VOL. 19, NO. 6, MARCH 2024 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

TOWARDS INTRUSION DETECTION IN IOT USING FEW-SHOT LEARNING

Theyab Althiyabi, Iftikhar Ahmad and Madini O. Alassafi
Faculty of Computing and Information Technology, King Abdulaziz University (KAU), Jeddah, Saudia Arabia
E-Mail: tmalthiyabi@stu.kau.edu.sa

ABSTRACT

The Internet of Things (IoT) is an emerging technology that covers various domains and has become an essential part of the upcoming technological revolution. IoT applications include healthcare, smart-cities, smart-cars, industries, quality of life, and several other fields. IoT typically consists of lightweight sensor devices that facilitate procedures such as automation, real-time trackable data collection, and data-driven decisions. However, securing IoT networks is an accessible research area for several reasons. The main security challenges are limited resources that are incapable of dealing with complex and advanced security tools; and lack of required data for training the security systems like Intrusion detection systems as a result of their heterogeneous nature. This research proposed a Few-shot learning IoT intrusion detection system model based on a Siamese network to overcome the above limitation. The model aims to classify and distinguish normal and attacked traffic. The experiment utilized an IoT dataset in different scenarios to analyze and validate the behavior with three categories with different numbers of data in each. The performance result achieves more than 99% accuracy and shows an efficient detection ability using only less than 1% of the dataset.

Keywords: few-shot learning, intrusion detection system, cyber-security, Internet of Things, Siamese network.

Manuscript Received 19 March 2024; Revised 5 April 2024; Published 15 May 2024

1. INTRODUCTION

Nowadays, the revolution era of technology is rapidly growing, with innovations and enhancements that are regularly updated. However, technology is not only computers, servers, and storage connected to a network, but the services it provides are also technology. Technology revolutionized the world in all fields of life and impacted critical aspects such as countries' economic growth, drivers of rapid inventions and education, and healthcare. The Internet of Things IoT contributes to technology by facilitating communication and data transmission in various domains. IoT is simply a collection of small physical devices and sensors connected to the internet to provide meaningful full real-time data that can, later on, be analyzed as a whole, that ends with meaningful information, and that helps in decision-making [1]. The Dark Side of IoT is the drawbacks in cyber security strength based on its natural diversity, limited resources, and lack of security standards [2]. Hence, securing IoT is one of the trend research areas nowadays with the attempt to fulfill the security gaps.

Intrusion detection system IDS plays a crucial role in network security and is considered one of the robust countermeasures to prevent malicious intrusions in IoT networks. IDS are defined as software or hardware systems that intend to secure the network from unauthorized behavior or abnormal activity that could be security intrusions or breaches [3]. The intrusion detection system has two main types: signature-based and anomaly-based. The signature-based IDS monitors network traffic and reports any matches with well-known registered attacks and intrusion patterns. Whereas anomaly intrusion detection analyses the behaviour of certain networks and traffic flows, keeps them for their normal activity, and alerts for any detected abnormal suspicious activity. Typically, the rate of false alerts, also known as false alarms, is higher in IDS anomaly-based than signature-based. However, the signature IDS cannot detect new intrusion attacks or zero-day attacks as well as it is not regularly updated. That's why, with the increasing number of new intrusions robust anomaly IDS is highly demanded especially in a heterogeneous network such as IoT.

Nowadays, the revolution of networks and digital communication has challenged traditional intrusion detection systems IDSs to deal with rapid network communication with huge amounts of complex data transmission and operations, which leads to poor performance and efficiency by using limited traditional approaches. Artificial intelligence (AI) plays an essential role in devolving current IDSs and enhancing the detection performance. Several research studies investigate both machine learning ML and Deep Learning DL in IDSs. ML-based IDS requires manual feature extraction which, unlike the other advanced AI techniques, gives a low level of performance. In contrast to the suboptimal efficiency of IDS based on ML, Deep-learning (DL) IDSs show tremendous improvements in the performance and automated feature extraction process. However, DL methods require large data and huge computational resources for processing to train the IDSs [4, 5]. Hence, data gathering becomes an issue that limits IDS performance, as data may scarce sometimes, especially in IoT networks. Because of these drawbacks of both ML and DL intrusion detection systems, many scholars investigated the possibility of how to gain a high level of accuracy with minimum data and low resource consumption.

Few-shot learning FSL is one of the emerging AI methods; classified under the umbrella term of meta-learning, also known as learning by learning. FSL overcomes the limitation of data starving in training, as the model can successfully classify rules based on only a few samples. Various methods and approaches help to implement FSL. Metric-based is one of the common FSL approaches that use the nearest distance matrix to compute the similarity among the samples and then classify it according to the calculated distance [6]. Siamese network is a metric-based method that uses distance metrics for classification. The process of measuring the similarities is by comparing two input samples, and computing the similarity of vital feature vectors of these two pairs, the output of this process is to classify based on distance similarity or dissimilarity [7]. Figure-1 illustrates the Intrusion detection system classification and shows exactly where FSL and Siamese networks are categorized under which position in the subclasses. This paper will explore the performance of applying a Siamese network method in an IoT Intrusion detection system.

The paper falls into six parts. First comes an introduction that provides a brief conceptual background of the vital contents. The second section is the literature review. The third section explains the proposed method, while the fourth section shows the experiment and the fifth discusses the result of the experiment. Finally, the last section is the conclusion, followed by the references.

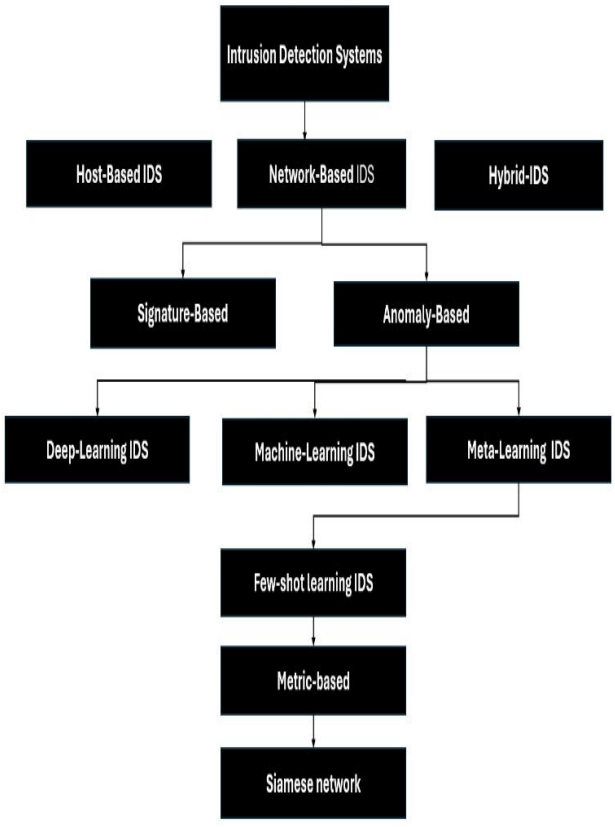

Figure-1. Intrusion detection system's classification.

