Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 171 (2020) 780–789

Procedia Computer Science

www.elsevier.com/locate/procedia

Third International Conference on Computing and Network Communications (CoCoNet'19)

Siam-IDS: Handling class imbalance problem in Intrusion Detection Systems using Siamese Neural Network

Punam Bedi[a], Neha Gupta[b,*], Vinita Jindal[c]

[a,b]Department of Computer Science, University of Delhi, Delhi 110007, India
[c]Keshav Mahavidyalaya, Department of Computer Science, University of Delhi, Delhi 110007, India

Abstract

To tackle new and complex attacks, modern Intrusion Detection Systems (IDSs) are developed using Deep Learning (DL) techniques and are trained on intrusion detection datasets such as KDD and NSL-KDD. These two datasets have a large number of samples in Denial of Service and Probe attack classes besides the Normal class, but the number of samples in Remote to Local (R2L) and User to Root (U2R) attack classes is very less. R2L and U2R attacks represent the minority classes of these two datasets and due to lack of training samples in these minority classes, DL based IDSs are unable to detect them accurately. This leads to class imbalance problem and increases the chances of the network being compromised due to undetected intrusions. To handle this class imbalance problem in IDSs, this paper proposes Siam-IDS which is a novel IDS based on Siamese Neural Network (Siamese-NN). The proposed Siam-IDS is able to detect R2L and U2R attacks without using traditional class balancing techniques such as oversampling and random undersampling. The performance of Siam-IDS was compared with existing IDSs developed using DL techniques namely Deep Neural Network (DNN) and Convolutional Neural Network (CNN). Siam-IDS was able to achieve higher recall values for both R2L and U2R attack classes when compared to its counterparts.

Keywords: Intrusion Detection System (IDS); Siamese Neural Network; Class imbalance problem; Remote to Local attack (R2L); User to Root attack (U2R); NSL-KDD dataset

* Corresponding author.
E-mail address: neha.phd.2018@gmail.com

1877-0509
10.1016/j.procs.2020.04.085

1. Introduction

With the growth of the Internet and availability of digital devices, the amount of data being exchanged over computer networks has increased exponentially. These networks have become an easy target for hackers who try to compromise the confidentiality, integrity or availability of online data. To secure this data from malicious access and manipulation, Intrusion Detection Systems (IDSs) were introduced. IDSs monitor and analyze the data being exchanged over a network to identify intrusions. Based on the type of intrusion detection technique, IDSs can be categorized as Signature based IDSs (S-IDSs) and Anomaly based IDSs (A-IDSs). S-IDSs identify intrusions by matching incoming network traffic with previously known intrusion patterns. This type of IDSs is capable of detecting known attacks, but they are unable to identify new types of attacks. In contrast, A-IDSs create a model of normal network behavior and raise an alert for every network activity that deviates from the established normal profile. This makes A-IDSs sensitive to previously known as well as unknown intrusions. In this paper, we aim to enhance the intrusion detection capability of A-IDSs.

In recent times, A-IDSs have been developed using supervised Deep Learning (DL) techniques for identifying novel network attacks [4]. DL algorithms provide very accurate classifications when trained on a large amount of data that is evenly distributed among different target classes. But their performance degrades when either the total amount of training data is insufficient or the number of training samples in some classes is far more than the samples in other classes of the dataset. The latter scenario is referred to as a class imbalance problem and the datasets having such uneven data distribution are called imbalanced datasets. In such cases, the target classes having a dominant share in the imbalanced dataset are known as majority classes, while the classes which are outweighed by majority classes are known as minority classes.

Intrusion detection datasets such as KDD and NSL-KDD are examples of imbalanced datasets in which Normal network traffic samples represent the largest majority class and attack categories such as Remote to Local (R2L) and User to Root (U2R) represent minority classes. Since the number of samples in these minority classes contribute less than 1% of the total training dataset, IDSs trained using DL techniques on imbalanced datasets perform poorly when classifying R2L and U2R attacks. In literature, researchers tried to handle the class imbalance problem in IDSs by using traditional methods such as oversampling, random undersampling and Synthetic Minority Oversampling Technique (SMOTE). Oversampling replicates the samples of the minority class, random undersampling eliminates the samples of majority class randomly and SMOTE creates new synthetic samples of the minority class. Though these methods have been used extensively, but each of them has its disadvantages. Oversampling leads to overfitting of minority class, random undersampling may lead to loss of important information from majority class, while synthetic samples may not accurately represent the minority class and lead to overlapping among classes.

