NETWORK INTRUSION DETECTION SYSTEM USING DEEP LEARNING METHOD WITH KDD CUP'99 DATASET

Authors:

Mohamed Hamada,

Presenter: Jesse Jeremiah Tanimu,

Mohammed Hassan,

Patience Robert,

Anand Mahendran

November, 2022

Overview

1. Introduction
2. Problem Statement
3. Aim and objectives
4. Related Work
5. Methodology
6. Results and Discussion
7. Summary and conclusion
8. Future Work
9. References

Introduction

- **Intrusion Detection System (IDS)**
- IDS is a system that monitors network traffic for suspicious activity and issues alert when such activity is discovered.
- It is a software application that scan a network or a system for harmful activity or policy breaching.
- Any malicious venture is normally reported either to an administrator or collected centrally using a security information/event management system.

Introduction

- **Types of IDS**
- **Anomaly-based IDS** : it keeps track of activities within the specific scope, looking for instances of malicious behavior as they can define it. This is difficult because it can lead to false positives. For instance, outbound Uniform Resource Locator (URLs) of web activity might be considered and sites involving certain domains or URL length/contents might automatically be blocked even though it's a human being trying to go there not malware and that user may have a legitimate reason.
- The major challenge of anomaly-based IDS is defining its rule set. The efficiency of the system depends on how well it is implemented and tested on all protocols. But once the rules are defined and protocol is built then anomaly-based IDS works well.
- It major advantage over signature-based engines is that a novel attack for which a signature does not exist can be detected if it falls out of the normal traffic patterns.

Introduction

- **Types of IDS**
- **Signature-Based IDS** :It is also known as knowledge based IDS, involves looking for specific signatures byte combination that when they occur almost invariably imply bad news.
- These solutions generate fewer false positives than anomaly-based IDS solutions because the search criterion is so specific but they also only cover signatures that are already in the search database (which means novel attacks can have access into the network).
- In this technique, novel attacks cannot be detected and can easily deceived because they are only based on regular expressions and strings.
- At the heart of the concept of IDS is the simple fact that machines, neural networks and deep learning algorithms can be used to identify normal traffic and attacks on the computer network.

Problem Statement

- Several attempts have been made in the past to mitigate the problem of network intrusion.
- Sparse autoencoders was applied on network intrusion detection. In a bit to impose sparsity to their work on network intrusion detection, Al-Qatf, M., Lasheng, Y., Al-Habib, M. and Al-Sabahi, K., (2018) employed KL divergence regularization technique and Wahyudi, B. (2018), combined KL divergence and L2 regularization techniques.
- Even though L2 regularization technique work well with high-dimensional data where many features are correlated, we found out that L2 regularization technique has interpretative issues, that is L2 regularization does not promote sparsity.
- Hence, the need for a far better regularization technique and even a suitable activation function for our proposed model in order to have an improved detection accuracy.

Aim and Objectives

AIM

- To propose an improved deep sparse autoencoder driven network intrusion detection system, that has linear behaviour, representational sparsity and addresses the problem associated with model interpretation in order to improve the limited accuracy of existing solutions.

OBJECTIVES

- To address model interpretation issues in order to provide a more accurate model that promote sparsity.
- To apply a more effective ReLU activation function in order to achieve representational sparsity and linear behaviour.