2. LITERATURE REVIEW

IoT network security challenges the natural diversity of IoT networks, leading to unpredicted intrusions and attacks. The most IoT intrusions and meliaceous attacks are smart and suddenly occur such as zero-day attacks [8]. Yu *et al.* (2023) believe that traditional IDSs for securing IoT networks cannot detect novel attacks, although traditional IDSs may have a high accuracy detection rate it failed in interaction and real-time detection speed. Therefore, Yu *et al*. [9]. Proposed a novel model that has the efficiency of both high detection accuracy and real-time interference speed rate. The model comprises three parts: data processing, data augmentation, and image classification. The first part of the model is developed by utilizing Gramian Angular Fields, and it transmits the real-time traffic from one dimension to two dimensions. In the second part of the model, Denoising Diffusion Probabilistic is mainly employed to overcome the issue of limited samples, while "Few-shot learning" challenges traditional IDS to detect attacks with rare sample data and provides better generalization. The third part of the model uses the architecture of the micro-neural network search approach. The proposed model reached an accuracy of over 99.20% by experimenting with six types of datasets that combine self-constructed and standard datasets such as CICIDS2018, IoT_23, and N-BaIoT. However, the model is extremely complex, which may lead to increased resource consumption which may challenge to implementation of it on limited resource networks as IoT. In addition, the model cannot be generalized [9].

The imbalanced data is another issue in IDS that is linked to scarce data, which requires traditional over-sampling and under-sampling techniques to overcome the data needs IDS training. Bedi *et al.* (2020) developed Siamese Network-based IDS to eliminate the problem of imbalanced data and to perform well with a limited amount of data. The novel model called Siam-IDS uses Siamese networks to compute the distance by a Euclidean distance of two paired networks with the same weight and extracted feature vectors. The output of computing the similarity among paired networks helps in detecting attacks based on the loss similarity results. The experiment was conducted to test the performance of various types of attacks on the NSL-KDD Dataset such as U2R attack, DoS attack, Probe attack, and R2L attack. The developed IDS DNN and CNN models showed better performance in terms of recall compared to other IDS models. However, the model requires more validation as it is applied only to the most common old standard dataset NSL-KDD, and on five categories only. The model may have lower performance in different real types of scare attacks and different datasets [10].

In IoT network security, Thein *et al*. (2023), proposed a network anomaly malicious traffic detection system based on a prototypical graph neural network. Monitoring and filtering network traffic is a crucial security countermeasure to secure IoT networks. The traditional IDS that is based on supervised ML techniques encounters difficulty in dealing with a few of attack

dataset samples, as these data samples are vital in training, and fail in detecting zero-day cyber-attacks for the same reason. Hence, to overcome this problem, an FSL_IDS method is proposed. The method is not dependent on any signature-based or labels of prior knowledge; it converts raw network traffic into image form and applies CNN to extract vital features. Then Euclidean distance calculates the loss function and computes the similarity. The study experiment was conducted on the IoT-23 Dataset by using 5-10 shots (sample) for 4-way (classes) for FSL architecture. The result shows high performance as it reaches an average of 90.72% for the F1 score. Even though the method is validated and reaches better performance than the two other approaches; one fussy graph and the other the prototypical network, only one dataset is used as well as detected for only N-classes meaning it predicts for 4-classes and is not adjustable. In addition, it cannot detect novel attacks [11].

Similarly, Li *et al*. (2022), proposed an IDS model for IoT networks called (RFP-CNN) consisting of Recursive Feature Pyramids (RFP) and Neural Architecture (CNN). The protection of IoT networks is crucial as they are susceptible to be easily exploited by attackers. Securing IoT networks requires advanced tools that can handle the huge variety and heterogeneity of IoT networks as each domain lacks sufficient attack records for training AI tools. The proposed model used a Siamese network to extract features for the domain discriminator. Additionally, it utilized an improved Cascade R-CNN for unsupervised domain adaptation regularization which is adversarial and enhances the efficiency of IDS to detect IoT network. This technique improves the model's precision and makes IDSs more adaptive and robust in detecting network traffic attacks including novel attacks. The model archived promising results, when tested with four datasets, and validated against other models. However, it is more sensitive to data noise and consumes large amounts of resources and time [12].

Miao *et al. (2023)* presented a novel IDS traffic classification model based on Siamese Networks and Prototypical Networks. For several scholars, the main contribution and the value behind developing few-shot learning intrusion detection systems is the ability to deal with a small amount of data in training and to overcome the scarcity of attack data scarcity. Despite, various enhancements added by incorporating FSL in IDS, such as rapid detection and the ability to detect novel types of attacks. The issue of detecting out-of-distribution samples is not extensively addressed. Therefore, the presented model aims to improve efficiency and support the detection of out-of-distribution samples in identifying unknown traffic in IoT networks. The model consists of two frameworks meta-learning and testing-learning. The discriminative features are extracted by CNN and passed to the twin pair networks which then classify traffic based on computed metric distances using an equation. The performance of the presented model achieved 99.33% accuracy with only five samples, which is outstanding compared to other IDS models, especially with the added capability of detecting out of distribution samples. However, the presented supervised approach heavily relies on data labeling and prior knowledge [13].

In encrypted traffic inspection, Yang *et al.* (2023) proposed an enhanced meta-learning model based on multi-task representation for traffic classification using few-shot learning. Encrypting traffic can be costly for security enhancement. Traditional traffic classification based on machine learning and deep learning faces the problem of collecting huge amounts of encrypted traffic and the additional effort in labeling it. Therefore, using of few-shot learning for encrypted traffic will cause to reduce the number of labels, which is practical. The proposed model tackled the issue of multi-classification tasks for encrypted traffic and improved the differences in traffic representation using the flow discrepancy enhancement module. The model shows high performance in encrypted traffic classification. However, it cannot identify OOD samples as it is limited to classifying ID numbers of samples [14].