To handle the class imbalance problem in IDSs and correctly predict both majority and minority classes, in this paper we use a Few-Shot Learning technique called Siamese Neural Network (Siamese-NN). Siamese-NN method computes the similarity between its inputs to differentiate between similar and dissimilar samples. In this paper, a novel IDS named Siam-IDS has been developed using Siamese-NN. The proposed Siam-IDS handles the class imbalance problem in IDSs without using traditional class balancing mechanisms such as oversampling, random undersampling and SMOTE. The training and testing of the proposed system were carried out using the NSL-KDD dataset. Siam-IDS achieves higher recall values when compared with IDSs developed using DL techniques such as Deep Neural Network (DNN) and Convolutional Neural Network (CNN).

The remaining paper is organized as follows: Section 2 presents a literature review of class imbalance problem in IDSs and recent works utilizing Siamese-NN; Section 3 gives the details of Siamese-NN and NSL-KDD dataset; Section 4 describes the architecture of the proposed Siam-IDS; Section 5 explains the experimental setup followed for training and testing the proposed system; Section 6 presents the results followed by Section 7 that concludes this paper.

2. Related Work

In the field of A-IDSs, the class imbalance problem is still counted in the list of existing challenges. Rodda et. al. [9] analyzed the performance of Naïve Bayes (NB), BayesNet, J48 Decision Tree and Random Forest (RF) classifiers for IDSs and found that all these classifiers performed poorly in U2R minority class. Several efforts have been made to improve the detection accuracy of minority classes present in intrusion detection datasets. In their work, Sun et. al. [12] created a double layer model for efficient intrusion detection using KDD dataset. They oversampled the minority class using synthetic samples and used Gradient Boosting Decision Tree, K-Nearest Neighbor and Fly Optimization Algorithm to segregate normal and attack classes. Folino et. al. [13] developed a distributed Cellular Genetic Programming framework for combining ensemble of intrusion detection classifiers and achieved high values for precision and recall. In his work, Dada [14] proposed a hybrid approach for combining Support Vector Machine (SVM), K Nearest Neighbor and Primal-Dual Particle Swarm Optimization to achieve high accuracy for IDSs. The work of Yuan et. al. [15] used a combination of C5 Decision Tree and NB algorithm to develop A-IDS. Although the aforementioned works tried to address the class imbalance problem for A-IDSs, but each of them was trained and tested using KDD dataset. This dataset has received much criticism by the research community due to the presence of duplicate, redundant and obsolete samples which do not represent modern-day intrusions. For these reasons A-IDSs trained on KDD tend to perform poorly when deployed in real-world environment. KDD dataset was succeeded by NSL-KDD dataset and overcomes many of its drawbacks [16]. So, in this paper NSL-KDD dataset has been selected for training and testing of the proposed system.

In literature, several researchers have utilized traditional methods for handling the class imbalance problem by using minority class oversampling or majority class undersampling methods. Abdulhammed et. al. [17] compared the performance of RF, DNN and Variational Auto-Encoder with and without the use of traditional balancing techniques. Chowdhury et. al. [6] developed a Few-Shot Learning technique of extracting features from different layers of CNN and used them as inputs to SVM and DNN. The efficiency of this work was checked on KDD and NSL-KDD datasets after balancing them using traditional balancing approaches. Jiang et. al. [5] combined RF with Rough Set Theory to develop an A-IDS by using NSL-KDD. To improve the intrusion detection process, the dataset was balanced using oversampling prior to experimentation. Though these works improved the detection rate of minority classes, but each of them used either oversampling or random undersampling technique. Oversampling or replication of minority class samples leads to overfitting of these classes, while undersampling or random elimination of majority classes may lead to loss of crucial information.

To address the class imbalance issue, Chawla et. al. [10] introduced SMOTE which synthesizes new minority samples between existing minority samples present in the dataset. Parsei et. al. [7] combined SMOTE with cluster center and nearest neighbor (CANN) technique to improve the class-wise detection capability of CANN. Seo et. al. [11] tried to optimize SMOTE ratios to improve the detection of imbalanced classes in KDD dataset. Since synthetic samples may not be true representatives of the minority class, use of SMOTE may lead to the addition of noise and cause overlapping among classes. The proposed Siam-IDS handles the class imbalance problem without using any of the three traditional techniques (oversampling, random undersampling, SMOTE) of dataset balancing. It makes use of Siamese-NN which was originally developed by Bromley et. al. [1] for signature verification. In the recent past, Siamese-NN has been used for different tasks including image recognition [3], age estimation [18], gait recognition [19], object tracking, sentence similarity etc. The novelty of the proposed work is that it utilizes Siamese-NN for handling class imbalance problem in IDSs and to the best of our knowledge no existing research work has used it in this manner.

