A comparison of neural network-based network IDS techniques for achieving interpretability

Sam Denton

Science and Technology Department, Bournemouth University

ABSTRACT

Deep learning and neural network techniques have proven to be an effective solution for network intrusion detection, and look set to dominate the field of cyber security. However, these technologies suffer a major drawback, their black box nature. The inability to interpret, explain and understand how neural networks reach classification results hinders our ability to trust and rely on them for important tasks such as intrusion detection, especially when applied to critical infrastructure. Lack of understanding makes it hard to be sure that deep learning and neural network techniques will transition reliably from controlled testing environments to working situations, were the results from their classifications will mean the difference between an attack being detected or allowed into a network, with potentially catastrophic consequences.

In this paper I will examine some of the current 'state of the art' techniques which can improve on interpretability and explainability when applied to network intrusion detection systems, looking at how they affect the systems accuracy and false positive rates, as well as giving detailed explanations as to how they aid our understanding of the systems.

1 INTRODUCTION

1.1 PROBLEM DEFINITION

Since the first neural networks were proposed in 1943, it has been apparent that, while this technology has the potential to solve problems beyond the scope of a human brain and standard computation algorithms, we struggle to explain and interpret how the network reached the solution. This is an issue because many of the problems these networks are tackling could have significant impact on the human population, and to be able to trust the results, we need to understand why they are being made. Being able to understand the results also allows us to correct mistakes or misclassifications and build more powerful and reliable neural networks. If the neural network could explain its classifications and hence detections in comprehensible natural language, we would be able to quickly understand what caused the detection, and if additional action is required. This is invaluable, both to generate greater levels of trust in the system, but also identify false positives, of which IDSs tend to show many. A quote from the paper 'Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)' (Adadi, A. and Berrada, M. 2018) talks about the limitations of using black-box systems, despite their promise:

> 'even with such unprecedented advancements, a key impediment to the use of AI-based systems is that they often lack transparency. Indeed, the black-box nature of these systems allows powerful predictions, but it cannot be directly explained'.

This sums up well the current state of almost all systems utilising machine learning techniques like neural networks and shows the need for greater levels of interpretability in our future systems.

Because of the desire to understand neural networks, the field of explainable artificial intelligence (interpretable AI, XAI) is almost as old as AI itself. Not much progress was made in the field until the last 10 years, but it is becoming apparent that for AI to progress further into modern society, we must learn to trust it, and the first step to trust is understanding. Because of this, there has been a big push for XAI and many techniques have been developed. The Defence Advanced Research Projects Agency (DARPA) has expressed great interest in the field, and in 2016 created a program specifically for aiding and funding the progression of XAI. Quoting their official website:

> "the effectiveness of these systems is limited by the machine's current inability to explain their decisions and actions to human users. Explainable AI—especially explainable machine learning—will be essential to understand, appropriately trust, and effectively manage an emerging generation of artificially intelligent

machine partners" (Gunning, D. 2016).

1.2 IMPACT

This problem is especially relevant when neural networks are applied to network intrusion detection systems, and even more so when we wish to apply them to critical infrastructure. These systems are the backbone for many important aspects of modern civilisation such as banking, communication, medical care and many others and as such it is imperative that we understand and trust the systems protecting them (Amarasinghe, K. and Manic, M. 2018) (Amarasinghe, K., Kenney, K. and Manic, M. 2018).

Regardless of how accurate and robust the systems become, there will always be edge cases which have not been tested against. If an attack contained data which the network had never seen before, we have no way of ensuring it would react in a suitable manner, since we don't understand how it evaluates the data. This could have catastrophic affects as an attack on critical infrastructure which is not detected could cause irreparable damage, and potentially be life threatening. Less dangerous, but maybe more likely is the opposite. If normal but unusual data is shown as an attack, time and money could be wasted trying to prevent something which does not exist.