Huang *et al.* (2020) presented a model for anomaly detection based on network structure incorporating a gate for dealing with imbalanced data. The supervised learning approach typically requires massive efforts in labeling huge amounts of data samples. Various methods cannot detect unknown types, as they depend only on labelled data even though they use anomaly-based few-shot learning. The proposed model aims to solve the issue of imbalanced data, by presenting a network structure that acts as a robust gate to distinguish and define if the test sample is seen or unseen from a set of samples that are the support set preemptively. The model is based on similarity metrics that utilize CNN to act as an encoder. The model is tested with a common standard dataset NSL-KDD, and the accomplished result is reasonable. However, replacing feature extractions with meta-learning may lead to instability and inefficient detection. Hence, further validation is needed to prove the above facts with various dataset validations [15].

In smart home intrusion detection, Chen *et al.* (2023) proposed a method called the "EM-FEDE Enhancement Method based on Feature Enhancement and Data Enhancement". The smart home is one of the IoT network applications, the number of which is rapidly increasing, making security a major concern. Malicious intrusions in smart homes may compromise privacy and damage vital devices and the main controller of smart homes. Due to the diversity of smart homes and lack of standards for collecting data samples for training IDS-based ML, DL is challenging. The proposed methods enhance the scenarios of having only a few shot samples in developing smart home IDS. The method consists of three parts. The first part is a feature enhancement that works on analyzing historical data of smart homes and analyzing it to ensure the data quality and extend features based on that. The second part is the data enhancement, which includes data filtering, removing duplication and unnecessary values. The final part is similarity measures, which mainly use Wasserstein distance. The generator that produces a variant range of counterfeit samples does this. On the other hand, the discriminator works as a

differentiator between authentic and counterfeit examples. The proposed method shows outstanding performance, exceeding a 21.9% improvement in accuracy. The experiment validates various IDS methods, but the optimal expansion ratio is not specified in the study [16].

Table-1. Literature Review Summary.

Ref	Author	Year	Method	Performance	Limitation
[9]	Yan *et al.*	2023	GAF Gramian Angular Fields, DDPM Denoising Diffusion Probabilistic and variable network ETNet V2 neural architecture	99.20%	Complex and not validated in terms of generalization
[10]	Bedi *et al.*	2020	Siamese Neural Network	Recall the highest 91.22%, and 55.22% lowest	Needs more validation as - it is validated by only one common old dataset that has five classes.
[11]	Thein *et al.*	2023	Prototypical Graph, Neural Network	F1 Score average 90.72%	Cannot detect novel attacks. Only one dataset is used
[12]	Li *et al.*	2022	Recursive Feature Pyramids RFP and Neural Architecture CNN	Accuracy 96.68%	Sensitive to noise in data. Consume large amounts of resources and time.
[13]	Miao *et al.*	2023	Siamese Prototypical Network	Accuracy 98.33% with 5 samples only	The presented supervised approach heavily relies on data labelling and prior knowledge.
[14]	Yang *et al.*	2023	Enhanced meta-learning model based on multi-task representation	F1 +90%	cannot identify OOD samples as it is limited to classify ID numbers of samples
[15]	Huang *et al.*	2020	Network structure incorporating a gate and CNN	Overall accuracy 84.70% using 5-shot	- No feature extraction. - Needs more validation. -One dataset is used.
[16]	Chen *et al.*	2023	Feature Enhancement and Data Enhancement	More than 21.9% accuracy improvement	The experiment validates various IDS methods, and the optimal expansion ratio is not specified in the study
[17]	Ayesha S *et al.*	2023	Few-Shot Self- Supervised	The highest F1 score in accuracy is 98.01% and the lowest is 59.60% with 5- shots.	Complex and may challenge generalization. Low result for one dataset

Ayesha S *et al*. [17] presented an IoT intrusion detection system framework based on a self-supervised few shot. Unlike other traditional networks, IoT networks spread rapidly with various heterogeneous functions and vendors, lacking standards, diversity, and several other aspects that challenge the security level of IoT networks. In contrast, traditional IDS require massive data for training, which is scarce in some cases in IoT networks, despite other issues related to imbalanced datasets. Therefore, the proposed framework aims to fill the gaps and overcome the challenges of collecting and labelling the enormous amount of data for training IDS. The framework is named FS3, which stands for "Few-Shot Self- Supervised". FS3 encompasses three phases: The first phase is handling imbalanced data by learning hidden patterns using self-supervised learning. The second phase combines contrastive learning and few-shot learning to enable IDSs to perform efficiently with a few labelled samples by computing the loss among classes using triplet loss and Multi-Similarity Miner. The third one is an extended classification of the sub-sample using the K-Nearest Neighbor to further enhance the performance of imbalanced sample classification. The experiment is conducted on three datasets including BoT-IoT and shows a significant improvement of up to 43.95% in terms of F1 score. However, the F1 score result of BoT-IoT is very low and the gap among/with the other highest exceeds 38%. Hence, it may be challenging to generalize the framework as it requires more validation [17].

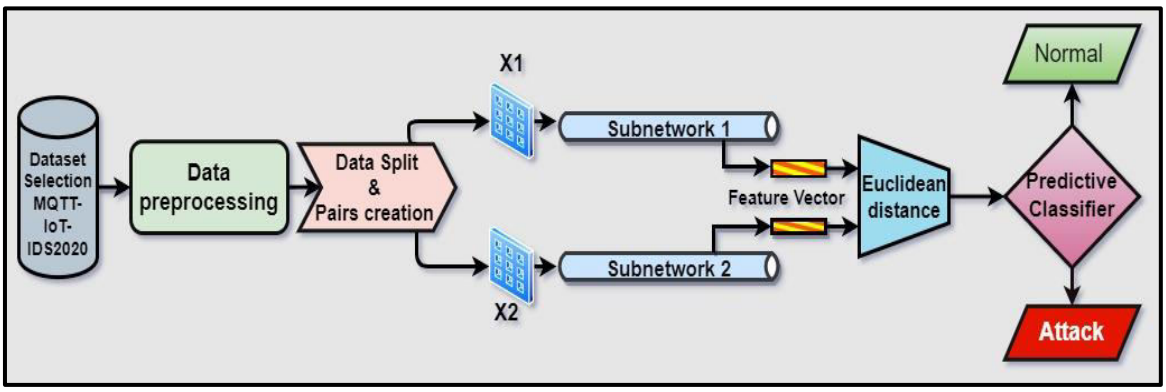

Figure-2. Proposed model brief background.