3. Background Concepts

This section explains the general architecture and working of Siamese Neural Network (Siamese-NN) followed by the details of the NSL-KDD intrusion detection dataset.

3.1. Siamese Neural Network

Siamese-NN is a Few-Shot Learning technique that was introduced by Bromley et. al. [1] for signature verification. It computes the similarity between a pair of inputs to differentiate between similar and dissimilar samples. It ensures that inputs that belong to the same class have high similarity value and inputs belonging to different classes have low similarity score. For similarity computation, Siamese-NN extracts the feature representations of its inputs and computes the similarity value between them using a distance function. The Siamese-NN must be trained to represent similar inputs with similar feature representations and dissimilar inputs with feature representations that are not similar to each other. The architecture of Siamese-NN consists of two identical neural networks which have the same set of weights. Each of these two sub-networks accepts an input and extracts its feature representation during the training phase. The two feature vectors are then input to a distance function that calculates the similarity score between them. If the inputs to the Siamese-NN are same or similar to each other, then the Siamese-NN is trained to output a high similarity value and if the inputs are dissimilar then the Siamese-NN must learn to output a low similarity value. To ensure that the Siamese-NN output remains unaffected by changing the order of inputs in the input pair, same set of weights must be used for both the sub-networks. Identical weights also guarantee that both the sub-networks learn same feature representation for the same input. Figure 1 depicts the architecture of a simple Siamese-NN.

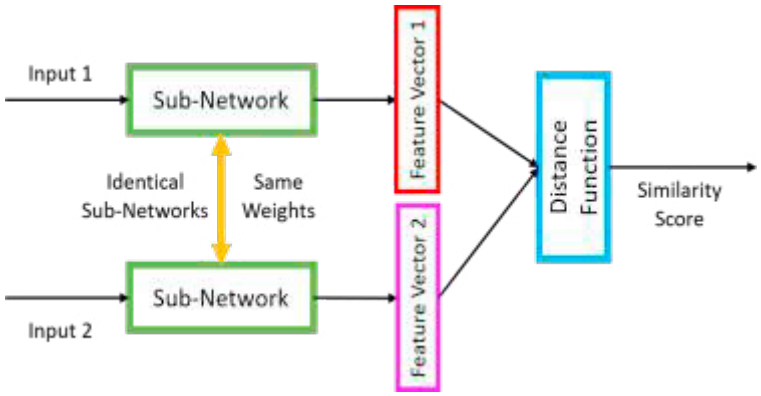

Fig. 1. Architecture of a simple Siamese Neural Network

3.2. NSL-KDD Dataset

To overcome the disadvantages of the KDD dataset, NSL-KDD dataset was developed by Tavallaee et. al. [2]. The NSL-KDD training and testing datasets contain 1,25,973 samples and 22,544 samples respectively. Each sample in the two sets comprises of 41 features (3 categorical and 38 numerical) and is categorized as Normal or a type of attack. For the purpose of experimentation, we grouped together different attack types into four main attack classes namely Denial-of-Service attack (DoS) attack, Probe attack, Remote to Local (R2L) attack and User to Root (U2R) attack. Table 1 specifies the class-wise distribution of the number of samples in NSL-KDD training and testing set. Since the number of R2L and U2R attack samples is less than 1% in the training dataset, they form the two minority classes of NSL-KDD. Table 2 summarizes the types of attacks present in each of the four main attack classes in NSL-KDD dataset. The sub-categories marked in bold are only present in testing dataset and are absent from the training dataset.