1.3 AIMS:

To research the 'state of the art' in solutions for the black box issue within Artificial Intelligence driven Network Intrusion Detection Systems, determine their suitability and recommend the most fitting solutions from current proposed designs, with potential improvements.

1.4 OBJECTIVES:

1. Explore current proposed solutions to the black box problem in neural network-based network intrusion detection systems

2. Determine the positives and negatives of each method, pertaining to how they affect detection rates and how well they fit my definitions for interpretability and explainability

3. Evaluate solutions to determine best fit to solve the problem, give recommendation for which solution is best and provide insight into potential improvement

1.5 EXPLAINABILITY VS INTERPRETABILITY

To fully appreciate the differences between solutions examined in this paper, as well as what I am looking for in an ideal solution, it must first be clear what the definitions of interpretability and explainability are, and the differences between them (Choudhury, A. 2019) (Došilović, F., Brčić, M. and Hlupić, N. 2018) (Gall, R. 2018).

1.5.1 Interpretability

Interpretability refers to the extent to which we can describe the cause effect relationship between the input data and output result. It has also been described as our ability to predict the output of the neural network based on changes to the input.

1.5.2 Explainability

Explainability expands on this by showing the extent to which we understand the internal mechanics of a neural network and can describe the networks reasoning behind classifications. This does not simply mean showing the layout of nodes and connections but means being able to extrapolate deeper meaning and learned knowledge from the network.

2 PROPOSED METHODS

Here I will present the five methods I have examined, with details about how they improve interpretability in neural networks.

2.1.1 Self-Organising Maps (SOM)

The first method I will examine is best detailed in the paper 'Intrusion Detection System Using Self-Organising Maps' (Alsulaiman, M., Alyahya, A., Alkharboush, R. and Alghafis, N. 2009). This paper demonstrates a NN-IDS system built using self-organising maps, utilising an architecture type called a Kohonen network, where one layer of the network is 2 dimensional rather than one. This idea was developed as a way to better emulate natural brains, allowing differed areas in the network to specialise to different features.
A SOM is a fully connected architecture, although some variations can include convolutional layers (Dozono, H., Niina, G. and Araki, S. 2016), especially when used for pattern recognition. The main way a SOM differs from a feed forward neural network is in the training algorithm used to alter connection weights. Instead of updating the entire network for each iteration of data, a SOM will select a single neuron which is closest to the current input data feature space. When this neuron is updated to closer match the target output it will also alter its neighbouring neurons by a percentage of the amount it is altered by, depending on their distance from the selected neuron. In this way, features of the input data are seen to "compete" for representation on the network. This results in a neural network where different areas of neurons represent different aspects of the data feature space and can be transcribed into a 2D map representing higher dimensions of features, an example of which could be this:

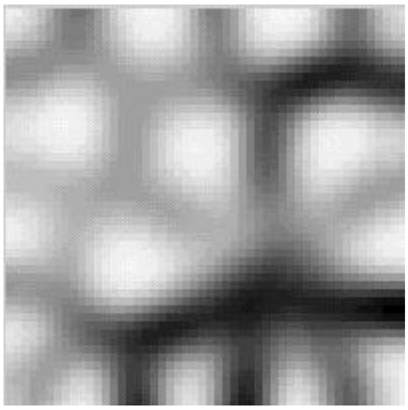

Figure 1: Example output from SOM (Ahn, J. and Syn, S. 2005)

This visual representation of the neural network shows clusters of neurons which have similar neighbours (in white) and neurons which are very different from their neighbours (in black). Each of these clusters represents a different feature or set of features from the data which are considered when making a classification and can aid in understanding how the network breaks down the feature space. Another neural network is then trained to use the two-dimensional representation of the data to make the classification, although this is often integrated directly into the SOM, and the visual representation is extracted from the SOM layer, while the classifier part of the network looks directly at the weights and distances within the SOM layer. This method was designed as a way to represent data with higher dimensional feature spaces in a 2D map, but it has the advantage of showing a visual representation of its structure which has been argued as a step towards interpretability (Fortuin, V., Hüser, M., Locatello, F., Strathmann, H. and Rätsch, G. 2019).

