A-GHSOM: Adaptive Growing Hierarchical Self Organizing Map for Network Intrusion Detection

Dennis Ippoliti and *Xiaobo Zhou*
University of Colorado at Colorado Springs, USA

ICCCN 2010, ETH, ZURICH

1

INTRODUCTION

- Anomaly detection and misuse detection are two major types of network intrusion detection systems.
- Machine learning approaches have been used for anomaly detection. In particular, approaches based on self-organizing maps (SOMs) of artificial neural networks have shown effectiveness at identifying “unknown” attacks.
- Effectiveness of using traditional SOM models is limited by the static nature of the model architecture. The size and dimensionality of the SOM model is fixed prior to the training process and is determined by trial and error.
- GHSOM is an SOM model that does not use a predetermined map topology. Instead, the size and the dimensionality of the map dynamically grow during the training process to optimally fit the training set based on user defined parameters.

2

GHSOM: PROS AND CONS

- Advantages: Trial and error are eliminated from the training process. An ideal topology is formed unsupervised based on the training data. Additionally, hierarchal relationships in the training data are discovered and modeled in the final configuration.
- Disadvantages: Does not account for concept drift. That is, although the topology is modeled to fit the training set, it is not adapted online to account for changes to live data that occur over time. There is no ability to adapt the model as live data is processed. Over time, subtle changes in legitimate traffic patterns as well as vulnerability to previously "unknown attacks" require updates to maintain accuracy.

3

A-GHSOM: FOUR ENHANCEMENTS

- We propose an adaptive growing hierarchical self organizing map (A-GHSOM) that adapts online to changes in the input data over time.
- Four significant enhancements.
  - **Threshold based training process:** The training process expands the GHSOM model to fit the training set. Instead of using the mean quantization error as a control parameter, we establish a new parameter of threshold error value that is more suitable to the network intrusion detection problem.
  - **Dynamic input normalization process:** monitors the range of observed values of input connections and uses the information to adapt the map scale during normalization online.
  - **Feedback-based threshold adaptation:** Quantization error is a measure of how closely an evaluated connection fits its matched neuron in the map. We use an error threshold that adapts over time to identify new attacks that may be initially matched to "normal" nodes in a GHSOM but are indeed malicious.
  - **Confidence filtering and forwarding:** Identify traffic patterns that are beyond the ability of a content oblivious system to evaluate. These connections can be filtered or forwarded to a content aware intrusion detection system for further evaluation.

4

WHY STILL USING KDD'99

- Many people argue to use live data for network intrusion detection as the KDD'99 dataset has some flaws.
- We conducted experiments using KDD'99 dataset for performance comparison of A-GHSOM with related approaches.
  - Major publicly available dataset.
  - Recent studies in [6], [10], [9], [11], [17] use the KDD'99 dataset for network intrusion detection. For performance comparison, the dataset is used.
- We are constructing live dataset. We will report new findings in future work.

5

A-GHSOM MAJOR BENEFIT

- Our baseline A-GHSOM approach achieves 92.21% accuracy and 0.58% false positive.
- Applying those four significant enhancements, the accuracy of the integrated A-GHSOM approach increases to 99.63% while false positive rate is just slightly increased to 1.8%.
- It is able to identify a subset of connections that are virtually indistinguishable based on the data available in the KDD'99 dataset.
  - Increases the accuracy on detecting previously "unknown" attacks to 94.04%
  - Maintained 1.8% false positive rate. When examining individual attack categories of Denial of Service (DoS), Probe, Remote to Local (R2L), and User to Root (U2R) it achieves accuracy of 99.82%, 99.58%, 92.66% and 87.14% respectively.
- Compared to eight representative intrusion detection approaches, A-GHSOM significantly increases the accuracy with low false positive rates.