3. METHODOLOGY

The proposed methodology is based on three main phases: dataset selection and processing, Siamese network architecture model, and predictive classifier. Figure-1 Illustrate a brief of the three phases in the proposed model.

A. Dataset Selection and Processing

a) Dataset Selection

The dataset is selected based on several scholars' reviews and to fulfill the requirements of developing an IoT intrusion detection system. The dataset should be an IoT dataset containing different types of common IoT intrusion. The MQTTIOT2020 dataset is recent and is the first dataset of its kind that is publicly available. The dataset is comprehensive, and variant compared to other IoT datasets, simulated MQTT IoT network. The chosen dataset MQTT-IoT-IDS2020, introduced by Hanan *et al.* [18], based on a designed simulated IoT network. The dataset simulation process of the IoT MQTT network includes several connected sensor devices, a broker, cameras, and attackers. The recorded data consists of five main scenarios tracking common attack types in IoT networks, with four intrusion attack classes, and one normalclass. Table-2; describes each attack in MQTT-IoT-IDS2020. All captured raw data is merged into feature extraction and recorded in a CSV file. The features are extracted at three levels: Packet, Unidirectional flow, and Bidirectional flow. In our model, we focus on bidirectional flow features. The dataset adds value to research in the field of intrusion detection as the CSV file is usable and contains vital features adaptable for training, testing, and processing in developing IDSs scholar such as [19], and [20] utilize this dataset in their experiments.

b) Data Pre-processing

The total number of records in MQTT-IoT-IDS2020 bidirectional flow exceeds 259380 in the CSV file. In few-shot learning, the aim is to achieve high accuracy by using a few samples for training and testing. Therefore, in our model less than 0.5% of the total number is used. For the preprocessing process, we cleaned the data, removed all duplicated and non-valued records, and performed normalization. Besides, we balanced the data into two binary classes: attack (1) which includes a combination of the four attacks, (MQTT Brute-Force, scan_A, scan_sU, and Sparta) as described in Table-1 and normal traffic indicated as (0) label. Furthermore, the model depends on data splitting to increase the number of the support set and query set, with an inverse relationship between them. Splitting the data into training and testing facilitates the evaluation model for robust generalization assessment, avoiding over-fitting and enabling comparative analysis [21], [22]. The Siamese network requires two pairs of samples for each training and testing. Hence, we generate several X paris (X1, X2) samples, where each sample is paired with another sample in the same class to measure the similarity score for training and later evaluation in testing.

Table-2. Attack types used in the MQTT-IoT-IDS2020 dataset.

Attack Type	Description
The MQTT Brute-Force	Attempt to access the network by systematically entering various potential credentials to infer MQTT credentials
Aggressive scan (scan_A)	Scanning tool that aims to explore vulnerability in uncovering accessible ports and services within the network.
UDP Scan (scan_sU)	Malicious scanning attack that explores vulnerability by scanning UDP ports for any weakness in the network
Sparta SSH Brute-Force (sparta)	Continuous iterated systematic attempts to discover valid SSH credentials

B. Siamese Network Architecture

The first step in Siamese network architecture is to create a base network. The base network is a feed-forward neural network that includes two identical sub-networks as illustrated in Figure-2 Sub-network1 and Sub-network2. Each sub-network consists of five layers: an Input layer, three hidden layers, and an output layer. The first layer is the input layer, which handles the input in two sub-network paths. The input is in the shape of the previously generated pre-processed twin samples Paris X1 and X2, with embedded 27 features. The input is associated with a flattened layer that merges the input into a dimensional array. The three hidden dense layers; each contains the ReLU activation and 128-unit neurons. The output layer is a feature vector that represents the learned embedding features. The feature vector is used in further distance computation. The output feature vector from sub-network 1 is compared with the feature vector out-put by sub-network 2. The distance is computed using Euclidean distance. In training, lists of pairs of samples X1 and X2 that have the same class are processed together through sub-networks 1 and 2. The output feature vectors from both networks are computed and compared in terms of distance to learn the similarity and loss distance based on the similarity of these two pairs. This process is repeated in several batch iteration epochs and tasks for both sub-networks until the model is well-trained and learns the distance between these two binary classifications: 0 for normal and 1 for attack. The contrastive loss is defined in the model for rapid learning, associated with the Adam optimizer.

C. Predictive Classifier

Since the model is well-trained based on the support set similarity score of the distance comparison of the two classes of the two pairs using Siamese network architecture. Then in the same way, the model trained, and it will predict the distance score for the two pairs of unlabelled query sets in testing the model. The distance threshold from training is utilized for converting predicted distances into binary predictions. The class normal is represented as 0, while class 1 is represented as an attack. The evaluation process contains a pair list of batches of query sets. If the computing distance is greater than the trained distance threshold, it will be classified as 0 normal which is similar. If not, it will be dissimilar, in this case 1, which indicates the attack class. The model performance is evaluated by computing a confusion matrix as shown in Table-3. True similar represents true positive, while true dissimilar corresponds to true negative, false similar is equal to false positive, and false dissimilar refers to false negative. Furthermore, the performance is evaluated by using various other matrices such as accuracy, F1 score, precision, and recall.

Table-3. Confusion matrix for performance evaluation.

Confusion Matrix		
	Normal (0)	**Attack (1)**
Normal (0)	Ture Similar	Fale Dissimilar
Attack (1)	False Similar	True Dissimilar

4. EXPERIMENT

The experiment was conducted by using a laptop running on an Intel Core (TM), i5-7200U 2.50GHz processer, 2701 MHz, with 2 Core(s), and 12GB of RAM. The software used included: Python 3.10 64bit, Visual Studio Code under an Anaconda 2.3.2-isolated environment, and the main programming libraries utilized were: Tensor-Flow, Keras, NumPy, Matplotlib, Scikit-learn, Seaborn, and Pandas.

