1. Introduction

Intrusion detection systems (IDSs) are crucial components in the current computing infrastructures to help monitor and identify undesirable and malicious network traffic (such as unauthorized system access or poorly configured systems) [1]. Along with the growth of network-based applications and systems, the number of cyberthreats is increasing [2]. IDSs play a vital role in cyber security by forewarning security administrators about malicious activities such as distributed denial-of-service (DDoS), port scan, and SQL injection attacks. Having reliable IDSs is a mandatory safeguard for protecting computing infrastructures against ever-increasing issues of intrusive activities.

The idea of creating reliable IDSs with improved accuracy and fewer requirements for human knowledge drives the development of machine learning based IDSs. Machine learning algorithms such as artificial neural networks (ANNs), fuzzy logic, and support vector machines (SVMs) have become extensively used in IDS studies. These machine learning algorithms can extract knowledge from dataset through complex pattern-matching processes. Extracting this knowledge requires most machine learning algorithms to be trained using datasets containing all targeted intrusions.

The requirement of acquiring datasets containing all targeted intrusions raises an important issue. In real-world situations, security experts collect intrusion data incrementally because intrusions do not emerge at once but gradually over time. It is possible to create a new model for these intrusions. However, training a model using a dataset containing all intrusions may takes a long time. Additionally, it is difficult to modify the previous trained to accommodate new intrusion variants because the training process is static and only performed once using datasets containing all targeted intrusions. To solve this problem, we need to develop an algorithm that can learn incrementally with a shorter training time as new intrusions emerge. However, the catastrophic forgetting problem becomes the main challenge to realizing this idea.

Catastrophic forgetting is a classic problem faced by many machine learning models and algorithms. Assume we have trained a classification model; then, we retrain this model using a new dataset containing new classes. In this situation, most current classification models may forget how to classify the old classes. Goodfellow et al. explain that when we train a machine learning model with a convex objective, it will always end with the same configuration at the end of the training process, regardless of how it was initialized. For example, a support vector machine (SVM) that is trained on two different tasks will completely forget how to perform the first task. If this retrained SVM model can correctly classify some data from the old task, it is only due to the similarity of both old and new tasks.

This research aims to solve the two problems I previously mentioned: the problem of ever-growing network intrusion variants and the catastrophic forgetting problem.

To address this problem, I will apply the state-of-the-art architecture in the field of Image Classification with Class Incremental Learning, that is Model-Centric Architecture. This approach use Dynamic Network, which is Network that automatically expands when learning new classes.

II. Related work

1. Intrusion Detections Systems

IDSs are security tools that identify malicious network activities on computer infrastructures. They monitor network traffic and systems logs to find malicious network activities that conventional firewalls cannot filter. There are two main categories of IDSs based on their detection method: signature-based and anomaly based IDSs. Signatured-based IDSs use pattern-matching techniques to find known malicious network activities. Signature-based IDSs are also known as knowledge-based detection or misuse detection. In contrast, anomaly-based IDSs analyze traffic analyze network traffic to a find a significant deviation between observed traffic and acknowledged traffic behaviour. Anomaly-based IDSs interpret this deviation of behaviour as an intrusion. One approach in building anomaly-based IDSs is using machine learning algorithm.

Most of the machine learning model used in previous studies of IDSs are static models and trained using a dataset containing all targeted intrusions. Only a few of them raised the issue of ever-evolving network intrusion variants. Studies by Constantinides et al [3], Chen et al, Yi et al, Xu et al, and Jiang et al are examples of those proposing an incremental learning method to solve the problem of ever-evolving network intrusion variants. Most of these studies utilized support vector machines (SVM) in their proposed incremental learning methods. SVMs belong to the supervised machine learning algorithm category commonly used for classification problems. Despite the prominent properties of SVMs, the training complexity of SVMs is highly dependent of the size of a dataset. Thus, SVMs are not as favored for large-scale data mining as for pattern recognition.