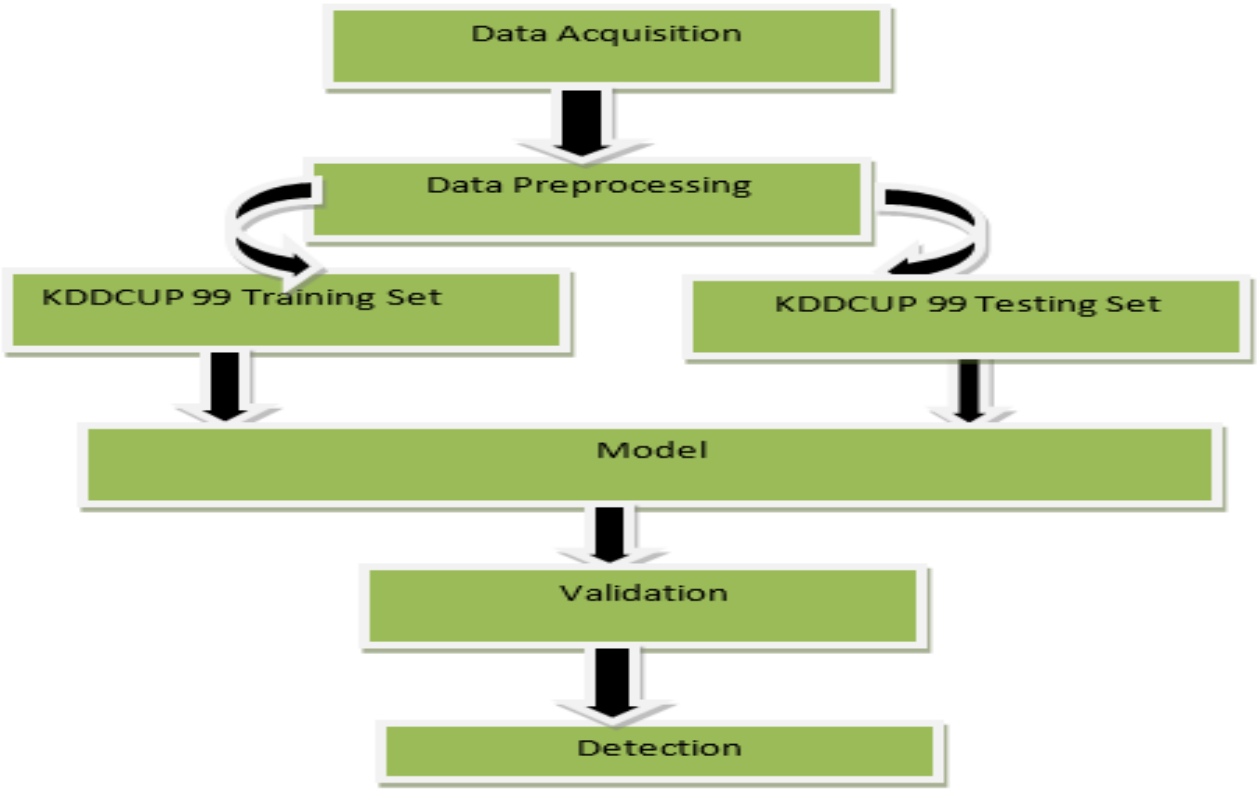
Related work

Author(s)	Techniques	Strengths	Limitations
Taher K. A, B. Yasin Jisan, M. and Md. M. Rahman, (2019),	ANN and SVM	The analysis of their result shows that the model built using ANN and wrapper feature selection achieved a detection accuracy rate of 94.02%.	No regularization technique applied
Wang W. et al (2018)	Naïve Bayes, KNN, SVM,RF,Multilayer perceptron NN, CNN	CNN achieved the better performance with an accuracy of 95%.	No regularization technique applied
Fahimeh. F. and Heikkonen, J., (2018).	Deep Autoencoder	The proposed approach achieved detection accuracy 94.71% on the KDDCUP99 test dataset.	Autoencoder learns better representations and its activations are more sparse which makes it perform better than original autoencoder when a regularization is

Related work

Author(s)	Techniques	Strengths	Limitations
Al-Qatf M., Lasheng, Y., Al-Habib, M. and Al-Sabahi, K., (2018),	KL divergence regularization technique	The proposed approach is used for feature learning and dimensionality reduction.	KL divergence regularization technique does not promote sparsity.
Wahyudi, M. (2018)	Combination of KL divergence and L2 regularization techniques.	L2 regularization technique work well with high-dimensional data where many features are correlated.	L2 regularization technique employed by has interpretative issues, that is L2 regularization does not promote sparsity

Methodology



Methodology cont…

Data Description

- We evaluated our proposed approach on the KDDCUP'99 dataset which is mostly widely used for the evaluation the intrusion detection system.
- We used original KDD-CUP'99 dataset which contains 494,021 samples for training the model in the training phase. It is common practice to use 10% of the original data as a training dataset since this dataset can represent the original KDD-CUP'99 data and allow for reduced computation [10].
- In the testing phase, the trained model is tested by using a test dataset contains 311029 samples with corrected labels.