Since the model is based on Few-shot learning, the experiment utilized less than 1% of the total downloaded CSV dataset file. The dataset was divided into three sets of balanced data: the first set consisted of 120 total samples, the second set had 280 samples, the third set contained 520 samples. This division allowed for an efficient evaluation of the proposed model and analysis of its behaviour with varying number of trained and tested data. The experiment relied on the data split for training and testing, defining the number of support sets and query sets based on the data split for binary classification. All attack types in the dataset were merged into one class labelled as 1, while normal data was labelled as 0. This data split added more adaptability and may enhance the generalization of the model.

The model was trained on pairs of training sets, and each half of the pairs is processed separately in one of the identical twin networks that share the same weight. The raining of model was for 20 epochs and 16 size of the batch. The first dataset included 120 balanced samples: 60 attacked and 60 normal samples. Then, data was divided into two sets: support set represents training and query set represents testing and evaluation. The data deviation in support set and query set was experimented with in four scenarios. First split the data into 80:20, which means that 80% of the data was used as training (Support Set), while the remaining 20% was used as testing (Query Set). The second scenario was of 60:40, means 60% training and 40% testing. The third scenario consisted of 40:60, 40% for training and 60 % for testing. The fourth scenario used 20:80 means 20% for training and 80% for testing. The same approach and scenarios were applied for all three sets (120, 280, and 520). This included the pairs' generation for the twin networks. Table-4. Illustrates all data information and details of the number of samples.

Table-4. Number of samples in each set.

Set	Split	#TR	#TS	#TR_ P	#TS_P	0:1 TR	0:1 TS
120	80:20	96	24	6863	412	49:47	13:11
	60:40	72	48	3848	1700	38:34	26:22
	40:60	48	72	1688	3836	28:20	40:32
	20:80	24	96	416	6860	14:10	50:46
280	80:20	224	56	37520	2324	112:112	28:28
	60:40	168	112	21075	9343	87:81	59:53
	40:60	112	168	9348	21080	58:54	86:82
	20:80	56	224	2320	37516	30:26	114:110
520	80:20	416	104	129575	8051	211:205	55:49
	60:40	312	208	7816	32308	162:150	110:98
	40:60	208	312	32308	7816	110:89	162:150
	20:80	104	416	8051	129575	55:49	211:205

Key:
SET: Total number of samples.
#TR: Number of training samples after the split.
#TS: Number of testing samples after the split.
#TR_ P: Number of training sample pairs.
#TS_P: Number of testing samples pairs.
0:1 TR: Number of attack and normal samples in training as 0 normal and 1 for attack.
0:1 TS: Number of attacks and normal samples in testing as 0 normal and 1 attack.

5. RESULTS AND DISCUSSIONS

The Siamese network model experiment was conducted on less than 1% of the MQTT-IoT-IDS2020 dataset, and the results are based on the divided sets; illustrated in Table-4. This division carried three parts: 120, 280, and 520 sets with the following split approaches for training and testing. The first split is 80:20, the second is 60:40, the third is 40:60, and the fourth is 20:80.

A. Result for the Set of 120 of MQTT-IoT-IDS2020.

The total amount of data in this set is 120, and the results are based on the separation of training and testing data and includes four scenarios. The first split for 80:20 reached 100% accuracy, F1 score, precision, and recall. The accuracy for the second split, 60:40 is 98.76%, while the F1 score is 98.07%, precision is 100%, and recall is 96.22%. The third split for 40:60 reached an accuracy of 98.20%, 97.22% for the F1 score, 100% for precision, and 94.59 % for the recall. The accuracy for the last split 20:80 is 83.20% and for the F1 score is 65.79%, for precision is 100%, and for the recall is 49.02%. Figure-3 provides a comparative summary for all results of the 120' Set and Figure-4 shows the percentage of the confusion matrix. Furthermore, the time taken to train the Siamese model is illustrated in Table-5.

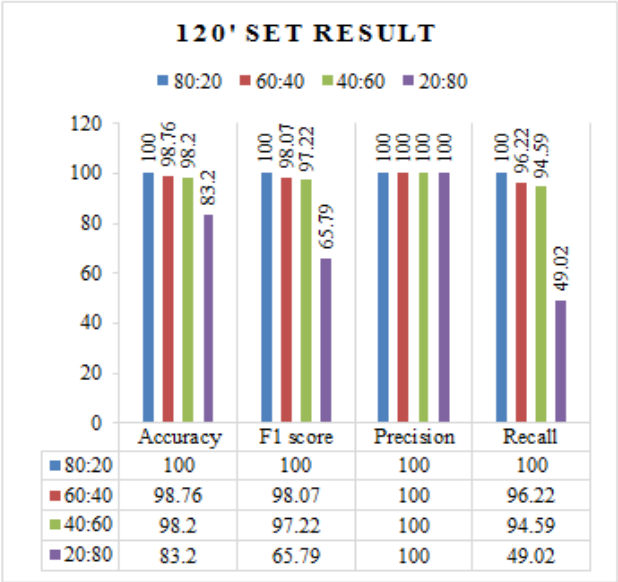

Figure-3. 120 Set of MQTT-IoT-IDS2020.

Table-5. Training time for 120' Set.

Split Scenarios	Training Time
80:20	24.1s
60:40	17.5s
40:60	8.3s
20:80	4.4s

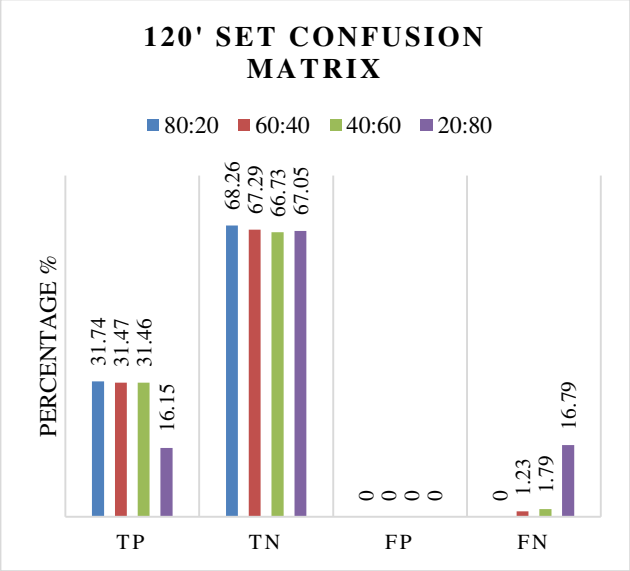

Figure-4. Confusion matrix of the 120' Set.