This method is most useful in aiding interpretability if paired with another method like an adversarial pair or layer-wise relevance propagation which can show exactly which input features are considered during a classification as it makes the results of these methods easier to read by a human.

2.1.2 Neuro-Fuzzy Classifying

The second method is neuro-fuzzy classifying. This paper (NadjaranToosi, A. (2007) shows how a fuzzy rule system can be applied to intrusion detection to aid in reduction of false positives, however this method of structuring a neural network has potential to aid interpretability (Fernandez, A., Herrera, F., Cordon, O., Jose del Jesus, M. and Marcelloni, F. 2019). Neuro-fuzzy clustering is a method for forcing a neural network to classify features of a dataset using natural language if-then statements. For example, when examining an image to determine if it contains a cat, it may generate if-then statements like "if these pixels contains this colour" then alter the classification in some way. If the statement is satisfied, then the probability increases that the image contains a cat. The neural network is tasked with deciding what features of the input data satisfy the statements, and to what degree. In this method a human must give the possible classifications beforehand. This is considered a fuzzy logic system as the neural network generates a "score" for how well the input data satisfies each statement, rather than outputting a binary 'yes' or 'no' result, emulating the way a human brain uses uncertain probabilistic values when making a decision. This method has the advantage of using natural language rules when making classifications, meaning a human is able to read and change the rules, improving interpretability and allowing improvements to be made.

2.1.3 Neuro-Fuzzy Clustering

The next method is similar to neuro-fuzzy classifying, but instead uses fuzzy clustering. The principals are the same, however the network is not given specific classification labels to sort into. Instead it creates its own unlabelled clusters based on patterns it finds in the data. Despite the end results being similar, the subtle difference means that this method is trained using unsupervised learning, where Neuro-Fuzzy Classification uses supervised training. These different approaches mean the two methods will find different patterns and will result in a completely different network structure. The paper 'Intrusion Detection using Fuzzy Clustering and Artificial Neural

Network' (Surana, S. 2014) shows how this can be used in intrusion detection, but the principals which make it useful for interpretability are the same as fuzzy classifying. I have included it as a separate solution due to the argument that not defining the classifications before-hand gives the network greater freedom to learn patterns, improving its depth of knowledge and understanding of context, which can aid in generalisation of the network. This can be very useful for explainability if this method was progressed further.

2.1.4 Adversarial Pair of Neural Networks

The fourth method uses an adversarial pair of networks. As shown in the paper 'An Adversarial Approach for Explainable AI in Intrusion Detection Systems' (Marino, D., Wickramasinghe, C. and Manic, M. 2018) a second network can be trained alongside a NN-IDS which is tasked with explaining the classifications of the first network (Tomsett, R. 2018). This method has been used previously for penetration testing NN-IDSs and involves training the second network to attempt to figure out the smallest degree of change needed to the least number of input features, to alter the results of the classification. Previously, this was then used to train the IDS to improve its accuracy and prevent attackers doing the same, however this work used the adversary to gain an understanding of what the NN-IDS considers when making the classification. The paper found that by having the adversarial network focus on misclassified data they could generate natural language statements for what went wrong and show the reasons for the mistake. As an example, the paper shows the generated reasons the NN-IDS misclassified some normal data as a DOS attack:

"Normal samples were mis-classified as DOS because:

- high number of connections to the same host (count) and to the same destination address (dst host count)
- low connection duration (duration)
- low number of operations performed as root in the connection (num root)
- low percentage of samples have successfully logged in (logged in, is guest login)
- high percentage of connections originated from the same source port (dst host same src port rate)
- low percentage of connections directed to different services (diff srv rate)
- low number of connections were directed to the same destination port (dst host srv count)"

Examining these reasons shows why the data was misclassified and gives an understanding of how we can alter the NN-IDS to better classify this data in the future. This technique can be used to both improve the accuracy of the network, and gain a better understanding of classifications, and can be used to help justify any decisions made based on detection.