Methodology cont…

Data Description

- We also categorize the attacks into the five (5) major traffics.
- - Normal
- - DoS= Denial of Service
- - R2L= Remote to Local
- - U2R= User to Remote
- - Probe

Table: 1. Mapping of attack class with attack type

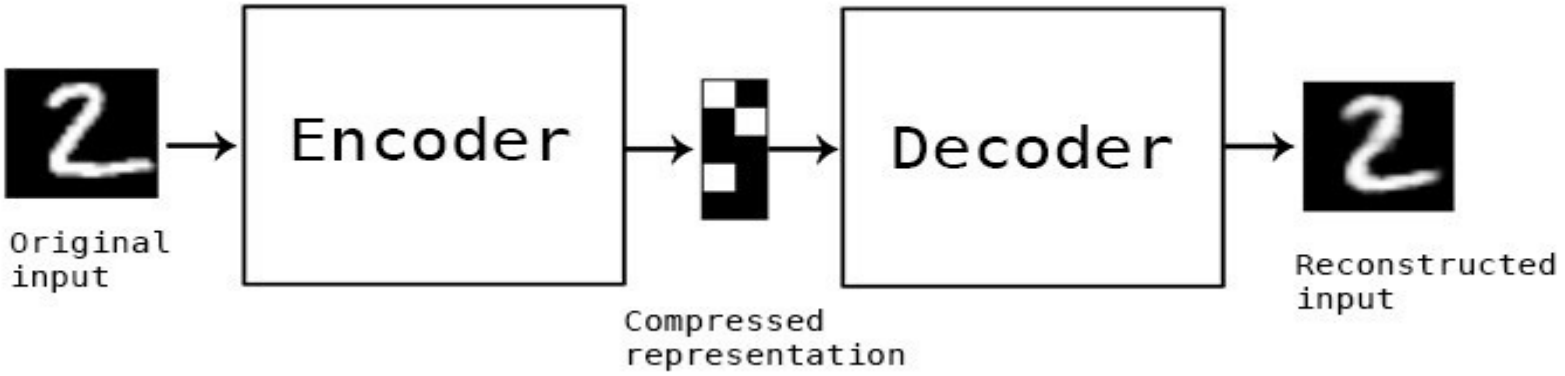
1	DoS	Back, Land, Neptune, Pod, Smurf, Teardrop, Apache2, Udpstorm, Processtable, Worm (10)
2	Probe	Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint (6)
3	R2L	Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named (16)
4	U2R	Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps (7)

Methodology cont…

- **Data Preprocessing**
- In preprocessing stage, we simply do the job of applying transformations to the data before feeding the model.
- The non-numerical attack types are converted into the numeric categorizes. In the binary classification, 1 and 0 are assigned to the normal and attack class by using binary coding, respectively. In the multi-classification, we used one hot encoding to convert five categorical classes into five binary classes, with only one active.
- Data preprocessing is a technique that is applied to convert the raw data into a clean dataset.

Methodology cont…

- **What is AutoEncoder?**
- Autoencoders are an important part of unsupervised learning models in the development of deep learning. While autoencoders aim to compress representations and preserve essential information for reconstructing input data, they are often used for dimensionality reduction or feature learning.