B. Result of the Set 280 of MQTT-IoT-IDS2020.

Figure-5. The 280 Set of MQTT-IoT-IDS2020.

The second experiment was conducted on MQTT-IoT-IDS2020 using 280 data for training and testing. The results are illustrated in Figure-5. For the first scenario where 80% of the data is assigned for training and 20% for testing, which achieved 98.83% accuracy, F1 achieved 98.18%, precision had 100%, and the recall achieved 96.43% accuracy. In the second scenario split of 60:40, has 99.44% accuracy, the F1 has 99.15%, the precision has 100%, and the recall has 93.31%. The third scenario split 40:60 achieved 99.59% accuracy, F1 got 99.38, the precision got 100%, and the recall got 98.78%. The last scenario split of 20:80 obtained 97.69%accuracy, F1 has 96.39%, the precision remains the same with the other split at 100%, and the recall has 93.03%. The training time is shown in Table-6 and confusion matrix is illustrated in Figure-6.

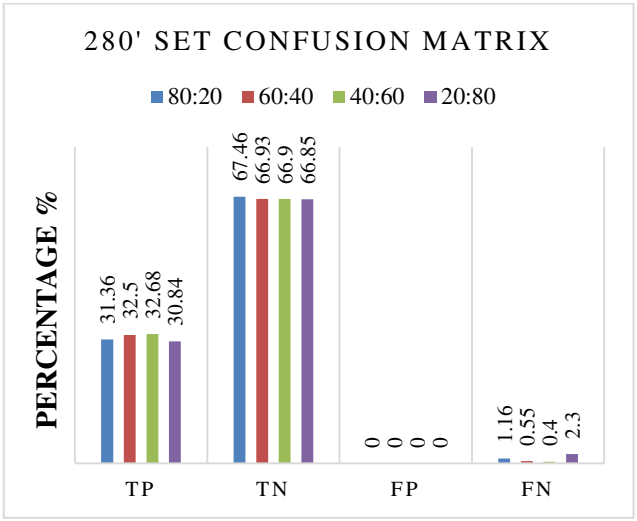

Figure-6. Confusion matrix of the 280' Set.

Table-6. Training time for 280' Set

Split Scenarios	Training Time
80:20	2 m 34.8s
60:40	1m 47.8s
40:60	1m 5.0s
20:80	29.2s

C. Results for the Set of 520 of MQTT-IoT-IDS2020

The last experiment was conducted on a larger amount of data, totalling 520 for both training and testing. In the first split, with 80 for training and 20 for testing, the accuracy result was 96.12%, with F1 score of 94.27%, precision 92.17%, and recall of 96.46%. In the second scenario with the split of 60:40 for training and testing, the accuracy was 97.4%, the F1 score was 96.09%, the precision was 95.98%, and the recall was 96.2%. In the third scenario, with the split of 40:60, the accuracy was 98.27%, the F1 score was 97.4%, the precision was 97.36%, and the recall was 97.44%. In the last scenario where the data was divided into 20 for training and 80 for testing, the accuracy was 98.41%, the F1 score was 97.59%, the precision was 98.08%, and the recall was 97.11%. The results are illustrated in Figure-7, the confusion matrix is shown in Figure-8, and the training time is shown in Table-7.

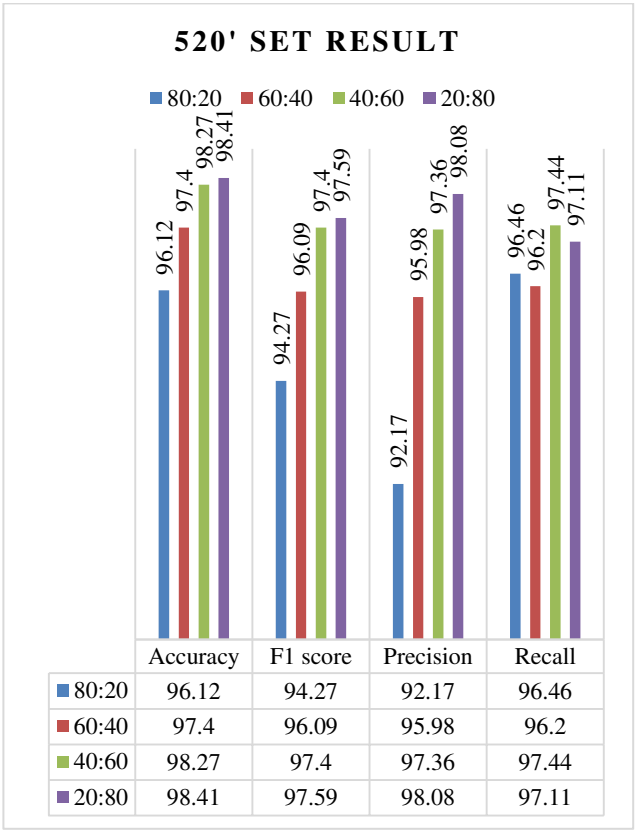

Figure-7. 520 Set of MQTT-IoT-IDS2020.

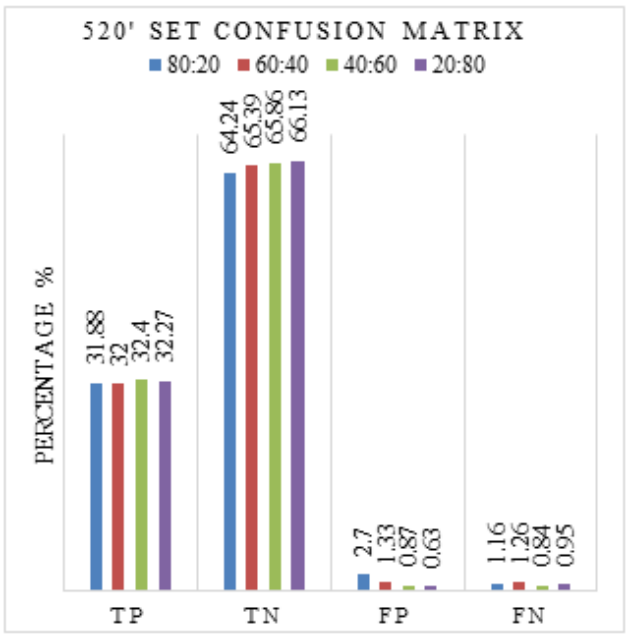

Figure-8. Confusion matrix of the 520' Set.