6

SOM RELATED WORK

- Approaches based on SOMs have shown effectiveness at identifying both known and "unknown" attacks [Jiang et al., ICIAS 2009]
- The approach in [Sarasamma et al., IEEE Trans on Cybernetics] used a hierarchy network built on an SOM architecture with no neighborhood or transfer functions. Experiments were conducted with a variety of random training sets. Five training sets were used, each one consisting of a different distribution of connection classes. Each layer was trained on individual and exclusive feature sets.
- The work in [Kayacik et al., AI 2007] examined several approaches related to the application of SOMs to intrusion detection. Experiments were conducted with three different partitions of the training data, the entire 10 percent set, normal only connections, and a filtered set consisting of equal numbers of attack vs. normal connections. It also compared the effectiveness of using only six basic features to using all 41 features available in the sample data.
- Major issue: The size and dimensionality of the model is fixed prior to the training process.

7

INTRODUCTION TO GHSOM

- The GHSOM is a model with a hierarchical structure composed of independent growing SOMs. The size and the dimensionality of the map architecture are determined during the training phase.
- The initial map size is very small, usually a single layer 2 x 2. During the training process, the map grows both vertically and horizontally until the training process is complete.
- Two configurable parameters are used, δ_E *and* δ_D. δ_E *represents* the target quantization error for the map, and δ_D *represents the* maximum dimensionality of a single layer.
- After each training iteration, the deviation of the input data (quantization error) is computed. The map grows horizontally by adding rows and columns to the map to reduce quantization error. It grows vertically by adding child layers to parent layers that exceed the maximum dimensionality specified by δ_D. *The process* continues until the quantization error of the map is less than δ_E.

8

A GHSOM ARCHITECTURE

- At the end of the training process, each layer and sub-layer can have a different number of maps and sub-maps with varying dimensionality.

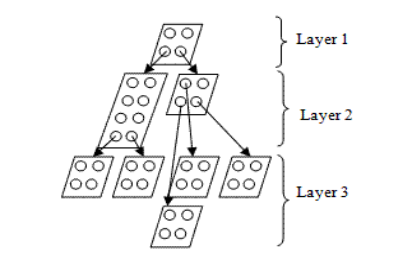


Fig. 1. A GHSOM architecture after off-line training.

9

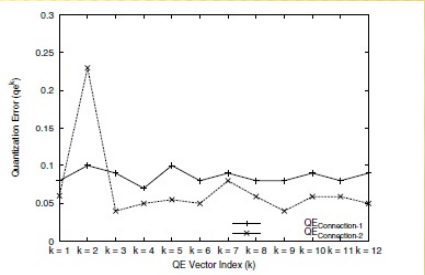
GHSOM FOR INTRUSION DETECTION

- GHSOMs has been used in network intrusion detection [Palomo et al., ICANN 2008].The map grows until the mean quantization error of the map is less than a predetermined training parameter. A method for calculating the quantization error based on both numeric and symbolic data was proposed. The map is only adapted during the training phase. Once training is complete and live data is applied, no adaptation or new learning occurs.
- Our Adaptive GHSOM improves upon existing efforts in four important enhancements.

10

I: THRESHOLD-BASED TRAINING PROCESS

- Instead of using the mean quantization error as a control parameter, we establish a new parameter of threshold error value that is more suitable to the network intrusion detection problem.
- Rationale: the input patterns of an anomalous connection often matches the input pattern of a normal connection very closely except for one or two parameters.
- Example: consider three vectors representing two connections and one weight vector
  - Using the mean quantization error or Euclidian distance, connection-1 has the higher mean quantization error while obviously connecton-2 has an anomalous value in the vector.



. Quantization error vectors of two sample connections.

11

I: THRESHOLD-BASED TRAINING PROCESS

- For effective anomaly detection, we are interested in not only the aggregate small discrepancies, but also identifying large single discrepancies.
- The threshold error value is calculated using the quantization error vector as follows:

$$TEV_j = \sum_{k=0}^{n} f(k)$$

$$f(k) = \begin{cases} qe_j^k & \text{if } k > \tau_1 \\ 0 & \text{if } k \leq \tau_1 \end{cases}$$

where $qe_j^k = |w_j^k - v_j^k|, k = 1, 2, ...n.$

- The node with the highest threshold error value is considered the *highest error node*. It is used for hierarchical map growing.
  - Please refer to the paper for more technical details.