2.1.5 Layer-Wise Relevance Propagation

The final method I have examined is Layer-Wise Relevance Propagation. It is a technique which has been developed to help generate values showing the relevance of different input features when making a classification. As shown in this paper ‘Improving User Trust on Deep Neural Networks Based Intrusion Detection Systems’ (Amarasinghe, K. and Manic, M. 2018) this method can be used to great success in IDS systems to help improve interpretability. The result is achieved by manually examining the weights in the network through back propagation and developing a ‘heat map’ showing which nodes contributed most to the classification, continuing backwards until you reach the input nodes. There are algorithms to automate the process for larger networks, but they have issues with progressively larger contribution values scaling indefinitely. The advantage of this method is that it can be applied to any neural network architecture to generate basic explanations as to which features were important to the classification, and to what degree.

3 REQUIREMENTS

For a solution to be considered a success there are a number of criteria it can be checked against.

Must

- The solution must be as accurate as current solutions or show the ability to become as accurate with further development, with a benchmark of 97%
- It must have a false positive rate as low as current solutions or show the ability to reduce the false positive rate with further development, achieving below 3%
- It must have the ability to show which input data features are considered when making a classification.
- It must be able to scale to a network of any size or type.
- It must produce a result which aids in a humans ability to predict the classification based on the input values

Should

- It should be able to give a rating for how important each input data feature was when making a classification.
- It should be able to give natural language outputs (or equivalent such as visual representation) when describing features of the data.
- It should be able to give descriptions of how changes to the input data would affect the classification, which can aid in determining how to solve the issue.
- It should be able to give examples for input values which would result in a specific classification

Could

- It could have the ability to produce a description or model of ranges of input values which would result in different classifications
- It could have an intuitive user interface allowing for easy monitoring of the system and be able to display detections in a comprehensive manor which does not require security expertise to action upon.
- It could have the ability to be questioned by a user and generate natural language responses to gain further information about the classification.

Would

- It Would be able to generate potential solutions to detections and action upon those solutions automatically, making it an intrusion prevention system.

4 RESEARCH FINDINGS AND EVALUATION

Table 1: A comparison of each method against the requirements

	Must	**Should**	**Could**	**Would**
Self-organising maps	SOM's satisfy three of the five **Must** requirements, as it has been shown to outperform standard feed forward networks in both accuracy and false positive rates and can scale to any network size if trained correctly, however it does not make it clear which input features are used in a classification unless paired with another of the examined methods. It also provides no aid to a human attempting to predict the results of the classification.	This method only satisfies one of the four **Should** requirements, as it does produce a visual representation of the data, however it is unable to give a rating for the importance of different features during a classification, nor can it describe how changes to the input data would affect classifications or give examples of which inputs would lead to a specific classification.	Currently none of the **Could** requirements are fulfilled, as a SOM is unable to generate responses to questions from a human or give a range of input values which would cause a specific classification, however it could be integrated into a GUI which would aid simplicity.	This method does not satisfy the **Would** requirement, however it could aid a future method in understanding the features of the input data, improving its utility and accuracy as an intrusion prevention system.
Neuro-Fuzzy Clustering / Classification	Neuro-Fuzzy Classifying and Neuro-Fuzzy Clustering meet four of the five **Must** requirements; it has been shown to be as accurate as standard neural network IDSs even in early testing and shows improvement to false positive rates; it can scale to a network of any size, although it can take longer to train on large networks; it gives a clear natural language representation of which input features are considered for each classification; however, while it can be used to better predict the output this information is difficult to extrapolate and requires manual work to evaluat if	This method satisfies three of the four **Should** requirements as it gives a numerical value when scoring the contribution of different input features. The natural language if-then statements make it easy to understand the reasons for the classification and also make it clear how a classification would change if the input data was altered, though it would be difficult to manually work out for input data with greater dimensionality. However, it is unable to provide examples of inputs which would give specific classifications.	Currently none of the **Could** requirements are satisfied. It is unable to generate responses to questions from a human. Information about detections could be extrapolated from the if-then statements and displayed on a GUI, but this would require further work. It would be possible to work out ranges of results which would cause specific classifications, however this would require a lot of additional work.	This method currently has no preventative systems, meaning it does not meet the **Would** requirement, however information extrapolated from the if-then statements would make deciding on preventative measures easier for an IPS.