Table 1. Class-wise distribution of NSL-KDD samples

Class Name	Train set count	Attack % in Train set	Test set count	Attack % in Test set
Normal	67343	53.45	9711	43.07
DoS	45927	36.45	7458	33.08
Probe	11656	9.25	2421	10.73
R2L	995	0.007	2887	12.80
U2R	52	0.0004	67	0.002
Total	125973		22544	

Table 2. Types of attacks in four major attack classes

Attack Class	Sub-categories of Attack Class
DoS	back, land, neptune, pod, smurf, teardrop, **apache2, mailbomb, processtable, udpstorm**
Probe	ipsweep, satan, portsweep, nmap, **mscan, saint**
R2L	multihop, imap, gues_pswd, ftp_write, phf, spy, warezclient, warezmaster, **httptunnel, named, sendmail, worm, xlock, snmpgetattack, snmpguess, xsnoop**
U2R	buffer_overflow, loadmodule, perl, rootkit, **ps, sqlattack, xterm**

4. The proposed Siam-IDS architecture

In this paper, Siamese-NN has been used to develop a novel IDS named Siam-IDS. The proposed Siam-IDS consists of two identical DNNs comprising of an input layer, five hidden layers and four dropout layers. The hidden layers are made of 1024, 512, 256, 128 and 64 neurons respectively. Rectified Linear Unit (ReLU) activation function is used in each hidden layer. A dropout layer is present between consecutive hidden layers and specifies a dropout rate of 0.5. The selection of the aforementioned hyperparameters has been done after comparing different combinations of hidden layers (3,4,5), hidden neurons (256-128-64, 512-256-128-64, 1024-512-256-128-64) and activation functions (ReLU, Exponential Linear Unit). To compute the similarity, the extracted feature vectors are input to the 'distance layer' which uses the Euclidean distance to find how close or far apart the computed feature representations are with respect to each other. Feature representations of similar inputs lie close to each other and have a smaller Euclidean distance, while feature representations of dissimilar inputs lie far apart and have large Euclidean distance values. Figure 2 shows the architecture of the proposed Siam-IDS.

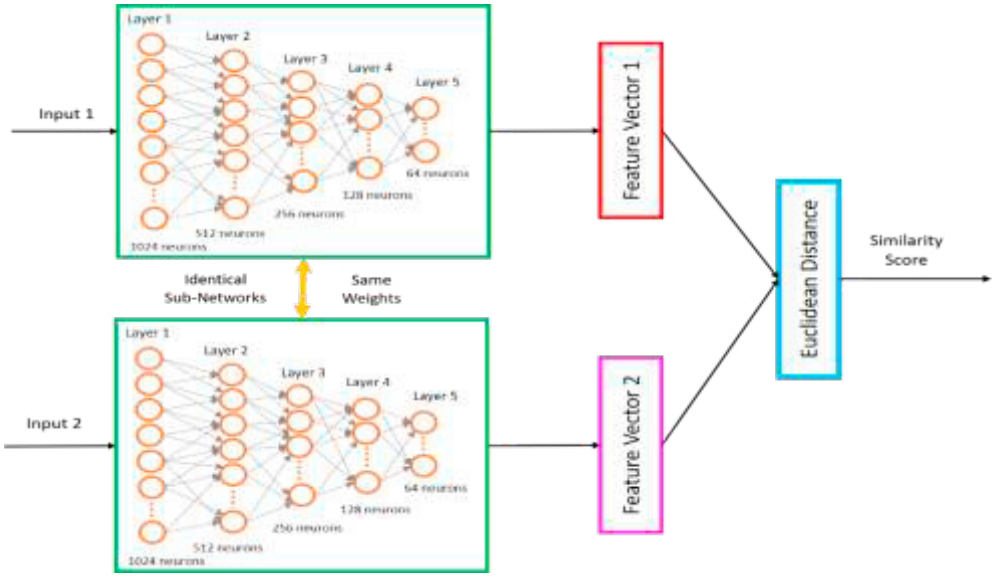

Fig. 2. Architecture of Siam-IDS

5. Experimental Setup

The proposed Siam-IDS was designed and developed on Windows 10 system having an Intel® Core™ i7-8750H processor. The proposed system was developed in Python language using Keras API. The development of Siam-IDS consisted of three main phases: Dataset Preparation phase, Training phase and Testing phase. Each of these phases has been discussed below. Figure 3 depicts these three phases of Siam-IDS.