12

II: DYNAMIC INPUT NORMALIZATION

- The dynamic input normalization process is designed to emphasize the true difference between individual input vectors. The idea is to normalize input patterns to values between 0 and 1 based on the system's scale values.
- To capture relative distance between two different values of the same feature, we adjust the input pattern so that all values are in the range of 0 to 1 using simple linear scaling for the normalization. We consider the "distance" to be how different these two values are from one another. The smaller the distance, the more similar the values are considered.

Minimum	Maximum	Value-1	Value-2	Normalized Distance
1	16	8	14	0.4
1	32	8	14	0.194
1	64	8	14	0.095
1	128	8	14	0.047
1	256	8	14	0.023

TABLE I
EFFECT OF SCALE ON NORMALIZED DISTANCE.

13

II: DYNAMIC INPUT NORMALIZATION

- By dynamically updating the scale of the input data, we are able to highlight the true distance between different values of the same feature. This is particularly useful when identifying "anomalous" behavior.
- To capture relative distance between two different values of the same feature, we adjust the input pattern so that all values are in the range of 0 to 1 using simple linear scaling for the normalization.
- The normalization function *F(x)* is given as:

$$F(x) = \begin{cases} \frac{x - min(t)}{max(t) - min(t)} & max(t) \neq min(t) \\ 0 & max(t) = min(t) \end{cases}$$

where *x* is a data point in the input pattern, *min(t)* and *max(t)* are the expected minimum and maximum values for the data point at time *t*, respectively.

14

II: DYNAMIC INPUT NORMALIZATION

- In an intrusion detection problem domain, there are several data points that the scale will be known ahead of time. Some points are in the scale of 0~*1, while some others are in 0~255.*
- In the KDD'99 dataset, 17 of the 42 data points have an undefined scale. Furthermore, the scale in the training data is not the same as the scale in the live data.
- We use an adaptive input normalization approach that automatically tunes the scaling parameters and weight vectors online based on the observed minimum and maximum values.
- Please refer to the paper for more technical details.

15

III: FEEDBACK-BASED THRESHOLD ADAPTATION

- A-GHSOM is further enhanced by use of feedback-based quantization error threshold adaptation. It adaptively adjusts thresholds for each node as input patterns are applied and add new nodes when appropriate.
- Each node is assigned two initial threshold parameters. *τ1 is used to calculate the threshold error* value. *τ2 is used as an upper limit on the acceptable total* quantization error and used to calculate the quantization error boundary (QEB) for a selected node as follows:

$$QEB_j = \begin{cases} 0 & tqe_j < \tau_2 \\ 1 & tqe_j >= \tau_2 \end{cases}$$

where $tqe_j = \sum_{k=1}^{n} qe_j^k$

16

III: FEEDBACK-BASED THRESHOLD ADAPTATION

- The threshold error value and quantization error boundary are used to determine if the connection being processed is within thresholds. If and only if they are identically equal to 0, the pattern is considered within thresholds.
- Each node in the A-GHSOM is marked either “normal”, “unmarked”, or “attack”.
- As patterns are examined, they are mapped to one of the three node types and considered within thresholds or not. Results are used to make predictions to identify the suspected connection type, either “normal” or “attack”.
- The actual connection type is then compared to the suspected connection type and a result of “correct” or “incorrect” is identified. The distinction of correct or incorrect is made based on operator feedback. It is assumed that to some degree, a network operator will be able to identify that an attack prediction was actually a false positive or that an attack was missed. In the absence of feedback, the system assumes that its prediction was correct.
- The map is dynamically updated based on feedback.

17

III: FEEDBACK-BASED THRESHOLD ADAPTATION

Best matching unit type	Within thresholds	Prediction	Result	Adaptation rules
Attack	YES	Attack	Correct	No Action
Attack	NO	Attack	Correct	No Action
Attack	YES	Attack	Incorrect	Grow Network
Attack	NO	Attack	Incorrect	Grow Network
Normal	YES	Normal	Correct	No Action
Normal	NO	Attack	Correct	No Action
Normal	YES	Normal	Incorrect	Lower τ_1, τ_2
Normal	NO	Attack	Incorrect	Raise τ_1, τ_2
Unmarked	YES	Attack	Correct	No Action
Unmarked	NO	Attack	Correct	Mark Node “Attack”
Unmarked	YES	Attack	Incorrect	Mark Node “Normal”
Unmarked	NO	Attack	Incorrect	Mark Node “Normal” & Raise τ_1, τ_2

The adaptation rules.