	the if-then statements are satisfied.			
Adversarial Pair of Networks	This method meets all five of the **Must** requirements. As it can be applied to any other neural network IDS, it is as accurate as the current best system with as low false of a positive rate and can be used to gain knowledge on how to improve those systems. It scales to a larger network as well as the system it is applied to. It gives clear natural language statements showing exactly which input features contributed to a classification and why. It also gives a clear picture of how inputs correlate to outputs, making it easier to predict the output classification.	This method meets two of the four **Should** requirements. It gives the best explanations for classifications with in-depth descriptions of the input data and shows how the classification would change if the input data was altered. It does not currently show the importance of each input feature for each classification type however this information could be extrapolated manually, or automatically in future works. This method could also be adapted to allow the output of example input data to create a specific output, but this is not currently implemented.	Currently none of the **Could** requirements are satisfied. This method could be adapted to allow for questioning the network with simple queries about detection, by having it manipulate the flagged input data and seeing the result, but this is not currently implemented. The natural language statements would make a GUI far more understandable, with a clear description of each detection, allowing an expert to quickly decide how to react, however this method is still in early testing and is not currently implemented into a finished product. The adversarial network could be altered to test different possibilities of input data to determine a model of how different ranges of inputs can lead to specific outputs, but this is not yet implemented.	This method currently has no preventative systems, meaning it does not meet the **Would** requirement, however information extrapolated from the statements would make deciding on preventative measures easier for an IPS.

Layer-Wise Relevance Propagation	This method meets three of the five **Must** requirements. As it can be applied to any other neural network IDS it is as accurate as the current best system with as low false positive. It gives a clear picture of which input features are important for a classification, with the added advantage of showing hidden layer nodes contributions as well. It partially aids in the ability to predict outputs, but this would require a lot of work tracing how values would change as they propagate through the network. While it can scale to a network of any size and feature dimensionality, it will slow considerably with current algorithms.	This method satisfies one of the four **Should** requirements. It clearly shows the degree of contribution from each input feature; however, it is unable to give any explanations for how these input features are relevant or how changing the results might change the classification. While it can give a rough idea as to which input values could result in a specific classification, it is not detailed or accurate enough to be of much use.	This method meets none of the **Could** requirements. It is unable to generate any explanations for input features or reasons for desertions made by the network. It provides little use for aiding a GUI and can only provide a general idea as to how different inputs would propagate through the network, and not any specific values or ranges.	Layer-wise relevance propagation does not meet the **Would** requirement and provides little aid for a future IPS.

4.1 Results

Table 2: The number of requirements from each prioritisation of MoSCoW assessment satisfied by each solution

	Number of Requirements Satisfied			
Method	**Must**	**Should**	**Could**	**Would**
Self-Organising maps	3	1	0	0
Neuro-Fuzzy Classification	4	3	0	0
Neuro-Fuzzy Clustering	4	3	0	0
Adversarial Pair of Networks	5	2	0	0
Layer-Wise Relevance Propagation	3	1	0	0

The data in table 2 gives a good representation of how each solution compares when assessed against my requirements for interpretability. It shows that while each method has some positives, the methods most accurately matching my criteria is the method utilising an adversarial network trained to give explanations for the first's classifications.

4.2 ANALYSIS

This research has shown that each of these solutions presents valid methods for intrusion detection with accuracy comparable to currently used systems, and while there is not yet a good solution for explainability in this field, some aspects of these methods have made progress towards interpretability. This shows the black-box problem to be unsolved, and still requires significant work in future to achieve an acceptable solution.