Methodology cont…

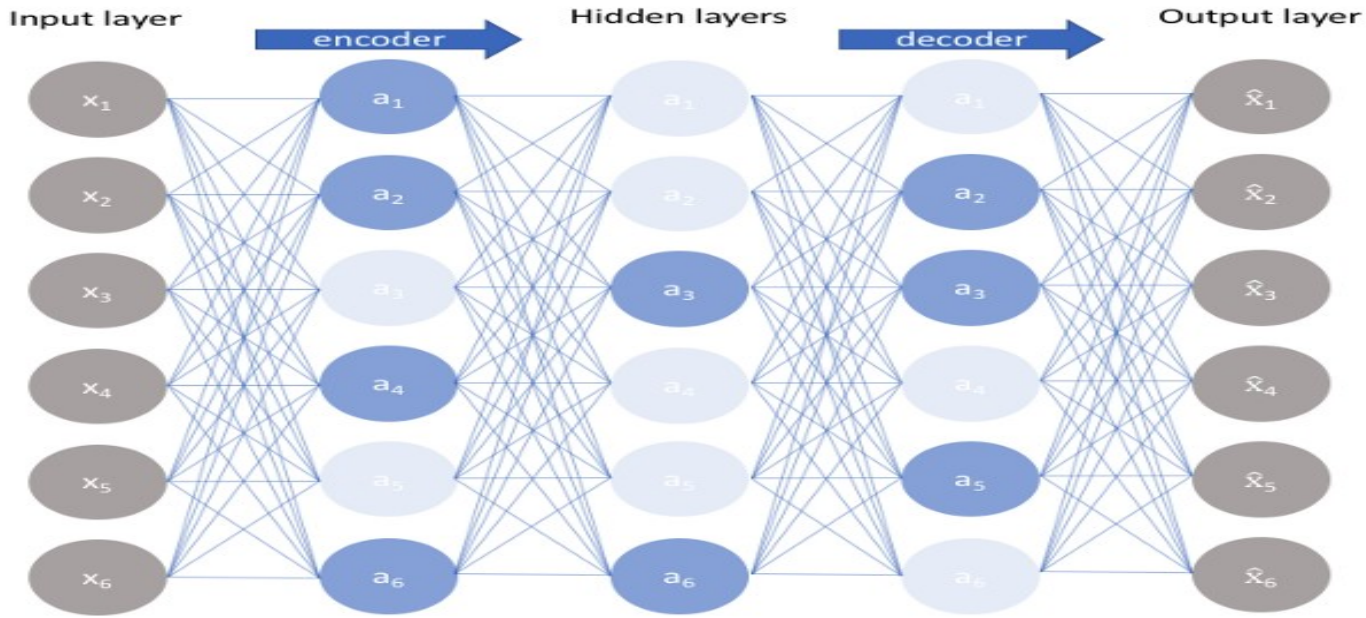
- **AutoEncoder**
- A basic autoencoder can be also simply regarded as neural networks, they are optimized using gradient descend based optimization methods.
- The difference between a basic autoencoder and neural networks is that autoencoder is composed of two symmetric parts: **encoder** and **decoder** with dimensions of compressed representations smaller than dimensions of the original input.

Methodology cont…

- **Sparse Autoencoders**
- A sparse autoencoder is simply an autoencoder whose training criterion involves a sparsity penalty. In most cases, we would construct our loss function by penalizing activations of hidden layers so that only a few nodes are encouraged to activate when a single sample is fed into the network.

Methodology cont…

- **Sparse Autoencoders**



- The intuition behind this method is that, fewer nodes activating while still keeping its performance would guarantee that the autoencoder is actually learning **latent representations** instead of redundant information in our input data.

Methodology cont…

- **Why L1 Regularization Sparse**
- L1 regularization and L2 regularization are widely used in machine learning and deep learning. L1 regularization adds “absolute value of magnitude” of coefficients as penalty term while L2 regularization adds “squared magnitude” of coefficient as a penalty term.
- To apply sparsity to our model, we used the L1 regularization technique which set weights to zero, and that will in turn lead to a lower loss value and more zeros weights produces a sparse model.
- We also used ReLu activation function that can actually allow the neurons to output a zero value that is, it penalizes the neurons as expected.
- Although L1 and L2 can both be used as regularization term, the key difference between them is that L1 regularization tends to shrink the penalty coefficient to zero while L2 regularization would move coefficients towards zero but they will never reach. Thus L1 regularization is often used as a method of feature extraction.