To test the performance of proposed IDS methods, researchers often used publicly available intrusion datasets. In this research, we use CICIDS2017. It is a newer dataset than the KDD Cup 1999 dataset, which has been commonly used in previous IDS incremental learning studies. We did not use the KDD 1999 dataset because it has serveral deficiencies. One of the critical deficiencies of the KDD Cup 1999 dataset is the significant number of redundant records, which causes a bias toward the more frequent records. Another unfortunate deficiency of the KDD Cup 1999 dataset is the fact that this dataset is very old. It was created in 1999 for The Third International Knowledge Discovery and Data Mining Tools Competition. Hindy et al explained that depending solely on old datasets cannot help the advancement of IDSs. Thus, it is better to use newer intrusion datasets that cover recent variants of intrusions.

1. Incremental Learning

Recently, many studies has preferred deep learning using artificial neural networks (ANNs) to process large scale data. Deep learning using ANNs is one of the propular algorithms for learning information from complex datasets. Deep Learning using ANNs can create complex model compared to traditional probabilistic machine learning techniques. Therefore, they have been broadly used for IDS.

III. Method

* 1. Problem Setup and Method Overview

DER model

* 1. The paper DER: Dynamic Expandable Representation for Class Incremental Learning introduces expandable representation. At step , the model is composed of a super feature extractor is build by expanding the feature extractor with a newly created extractor . Specifically, given an image , the feature extracted by is obtained by concatenation as follows

Here re reuse the previous and encourage the new extractor to learn only the novel aspect of new classes. The feature u is then fed into the classifier to make predictions as follows

= Softmax())

Then the prediction y = argmaxp\_h\_t(y|x). The classifier is designed to match its new input and output dimension for step t. The parameters of H\_t for the old features are inherited from H\_t-1 to retain old knowledge and its newly added parameters are randomly initialized.

To reduce catastrophic forgetting, we freeze the learned function at step t, as it captures the intrinsic structure of previous data. In detail, the parameters of last step super-feature extractor , and the statistics of Batch Normalization are not updated. Besides, we instantiate Ft with Ft-1 as initialization to reuse previous knowledge for fast adaptation and forward transfer.

MEMO Model: (Memory-Efficient Expandable Model)

The model from Paper A Model or 603 Exemplars: Toward Memory-Efficient Class-Incremental Learning try to solve the problem of expanding memory by answering the question: Given the same memory budget, if we share the generalized block and only extend specialized blocks for new tasks, can we further improve the performance ?

Concretely, we redefine the model structure by decomposing the embedding module into specialized and generalized blocks. Specialized blocks corresponds to the deep layers in the network, while generalized blocks corresponds to the rest shallow layers. We argue that the features of shallow layers can be shared across different incremental stages, i.e., there is no need to create an extra model

1. Datasets, Pre-processing and Performance Metrics
   1. Datasets

* The experiments were carried out using th ree open-source datasets: ToN\_IoT, CIC-IDS-2017 and KDD99 Dataset. All datasets are widely used in intrusion detection and present different characteristics which can be of value for testing the proposed model.

4.1.1. ToN\_IoT dataset

The ToN\_IoT dataset was collected using a large-scale network created by the University of New South Wales (UNSW) at the Australian Defence Force (ADFA). This network included physical systems, virtual devices, cloud platforms and IoT offering a large number of heterogeneous sources. The data include several captures from devices with different perspective of the network: IoT/IIoT, Network, Linux and Windows. For this set of experiments, the network data was used for the model training. Preference was given to the train\_test\_network data as it provides a sample of the network data, as a single file in CSV format, specially created with the intent of evaluating the efficiency of ML applications. The data contains 43 features in total and includes a large sample of normal traffic plus attack

4.1.2 CIC-IDS-2017 Dataset

The CIS-IDS-2017 was created by the Canadian Institute for Cybersecurity and was specifically designed to help developing solutions to anomaly detection. The dataset contains traffic generated from a network captured over seven days and includes a diverse range of attack scenarios. This is a larger dataset compared to the ToN\_IOT in number of samples, features, and classes. The diversity of data is one of the reasons behind its choice as it offers a more complex environment for network traffic analysis. In total, the dataset contains 79 features with each data sample labeled as either normal or as a specific attack type. A list of all types of attack is presented in Table 4.

[1] Intelligent intrusion detection systems using artificial neural network

Alex Shenfield, David Day, Aladdin Ayesh

[2] Efficient algorithm for intrusion attack by analyzing KDD cup 99

[3] A Novel Online Incremental Learning Intrusion Prevention System (seem like the paper treats the problem as data incremental.

[4] A population-based incremental learning approach with artificial immune system for network intrusion detection system.