18

IV: CONFIDENCE FORWARDING

- The fourth enhancement monitors the neuron consistency and accuracy and uses those measures to develop a neuron confidence rating. It then uses the rating to identify an appropriate action for the system to take regarding predictions made by nodes with which the system has low confidence.
- Rationale: In A-GHSOM, due to the addition of dynamic input normalization and feedback based threshold adaptation, is possible for two connections to be matched in an identical region on the map, but be placed into different classification categories depending on their threshold error value and quantization error boundary value.
- We monitor the frequency that each neuron predicts each connection class and use this condition to calculate a consistency rating.

19

IV: CONFIDENCE FORWARDING

- As the number of predictions by a node increases, three possible consistency scenarios can occur.
  - 1) A node trained “attack” consistently makes “attack” predictions.
  - 2) The node trained “normal” consistently makes “normal” predictions.
  - 3) A node trained “normal” makes a significant number of “attack” predictions.
- Situations 1 and 2 are not a concern if the accuracy is also high. If an “attack” neuron consistently makes inaccurate “attack” predictions, new neurons will be added to that region trained to predict “normal” and correct the deficiency.
- Likewise, if a “normal” neuron consistently makes inaccurate “normal” predictions, new neurons will be added to that region trained to predict “attack” and again, the deficiency is corrected.
- The third situation, however, is a major concern.

20

IV: CONFIDENCE FORWARDING

- We identify nodes that have low confidence and exclude traffic that they are processing. Indeed, our experiments have found that these nodes are identifying traffic that is not differentiable within the available data.
- Systems that have access to information such as data content are more suitable to accurately identify those connections. This is where we need turn to content-aware intrusion detection.

21

PERFORMANCE EVALUATION

- The KDD'99 dataset contains records describing TCPconnections. The dataset includes normal connections as well as 23 different types of attacks belonging to four categories: Denial of Service (DOS), Probe attacks, User to Root (U2R), and Remote to Local (R2L). The dataset includes a training set, and two test sets: the "whole" set and the "corrected" set. The "corrected" set includes 17 attack types that are not sampled in the training set and thus are "unknown" to a trained model. For all experiments, we trained our A-GHSOM model on the training set and tested it against the "corrected" set.
- For each connection, a decision is made to process the connection or to forward to the operator based on system confidence. For each connection processed, a classification of "attack" or "normal" is made.
- The performance metrics are the accuracy rate and the false positive rate. The accuracy rate is the ratio of the total number of correct attach predictions over the total number of attacks processed. The false positive rate is the ratio of the total number of false positive predictions over the total number of normal connections processed.

22

PERFORMANCE EVALUATION

- To establish a baseline for performance comparison, we processed the corrected dataset without applying any of the four enhancements. In this configuration, we train the AGHSOM map using *δ_E and δ_D of 0.5 and 100 respectively.* Without using thresholds and adaptations, connections are simply classified based on the node they are mapped to.

CATEGORY	ACCURACY
Overall	92.21
Known	92.81
Unknown	10.95
DOS	98.48
R2L	7.66
U2R	51.43
Probe	78.83

TABLE II
ACCURACY OF THE BASELINE A-GHSOM.

23

IMPACT OF THRESHOLD-BASED TRAINING

- In this experiment, the threshold-based training process is applied, but without adaptations. Furthermore, dynamic input normalization is not used and no operator feedback is assumed. Initial default threshold values for *$\tau 1$ and $\tau 2$ are set to 0.5 and* 1.0 respectively, but not adapted dynamically.