Methodology cont…

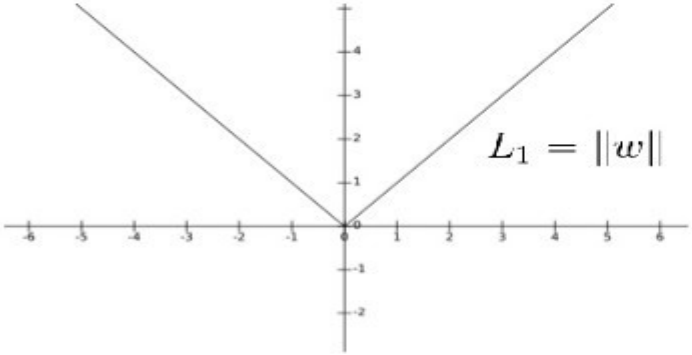
- **Why L1 Regularization Sparse**
- Consider that we have two loss functions L1 and L2 which represent L1 regularization and L2 regularization respectively.

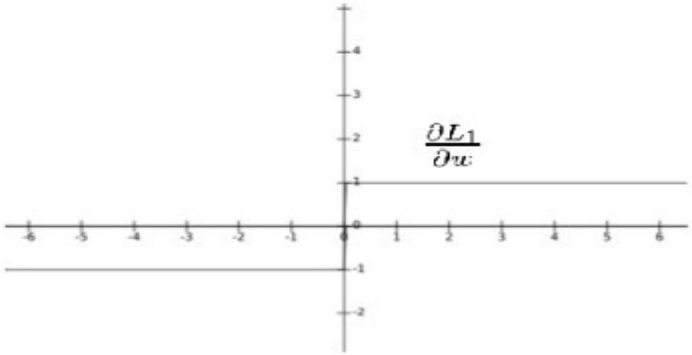
$$L_1 = \|w\| , L_2 = w^2$$

- Gradient descent is always used in optimizing neural networks. If we plot these two loss functions and their derivatives, it looks like this:

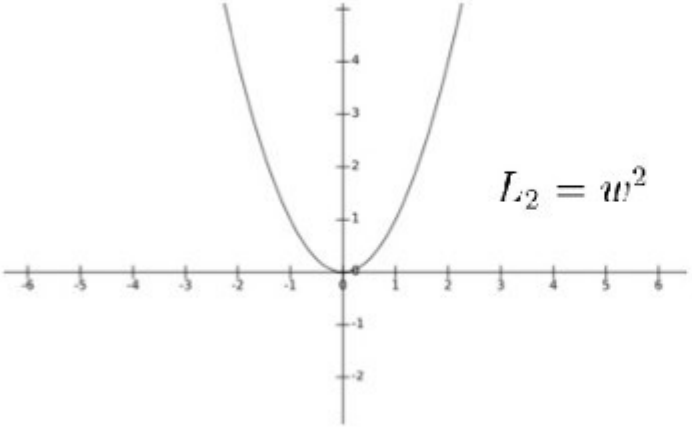
Methodology cont…

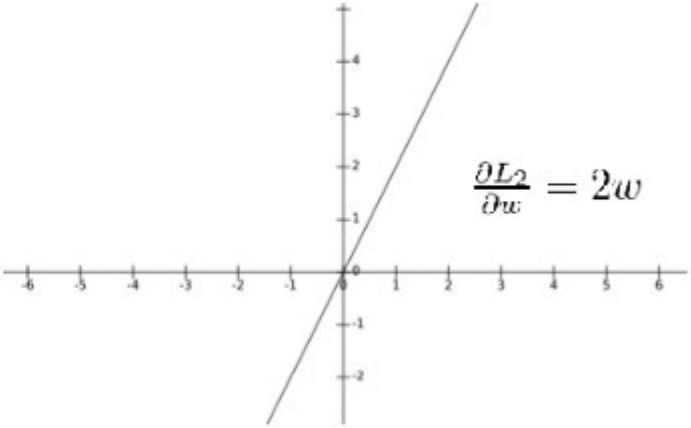
- **Why L1 Regularization Sparse**
- L1 regularization and its derivative