Table-7. Training time for 520' Set.

Split Scenarios	Training Time
80:20	10m 46.1s
60:40	7m 21.4s
40:60	4m 22.3s
20:80	1m 43.9s

D. Comparative Analysis and Discussions

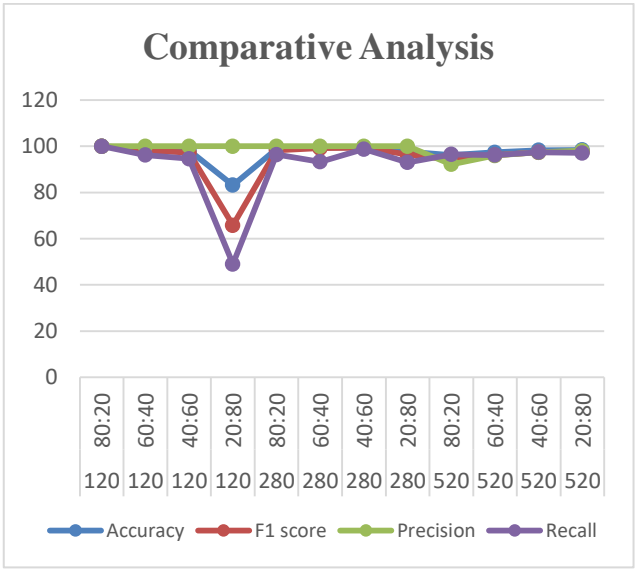

Figure-9. Comparative analysis linier result.

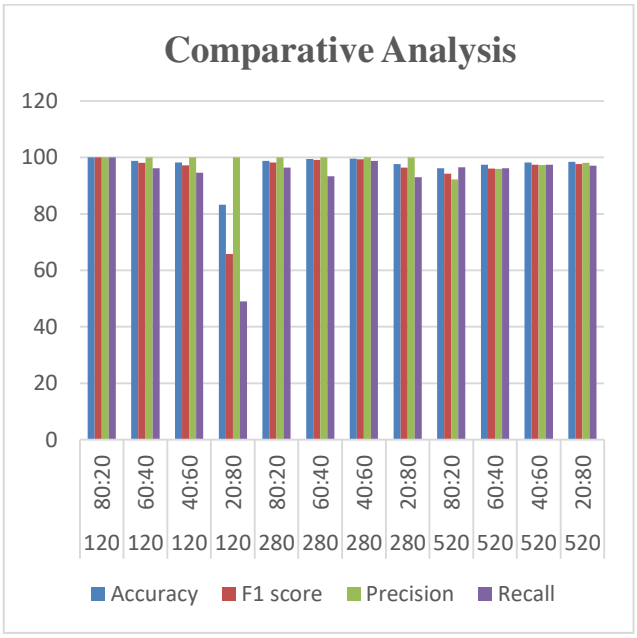

Figure-10. Comparative analysis result of the experiments.

The results of the three experiments show accuracy, F1 score, precision, and recall for all three data-sets (120, 280,520) and the four scenarios are above 95%

as shown in Figures 9, and 10. The only exception is 20:80 split of with the lowest data-set of 120 sample, where the model was trained with only 24 samples. Additionally, the training time is reduced when using fewer samples, which align with the lightweight resources typically used in IoT networks. The highest performance result was achieved with100% accuracy when the experiment was conducted with a total number of 120 data points, split 80:20 for training and testing. The lowest accuracy was observed in the same 120 data-set specifically in the 20:80 split. Overall, the model's performance in the conducted experiments with different scenarios is promising and could enhance IoT networks security. The MQTT-IoT-IDS2020 dataset, to the best of our knowledge, has not been used in intrusion detection systems based on Few-shot learning before, making it challenging to compare the results with similar approaches. However, using less than 1% of the data, the proposed model shows outstanding performance compared to other works such as [23], [24], and [25]) which utilize the same dataset with a larger amount of data for developing IDS models based on various DL and ML techniques. In addition, reached close performance compared to recent work [26] conducted using an FSL prototypical network with multiclass classification evaluated with the same dataset.

CONCLUSIONS

In this work, we propose an intrusion detection system model that is suitable for IoT networks as it uses only few-shot learning instead of the massive data needed in traditional IDS based on deep learning and machine learning techniques. The proposed model is based on a Siamese network that mainly depends on a distance vector metric to compute and learn the similarity in two pairs of dataset samples. Since this model is for IoT security, we used the IoT dataset MQTT-IoT-IDS2020 to validate the model. The experiment was conducted in three different sets based on the number of data for each experiment as well as the division of the dataset into training and testing for each set. The results of the experiment vary based on the number of data used in the experiment. However, in general, the performance of the proposed model for binary classification is outstanding in terms of accuracy, precision, F1 score, and recall. The model shows that the Siamese network in IDSs performs effectively when the dataset is scarce. Furthermore, IDS based on FSL can contribute to reducing the number of resources as well as minimizing the complexity and training time in IDS, which is crucial to fulfilling the requirements of an IoT network, where only limited resources and lightweight devices exist. The plan is to evaluate the behaviour of the model with the full dataset. However, that is inapplicable due to the limitation of the current computation resources we use for the experiment. This can be a future work direction as well as evaluating the model with different datasets.

REFERENCES

[1] S. Lam, S. T. Siddiqui, A. Ahmad, R. Ahmad, and M. Shuaib. 2020. Internet of Things (IoT) enabling technologies, requirements, and security challenges. in Advances in Data and Information Sciences, Singapore: Springer. pp. 119-126.

[2] Rachit S. Bhatt and P. R. Ragiri. 2021. Security trends in Internet of Things: a survey. SN Appl. Sci., 3(1): 121, doi: 10.1007/s42452-021-04156-9.

[3] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin and K.-Y. Tung. 2013. Intrusion detection system: A comprehensive review. J. Network Comput. Appl. 36(1): 16-24.

[4] P. Sun *et al*. 2020. DL-IDS: Extracting Features Using CNN-LSTM Hybrid Network for Intrusion Detection System. Security and Communication Networks, 2020: 1-11, doi: 10.1155/2020/8890306.