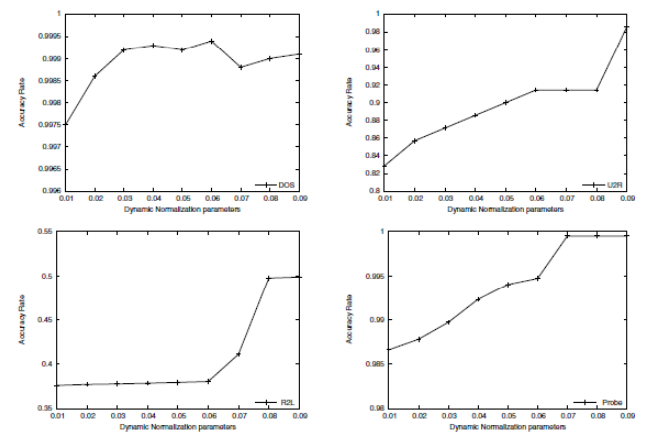
CATEGORY	ACCURACY
Overall	93.53
Known	99.37
Unknown	21.27
DOS	97.59
R2L	35.11
U2R	84.29
Probe	99.06

TABLE III
ACCURACY OF A-GHSOM WITH THRESHOLD BASED TRAINING.

24

IMPACT OF DYNAMIC INPUT NORMALIZATION

- We dynamically change the normalization parameters *α and β from .01 to .09* for every 15000 connections.

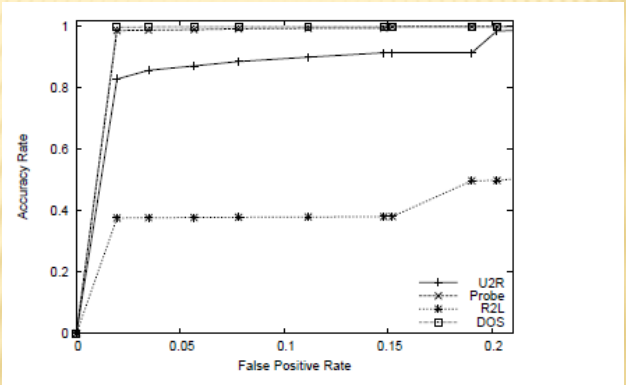


(a) On DOS accuracy. (b) On U2R accuracy.
(c) On R2L accuracy. (d) On PROBE accuracy.

25

IMPACT OF DYNAMIC INPUT NORMALIZATION

- Increased accuracy however comes at a cost of higher false positive rate because the A-GHSOM is too sensitive to the normalization parameters.

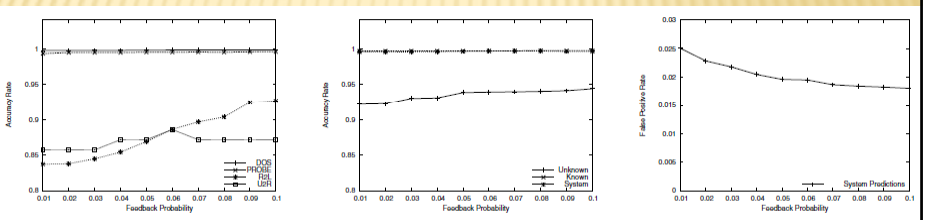


ROC diagrams of A-GHSOM with the dynamic input normalization.

26

IMPACT OF FEEDBACK THRESHOLD ADAPTATION

- Until now, the system is autonomously adapting. That is, no expert feedback is used. Next, operator feedback is used to further enhance the accuracy of intrusion detection. We use a control parameter of feedback probability to determine the likelihood that the system will be notified that it has made an error. The probability varies from 1% to 10%. Dynamic input normalization parameters and *are set to 0.04 per every 15000 connections.*



(a) Accuracy by categories. (b) Accuracy by known & unknown. (c) False positive rate.

IMPACT OF CONFIDENCE FILTERING/FORWARDING

- We found that even with 100% connection feedback, the AGHSOM approach has the false positive rate around 8% while producing excellent accuracy for both known and unknown attacks. We found that approximately 85% of the false positives were predicted by nodes that identified connection patterns not differentiable within the KDD dataset.