- L2 regularization and its derivative





Methodology cont…

- **Loss Function**
- Finally, after the previous analysis, we get the idea of using L1 regularization in sparse autoencoder and the loss function is as below:
- Loss function is the method of evaluating how well our algorithm models the dataset.
- *Obj* = Loss +$\lambda/2m$ * $\sum||W||$
- Where λ is the regularization parameter. *m* represents the number of samples from the dataset and *W* is the weight of the input data. Weight is the parameter that transforms the input data within the hidden layers of a neural network.
- The equation below shows the relationship between weight and other variables.
- y = Wx+b
- **x** is the input, **y** the output, **w** is the weight, and **b** is the bias.
- L1 regularization made the weight matrix of the layer's output y to return 0, due to the sparsity of L1 regularization, sparse autoencoder actually learns better representations and its activations are more sparse which makes it perform better than original autoencoder without L1 regularization.
- L1 impose sparsity in the neural network there by helping to overcome model overfitting.

Results

- **Binary Classification**
- The The binary classification model has an accuracy of 97% and the total training time was 46.7 seconds. The model hyperparamters include:
- Regularization technique: L1 Regularization
- Output layer's activation function: ReLU
- Epochs: 30
- Batch size: 100
- Drop out percentage: 50%
- Numbers of hidden layers: 1.

Results

- **Multiclass Classification**
- The accuracy of Multiclass Classification model is 96% on training and validation accuracy is 98%. This placed side by side with validation loss which is 0.0724 meant that our model is good.
- The model hyperparamters include:
- Regularization Technique: L1 regularization
- Output layer's Activation Function: ReLU
- Epochs: 20
- Batch size: 64
- Drop out percentage:10%
- Numbers of hidden layers: 3

Summary and Conclusion

- In our experiment to detect network intrusion, we have an accuracy of 96% which is higher than that of Al-Qatf M., Lasheng, Y., Al-Habib, M. and Al-Sabahi, K., (2018) and Wahyudi, M. (2018), because of the implementation of representational sparsity, linear behaviour of ReLU and the presence of L1 regularization technique in our experiment.
- We have found out that the use of Rectified linear activation function (ReLU) and L1 regularization technique increased the detection rate of our model more than previous works on Deep Sparse Autoencoders Network Intrusion Detection Systems.

Future Work

- Since L2 regularization technique works well with data of high dimension, we look forward to trying both L1 and L2 regularization techniques on a more recent network intrusion dataset of higher-dimensionality in our future work.

References

- [1] J. Fruhlinger (2020), What is a cyber attack? Recent examples show disturbing trends, https://www.csoonline.com/article/3237324/what-is-acyber-attack-recent-example-show-distrubing-trends.html.
- G. Gross, Intrusion Detection Techniques, methods and Best practices: Detecting Networking Intrusion in 2019. From https://alienvault.com/blogs/security-essentials/intrusion-detection-techquues-methods-best-practices, AT&A Cybersecurity-Security Essentials, updated on February 14, 2019.
- Taher K. A, B. M. Yasin Jisan and Md. M. Rahman, (2019), Network Intrusion Detection using Supervised Machine Learning Technique with Feature Selection, 2019 International Conference on Robotics
- [8] Al-Qatf , M., Lasheng, Y., Al-Habib, M. and Al-Sabahi, M., (2018), Deep Learning Approach Combining Sparse Autoencoder With SVM for Network Intrusion Detection, School of Information Science and Engineering, Central South University, Changsha 410083, China.
- [9] Wahyudi, M. (2018), Implementation and Analysis of Combined Machine Learning Method for Intrusion Detection System, International Journal of Communication Networks and Information Security (IJCNIS), Vol. 10, No. 2.
- [10] C.Versloot, (2020), What are L1, L2 and Elastic Net Regularization in neural networks? https://www.machinecurve.com/index.php/2020/01/21/what-are-l1-l2-and-elastic-net-regularization-in-neural-networks/.

THANK YOU
FOR
YOUR ATTENTION