhi and T. P. Jacob, “Artificial Intelligence outflanks all other machine learning classifiers in Network Intrusion Detection System on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing,” *ICT Express*, vol. 7, no. 3, pp. 366–370, Sep. 2021, doi: 10.1016/j.icte.2020.12.004.

[34] F. S. de Lima Filho, F. A. F. Silveira, A. de Medeiros Brito Junior, G. Vargas-Solar, and L. F. Silveira, “Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning,” *Secur. Commun. Netw.*, vol. 2019, p. e1574749, Oct. 2019, doi: 10.1155/2019/1574749.

[35] P. Bedi, N. Gupta, and V. Jindal, “Siam-IDS: Handling class imbalance problem in Intrusion Detection Systems using Siamese Neural Network,” *Procedia Comput. Sci.*, vol. 171, pp. 780–789, Jan. 2020, doi: 10.1016/j.procs.2020.04.085.

[36] S. Bagui and K. Li, “Resampling imbalanced data for network intrusion detection datasets,” *J. Big Data*, vol. 8, no. 1, p. 6, Jan. 2021, doi: 10.1186/s40537-020-00390-x.

[37] S. Karim, . R., P. Yunus, P. Khan, and M. Asif, “Implementation of K-Means Clustering for Intrusion Detection,” *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, pp. 1232–1241, Apr. 2019, doi: 10.32628/CSEIT1952332.