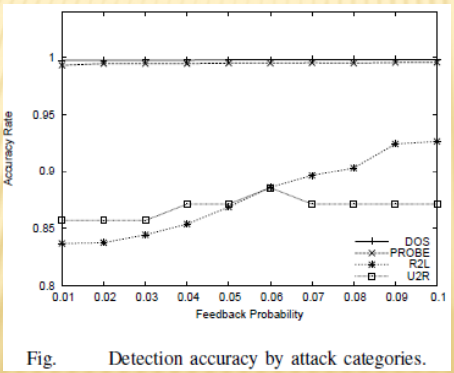


Fig. Detection accuracy by attack categories.

IMPACT OF INTEGRATED A-GHSOM

- Here, The system is adapted every 20000 connections using different parameters. Input normalization parameters *α and β are set to .03, and the* confidence forwarding probability is 80%. In the experiment, 91.9% of the connections are classified by the integrated approach. 8.1% of the connections are selected for forwarding.

Approach	Accuracy	False Positive
A-GHSOM	99.63%	1.8%
GHSOM [11], year 2008	90.87%	2.69%
Bouzida [1], year 2004	91.89%	0.48%
Sarasamma [12], year 2005	93.46%	3.99%
Kayacik [9], year 2007	90.4%	1.38%
Eskin Data Mining [4], year 2002	90.00%	2.0%
Eskin Clustering [4], year 2002	93.00%	10.0%
Eskin SVM [4], year 2002	98.00%	10.0%
Yu [17], year 2008	96.02%	4.92%

TABLE IV
SUMMARY OF PERFORMANCE COMPARISONS.

29

CONCLUSION AND FUTURE WORK

- A-GHSOM is able to consistently produce higher accuracy rates while maintaining low false positive rates. Its false positive rate is lower than all approaches except two approaches proposed in [SAR 2004] and [AI 2007]. Note that the false positive rate is just slightly higher (1.32% and 0.42%), but the improvement in the major performance metric accuracy is almost 10% higher. Accuracy and false positive are always tradeoffs. A-GHSOM is able to make the efficient tradeoff, and it is adaptive to a changing problem domain of network intrusion.
- The significance of the A-GHSOM lies in the online adaptation to the ever-changing problem domain of network intrusion and achieving very high accuracy in identifying network intrusions particularly those “unknown” attacks. Experiment results demonstrated the significant impact of each A-GHSOM enhancement on intrusion detection performance.
- Future work will develop the self-tuning capability of A-GHSOM.

30

IV: CONFIDENCE FORWARDING

- We monitor the frequency that each neuron predicts each connection class and use this condition to calculate a consistency rating. After each connection is processed at time *t and a prediction made by* neuron *j, consistency for the predicting node is calculated* according to:

$$Attack_t^j = \frac{\lambda(Attack_{t-1}^j + PREDICT_{Attack})}{\lambda + 1}.$$

$$Normal_t^j = \frac{\lambda(Attack_{t-1}^j + PREDICT_{Normal})}{\lambda + 1}.$$

$$Consistancy_t^j = \frac{max(Attack_t, Normal_t)}{Attack_t + Normal_t}.$$

- where *λ is the size of the history, PREDICTAttack = 1 if an* attack is predicted and 0 otherwise, and *PREDICTNormal* = 1 − *PREDICTAttack.*

31

IV: CONFIDENCE FORWARDING

- Because A-GHSOM also expects feedback from a network operator, we are able to estimate an accuracy rating for each node. As the amount of feedback received by the system is limited, the system assumes that its predictions are correct in the absence of feedback. *PREDICTAccurate is equal to 0 if* corrective feedback is received and 1 otherwise. Accuracy for predicting node *j is calculated according to*

$$Acc_t^j = \frac{\lambda(Acc_{t-1}^j) + PREDICT_{Accurate}}{\lambda + 1}.$$

- We use a combination of consistency and accuracy to calculate a confidence rating as follows.

$$Confidence_t^j = \frac{\lambda(Confidence_{t-1}^j) + Acc_t^j + Consistancy_t^j}{\lambda + 1}.$$

32