In the Dataset Preparation phase, the three categorical features (namely protocol, service and flag) of NSL-KDD training and testing datasets were converted to numeric values. This was done by assigning a unique value to each category present in these features. The five class labels were also assigned unique values (Normal: 0, DoS: 1, Probe: 2, R2L: 3, U2R: 4) in both the datasets. In addition, the values of all the features were normalized to map them to a uniform range. After these pre-processing steps, fifty training samples from each of the above mentioned five classes were selected for the training phase. This formed the training data subset Tr'. The number fifty was selected because the smallest minority class U2R contains only fifty-two training samples. It was ensured that the attack sub-types present in each of the four main attack classes are represented equally within the fifty samples of their respective class. Further, this training subset of two hundred and fifty samples (fifty samples for each of the five classes) was used to create similar and dissimilar input pairs for Siam-IDS. Similar pairs consisted of both inputs belonging to the same class, i.e. Normal-Normal, Dos-DoS, Probe-Probe, R2L-R2L and U2R-U2R, while dissimilar pairs consisted of training samples belonging to two different classes such as Normal-Probe, DoS-R2L, U2R-Probe etc. Each similar pair was assigned the label 1 and a 0 label was given to each dissimilar pair. As the two sub-networks present in Siam-IDS are symmetric, the order of inputs in dissimilar pairs is not important i.e. a dissimilar pair such as Probe-U2R was considered same as U2R-Probe pair and only one of them was included in the training set. A total of 31,226 labeled input pairs was created and used for training the Siam-IDS.

In the training phase, the Adam optimizer was used to minimize the Contrastive loss function. The choice of Adam optimizer was made after comparing its performance with Rmsprop optimizer function. Contrastive loss function is a distance-based function that minimizes the distance between similar pairs and maximizes the distance between dissimilar pairs [8]. Equation 1 defines the contrastive loss function.

$$L(W, (X_1, X_2, Y_t)) = ((1/2 * Y_t * {D_W}^2) + ((1/2) * (1 - Y_t) * \max(0, \mathrm{m} - D_W)^2) \quad (1)$$

Here L is the Contrastive loss function, W is the set of shared weights, X_1 and X_2 are feature representations of a pair of inputs, Y_t is the binary training label that depicts whether the pair of inputs is similar ($Y_t = 1$) or dissimilar ($Y_t = 0$), D_W is the Euclidean distance between X_1 and X_2, m is the margin which is taken to be 1 in the experiments. With these hyperparameters, Siam-IDS was trained for multiple epochs using the training subset of input pairs created during the data preparation phase.

Once the training was complete, the Siam-IDS was tested on preprocessed NSL-KDD testing dataset. For this purpose, we adopted the approach followed by Koch et. al. (Koch, Zemel and Salakhutdinov 2015) in their work. Ten samples from each of the five classes (Normal, DoS, Probe, R2L, U2R) were selected and each of them was paired with every test sample. Using the label of the test sample from the test dataset and the labels of the selected training samples from the training dataset, new binary labels were assigned to these test pairs. A label of 1 was assigned to test pairs in which the test sample and the training sample were similar, while a label of 0 was given to dissimilar pairs. On testing the proposed Siam-IDS using these labeled test pairs, ten similarity scores were obtained per class for each test sample. A class-wise average of these scores was computed to obtain five averaged similarity scores – one for each class. The class with the maximum averaged similarity score was selected as the final prediction for that test sample. The predictions made by the Siam-IDS for the test samples were then compared with the binary similarity labels that were assigned to them.

To compute the efficiency of Siam-IDS, recall and precision values were computed and compared with corresponding values obtained from IDSs developed using DNN and CNN. The DNN that was used for comparison consisted of five hidden layers with 1024, 512, 256, 128 and 64 neurons respectively. ReLU activation function was used in each hidden layer. A dropout layer was present between consecutive hidden layers having a dropout rate of 0.5. While the CNN consisted of three 1D convolutional layers with 64, 128, 256 neurons using 3*3 filters and ReLU activation function. Every convolutional layer was followed by a max-pooling layer of pool size 2 with and a

dropout layer with dropout rate of 0.5. The last layer in both these networks consisted of 5 output nodes and softmax activation function. The results of the conducted experiments have been mentioned in the following section.