The best method I have analysed for aiding interpretability in IDSs is using an adversarial network. The simple natural language statements produced based on why each individual data was classified the way it was is a great improvement over any method showing only which input features contributed to different classification types and provides clear insight into the cause effect relationship between input and output. The statements would make it clear if an attack was a genuine threat or false positive to any experienced user and give a good idea how the attack could be countered.

The method utilising layer-wise relevance propagation had the advantage of showing the internal structure of the network, however it fell short in its ability to give explanations for how the data was relevant to the classification. It was only able to produce a model for how data propagated through the network for a specific classification type, rather than showing this for each data. It required additional manual work to achieve its results, although automated solutions are possible.

The methods Neuro-Fuzzy Classification and Clustering were both able to produce information which aided the prediction of results by a human, however this information had to be manually extrapolated from the if-then statements to be useful. This information was almost as good as the statements from an adversarial network, however it lacked the ability to show this for each data and could only show what inputs were contributing to each classification type.

Finally, the visual representation of the input data generated by the self-organising maps gave a useful insight into the patterns and structure of the data, with clear demonstration of how input features were related. While this is useful for a classifier and could potentially make a network more accurate, it does not give a lot of information aiding interpretability. It is useful for understanding the data and can help a human to appreciate how a neural network might view input features, however it does not contribute to solving the black-box issue.

From this work, I conclude that the best method for improving interpretability in a neural network-based network intrusion detection system is to use a combination of three of the proposed methods I have shown. Including a self-organising map or kohonen layer to reduce the dimensionality and complexity of the input data can help with accuracy and pattern recognition as well as be a useful aid for a human to extrapolate information from. Training a second network as an adversary designed to give explanations for each classification based on how the network reacts to changes in the input data generates natural language statements which greatly aid a human's ability to predict the results and shows the relationship between input output correlation and is the best single method currently for improving interpretability. Additionally, using layer-wise relevance propagation to analyse the internal structure of the network can give even greater understanding to how the network classifies data, and helps to alleviate some of the unknown factors of the structure.

5 CONCLUSIONS

In this paper I have shown a number of methods which strive for a greater level of interpretability and explainability in the field of neural network-driven network intrusion detection systems. While none of the proposed methods fully encompass the requirements I set, some were successful in part. From this work I have concluded that an adversarial approach is most suitable for generating reliable explanations for classifications made by an NN-IDS. The ability to be applied to any neural network type is advantageous and allows it to stay on par with the current best systems, and its natural language statements showing exactly which input features contributed to each classification were easily understandable and would give useful insight to a human when a potential detection is made. Based on my definitions I believe this method qualifies as interpretable but falls short on being explainable. The adversary can consistently give insights into which input features led to specific classifications but is unable to explain the logic and reasoning behind them.

From this I conclude that the best solution for creating an interpretable intrusion detection system utilising neural networks is to use a combination of an adversarial network trained to test the reasons for each detection, include a self-organising map or kohonen layer to simplify input data features into a 2 dimensional space, aiding both pattern recognition and accuracy, and giving a human the ability to view features of the data in a better form, and to analyse the network with layer-wise relevance propagation to give a better understanding of the internal structure of the network.

In summary, this work has shown that a great deal more research and development is needed in this field before the black-box issue can be solved, and until it is, we will have to be sceptical of any results produced by a neural network. This does not diminish the usefulness of the technology; however, it does hinder its progress and adoption into modern society and can add risk to using it in intrusion detection, especially applied to critical infrastructure. Without a greater level of understanding as to the deeper knowledge gained by the network, we will be unable to predict how it will react to all situations. If an IDS built upon these technologies was to receive data in a form it has not trained with, or been tested against, it could react in a way which causes harm to the system, either by allowing an attack onto the system or by recommending defensive measures for normal data.