[5] P. Dini, A. Elhanashi, A. Begni, S. Saponara, Q. Zheng and K. Gasmi. 2023. Overview on Intrusion Detection Systems Design Exploiting Machine Learning for Networking Cybersecurity. Applied Sciences, 13(13): 7507, doi: 10.3390/app13137507.

[6] K. He, N. Pu, M. Lao, and M. S. Lew. 2023. Few-shot and meta-learning methods for image understanding: a survey. Int. J Multimed Info Retr, 12(2): 14, doi: 10.1007/s13735-023-00279-4.

[7] J. J. Valero-Mas, A. J. Gallego, and J. R. Rico-Juan. 2023. An overview of ensemble and feature learning in few-shot image classification using siamese networks. Multimed Tools Appl.doi: 10.1007/s11042-023-15607-3.regularization technique. Journal of Network and Computer Applications. 191: 1-18.

[8] A. A. Orunsolu, A. S. Sodiya, and A. Akinwale. 2022. A predictive model for phishing detection. J. King Saud Univ.-Comput. Infor. Sci. 34: 232-247.

[9] Y. Yan, Y. Yang, F. Shen, M. Gao and Y. Gu. 2023. GDE model: A variable intrusion detection model for few-shot attack. Journal of King Saud University - Computer and Information Sciences, 35(10): 101796, doi: 10.1016/j.jksuci.2023.101796.

[10] P. Bedi, N. Gupta and V. Jindal. 2020. Siam-IDS: Handling class imbalance problem in Intrusion Detection Systems using Siamese Neural Network. Procedia Computer Science, 171: 780-789, doi: 10.1016/j.procs.2020.04.085.

[11] T. T. Thein, Y. Shiraishi and M. Morii. 2023. Few-Shot Learning-Based Malicious IoT Traffic Detection with Prototypical Graph Neural Networks. IEICE Trans. Inf. & Syst., E106. D(9): 1480-1489, doi: 10.1587/transinf.2022OFP0004.

[12] K. Li, W. Ma, H. Duan, H. Xie and J. Zhu. 2022. Few-shot IoT attack detection based on RFP-CNN and adversarial unsupervised domain-adaptive regularization. Computers & Security, 121: 102856, doi: 10.1016/j.cose.2022.102856.

[13] G. Miao, G. Wu, Z. Zhang, Y. Tong and B. Lu. 2023. SPN: A Method of Few-Shot Traffic Classification with Out-of-Distribution Detection Based on Siamese Prototypical Network. IEEE Access, 11: 114403-114414, doi: 10.1109/ACCESS.2023.3325065.

[14] Yang *et al.* 2023. Few-shot encrypted traffic classification via multi-task representation enhanced meta-learning. Computer Networks, 228: 109731, doi: 10.1016/j.comnet.2023.109731.

[15] S. Huang *et al.* 2020. A Gated Few-shot Learning Model for Anomaly Detection. in 2020 International Conference on Information Networking (ICOIN), pp. 505-509. doi: 10.1109/ICOIN48656.2020.9016599.

[16] Y. Chen, J. Wang, T. Yang, Q. Li and N. A. Nijhum. 2023. An Enhancement Method in Few-Shot Scenarios for Intrusion Detection in Smart Home Environments. Electronics, 12(15): 3304, doi: 10.3390/electronics12153304.

[17] Ayesha S., S. A. B., and M. D. 2023. FS3: Few-Shot and Self-Supervised Framework for Efficient Intrusion Detection in Internet of Things Networks. in Annual Computer Security Applications Conference, Austin TX USA: ACM, pp. 138-149. doi: 10.1145/3627106.3627193.

[18] H. Hindy, E. Bayne, M. Bures, R. Atkinson, C. Tachtatzis and X. Bellekens. 2021. Machine Learning Based IoT Intrusion Detection System: An MQTT Case Study (MQTT-IoT-IDS2020 Dataset). in Selected Papers from the 12th International Networking Conference, vol. 180, B. Ghita and S. Shiaeles, Eds., in Lecture Notes in Networks and Systems, vol. 180. , Cham: Springer International Publishing, pp. 73-84. doi: 10.1007/978-3-030-64758-2_6.

[19] Y. Liu, Y. Zhou, K. Yang and X. Wang. 2023. Unsupervised Deep Learning for IoT Time Series. IEEE Internet of Things Journal, 10(16): 14285-14306, doi: 10.1109/JIOT.2023.3243391.

[20] L. D. Manocchio, S. Layeghy and M. Portmann. 2022. Network Intrusion Detection System in a Light Bulb. Presented at the 2022 32nd International Telecommunication Networks and Applications Conference (ITNAC), IEEE Computer Society, pp. 1-8. doi: 10.1109/ITNAC55475.2022.9998371.

[21] R. Xu, L. Xing, S. Shao, B. Liu, K. Zhang and W. Liu. 2022. Co-Learning for Few-Shot Learning. Neural Processing Letters. 54(3): 3339-33561.

[22] [R. Zhang and Q. Liu. 2023. Learning with few samples in deep learning for image classification, a mini-review. Front. Comput. Neurosci., 16: 1075294, Jan. 2023, doi: 10.3389/fncom.2022.1075294.

[23] M. A. Khan, M. A. Khan, S. U. Jan, J. Ahmad, S. S. Jamal, A. A. Shah, N. Pitropakis and W. J. Buchanan. 2021. A Deep Learning-Based Intrusion Detection System for MQTT Enabled IoT. Sensors, 21(7016), [Online]. Available: https://doi.org/10.3390/s21217016

[24] S. Chesney and K. Roy. 2022. AI Empowered Intrusion Detection for MQTT Networks. in Proceedings of the 2022 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), Durban, South Africa, pp. 1-6. [Online]. Available: https://doi.org/10.1109/icABCD54961.2022.9856124

[25] Mosaiyebzadeh, L. G. Araujo Rodriguez, D. Macedo Batista and R. Hirata. 2021. A Network Intrusion Detection System using Deep Learning against MQTT Attacks in IoT. in Proceedings of the 2021 IEEE Latin-American Conference on Communications (LATINCOM), Santo Domingo, Dominican Republic, pp. 1-6. [Online]. Available: https://doi.org/10.1109/LATINCOM53176.2021.9647850.

[26] T. Althiyabi, I. Ahmad and M. O. Alassafi. 2024. Enhancing IoT Security: A Few-Shot Learning Approach for Intrusion Detection. Mathematics, 12(7): 1055, doi: 10.3390/math12071055.