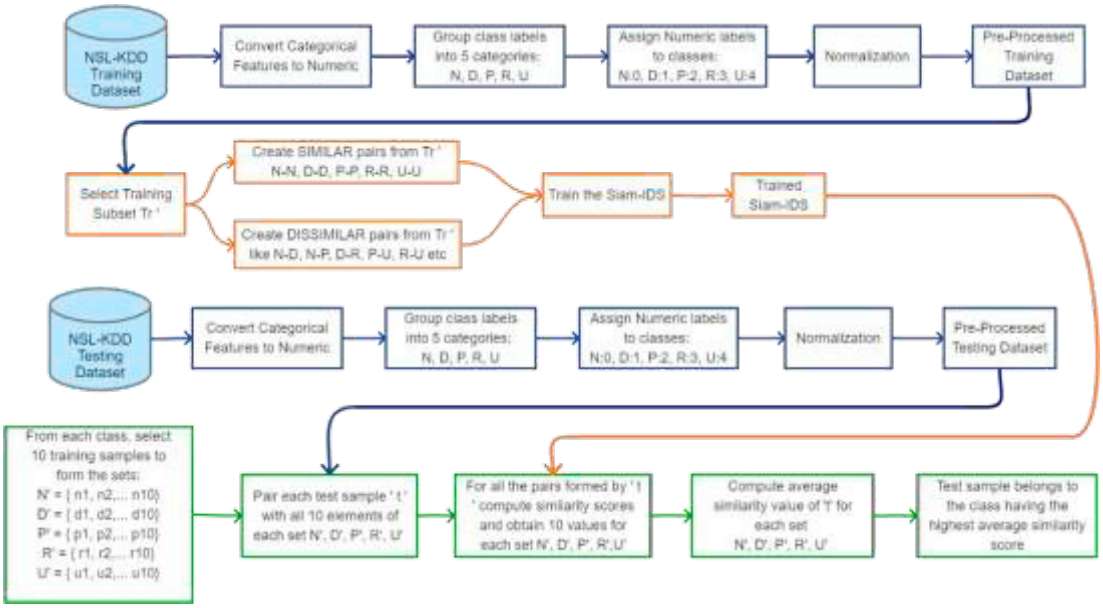

Fig. 3. The data preparation phase, training phase and testing phase of Siam-IDS

6. Results

To check the effectiveness of Siam-IDS, we compared the recall and precision values of the proposed system with the values obtained from the IDSs developed using DNN and CNN. For a given class A, recall indicates that out of all the test samples belonging to class A, how many test samples were correctly identified as class A samples by the classifier. On the other hand, precision indicates that out of all the test samples that the classifier identified as belonging to class A, how many of those predicted samples actually belonged to class A. In case of IDSs, a high recall value is very important so that most of the intrusions are correctly identified by the IDS and minimal number of intrusions go undetected. The formulae for recall and precision are given by equation 2 and equation 3 respectively.

$$Recall = TP/(TP + FN) \quad \ldots (2) \qquad Precision = TP/(TP + FP) \quad \ldots (3)$$

The recall and precision values for Siam-IDS and IDSs developed using DNN and CNN for 250, 500 and 1000 epochs have been shown in Table 3 and Table 4 respectively. These results have also been represented graphically in Figure 4 and Figure 5.

On comparing the performance of Siam-IDS with DNN and CNN based IDSs, it was found that there was a significant increase in the recall values of both minority classes i.e. R2L attack and U2R attack in case of Siam-IDS. This highlights the efficiency of the proposed system in handling the class imbalance problem present in intrusion detection datasets. Moreover, the recall value of one of the majority classes viz. DoS also improved when using Siam-IDS. For Normal and Probe classes, the recall values achieved by Siam-IDS were close behind the two DL based IDSs. In case of precision, Siam-IDS did not give good results for the two minority classes of the dataset. This indicates that many test samples that did not belong to the minority classes were classified as minority class samples by the proposed system. However, for DoS and Probe classes the performance of Siam-IDS was comparable with the DL based IDSs for all different epochs. In addition, Siam-IDS also outperformed DNN and CNN based IDSs for all the epochs in case of Normal class.

Table 3. Recall values for DNN, CNN and Siam-IDS

RECALL (%)						
Epochs	Classifier	Classes				
		Normal	DoS	Probe	R2L	U2R
250	DNN	97.42	77.42	69.76	5.78	19.4
	CNN	97.32	78.43	49.48	5.54	28.36
	Siam-IDS	89.65	83.35	66.67	**32.35**	**55.22**
500	DNN	97.3	81.52	66.01	4.43	10.45
	CNN	97.62	81.28	50.56	5.37	31.34
	Siam-IDS	92.99	84.16	65.76	**30.2**	**56.72**
1000	DNN	97.35	81.03	55.64	4.36	4.48
	CNN	97.48	76.98	49.57	6.44	17.91
	Siam-IDS	91.22	85.37	48.66	**33.25**	**56.72**

Table 4. Precision values for DNN, CNN and Siam-IDS

PRECISION (%)						
Epochs	Classifier	Classes				
		Normal	DoS	Probe	R2L	U2R
250	DNN	67.29	96.14	76.88	62.31	92.86
	CNN	64.36	96.38	74.83	96.97	76
	Siam-IDS	77.7	87.33	73.36	63.75	6.65
500	ANN	67.65	96.35	76.46	76.19	77.78
	CNN	65.35	96.48	78.16	97.48	72.41
	Siam-IDS	77.53	91.46	74.53	66.92	6.39
1000	DNN	66.61	94.05	75.84	85.14	100
	CNN	63.97	96.31	75.81	97.89	92.31
	Siam-IDS	77.13	85.35	75.18	57.94	10.11

Fig. 4. Graphical representation of Recall values (%) obtained for DNN, CNN and the proposed Siam-IDS

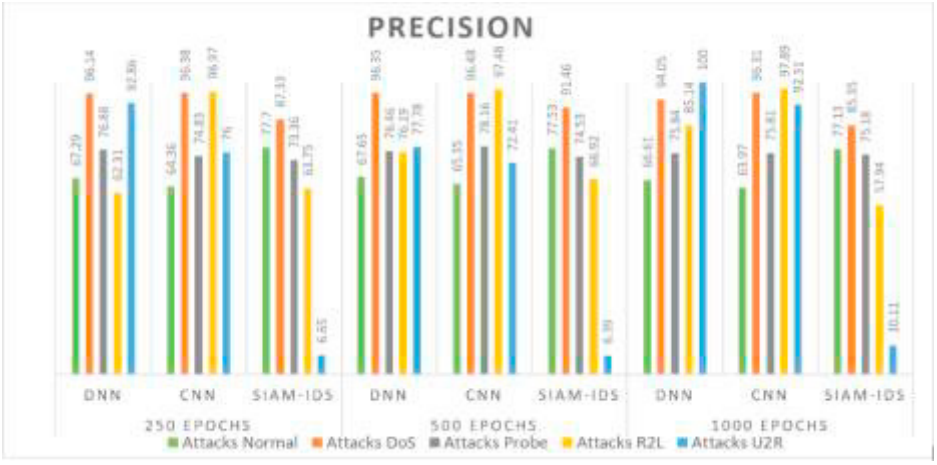

Fig. 5. Graphical representation of Precision values (%) obtained for DNN, CNN and the proposed Siam-IDS

Recall measures the proportion of actual attacks that were identified correctly as attacks and precision measures the proportion of actual attacks identified out of the predicted attacks for a particular attack class. While a low recall value is a sign of undetected attacks, a low precision score depicts the presence of false predicted attacks (false alarms). Although every IDS aims to achieve high precision by reducing the number of false alarms, but identifying a larger number of intrusions and achieving high recall is the primary responsibility of every IDS. This is because undetected intrusions are more harmful than false alarms that did not identify any real intrusion. False alarms can be handled by a security administrator unlike unidentified intrusions which may threaten the security of the network.

7. Conclusion

In this paper, a novel IDS named Siam-IDS has been developed using Siamese Neural Network to handle the class imbalance problem in NSL-KDD dataset. The developed system computes the similarity score between input pairs to identify samples belonging to same class from those belonging to different classes. Siam-IDS was trained using similar and dissimilar input pairs created from NSL-KDD training dataset. Contrastive loss function was used in Siam-IDS to maximize the similarity between similar input pairs and minimize the similarity between dissimilar input pairs. The efficiency of Siam-IDS was tested by pairing each test sample from NSL-KDD testing dataset with few samples from all five classes of the training dataset. During experimentation, the performance of Siam-IDS was evaluated against IDSs developed using DNN and CNN. It was found that the recall values obtained from Siam-IDS were significantly higher for both the minority attack classes as compared to DNN and CNN based IDSs. Though the proposed system displayed high recall values, its precision for minority classes was lower than IDSs based on DNN and CNN. Recall and precision values for the three majority classes were comparable for Siam-IDS and its counterparts. All these results highlight the efficiency of Siam-IDS in detecting minority attacks present in imbalanced NSL-KDD dataset.

Acknowledgment

The authors acknowledge University Grants Commission for partially funding the work in this paper via Junior Research Fellowship Ref. No. 3505 (NET-NOV-2017).

References

[1] Bromley, Jane, Guyon, Isabelle, LeCun, Yann, Sickinger, Eduard, and Shah, Roopak. (1994) “Signature Verification using a Siamese Time Delay Neural Network”. Advances in neural systems information processing: 737-744.
[2] Tavallaee Mahbod, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani. Canadian Institute for Cybersecurity, University of New Brunswick. (2009) Available from: https://www.unb.ca/cic/datasets/nsl.html.
[3] Koch, Gregory, Richard Zemel, and Ruslan Salakhutdinov. (2015) “Siamese Neural Networks for One-Shot Image Recognition” International Conference on Machine Learning. Lille, France: 1-8.
[4] Liu, Yuchen, Shengli Liu, and Xing Zhao. (2017) “Intrusion Detection Algorithm Based on Convolutional Neural Network”. 4th International Conference on Engineering Technology and Application (ICETA 2017): 9-13.
[5] Jiang, Jianguo, Qiwen Wang, Zhixin Shi, Bin Lv, and Biao Qi. (2018) “RST-RF: A Hybrid Model based on Rough Set Theory and Random Forest for Network Intrusion Detection”. International Conference on Cryptography, Security and Privacy. Guiyang, China: ACM: 77-81.
[6] Chowdhury, Md Moin Uddin, Frederick Hammond, Glenn Konowicz, Jiang Li, Chunsheng Xin, and Hongyi Wu. (2017) “A Few-shot Deep Learning Approach for Improved Intrusion Detection". IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON). New York, NY, USA: IEEE: 456-462.
[7] Parsaei, Mohammad Reza, Samaneh Miri Rostami, and Reza Javidan. (2016) “A Hybrid Data Mining Approach for Intrusion Detection on Imbalanced NSL-KDD Dataset”. International Journal of Advanced Computer Science and Applications 7 (6): 20-25.
[8] Hadsell, Raia, Sumit Chopra, and Yann LeCun. (2006) “Dimensionality Reduction by Learning an Invariant Mapping”. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). New York, NY, USA: IEEE: 1735-1742.
[9] Rodda, Sireesha, and Uma Shankar Rao Erothi. (2006) “Class Imbalance Problem in the Network Intrusion Detection Systems”. International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT). Chennai, India: IEEE: 2685-2688.
[10] Chawla, Nitesh V., Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. "SMOTE: Synthetic Minority Over-sampling Technique" Journal of Artificial Intelligence Research 16 (2002): 321–357.
[11] Seo, Jae-Hyun, and Yong-Hyuk Kim. (2018) “Machine-Learning Approach to Optimize SMOTE Ratio in Class Imbalance Dataset for Intrusion Detection”. Computational Intelligence and Neuroscience. Hindawi: 1-11.
[12] Sun, Chong, Kun Lv, Changzhen Hu, and Hui Xie. (2018) “A double-layer detection and classification approach for network attacks” 27th International Conference on Computer Communication and Networks (ICCCN). Hangzhou, China: IEEE: 1-8.

[13] Folino, Gianluigi, and Francesco Sergio Pisani. (2015) "Combining Ensemble of Classifiers by Using Genetic Programming for Cyber Security Applications". European Conference on the Applications of Evolutionary Computation. Springer, Cham: 54-66.
[14] Dada, E G. (2017) "A hybridized SVM-kNN-pdAPSO approach to intrusion detection system" Proc. Fac. Seminar Series: 14-21.
[15] Yuan, Yali, Liuwei Huo, and Dieter Hogrefe. (2017) "Two Layers Multi-class Detection method for network Intrusion Detection System" IEEE Symposium on Computers and Communications (ISCC). Heraklion, Greece: IEEE: 767-772.
[16] Gupta, Neha, Punam Bedi, and Vinita Jindal. (2019) "Effect of Activation Functions on the Performance of Deep Learning Algorithms for Network Intrusion Detection Systems". International Conference on Emerging Trends in Information Technology. Delhi, India: Springer Nature Switzerland: 1-12.
[17] Abdulhammed, Razan, Miad Faezipour, Abdelshakour Abuzneid, and Arafat AbuMallouh. (2019) "Deep and Machine Learning Approaches for Anomaly-Based Intrusion Detection of Imbalanced Network Traffic" IEEE Sensor Letters 3 (1): 1-4.
[18] Jeong, Yoosoo, Seungmin Lee, Daejin Park, and Kil Houm Park. (2018) "Accurate Age Estimation Using Multi-Task Siamese Network-Based Deep Metric Learning for Frontal Face Images" Symmetry 10 (385): 1-15.
[19] Zhang, Cheng, Wu Liu, Huadong Ma, and Huiyuan Fu. (2016) "Siamese neural network based gait recognition for human identification". International Conference on Acoustics, Speech and Signal Processing (ICASSP). Shanghai, China: IEEE: 2832-2836.