Acta Technica 62 (2017), No. 7A, 1649–1658

Cloud computing based packet intrusion detection for hyper fusion network storage

Zaiyi Pu[1]

Abstract. The design method of intrusion detection system based on Convolutional Neural Network (CNN) of hypergraph feature reduction is proposed to improve the precision and calculation efficiency of network intrusion detection system. Firstly, the feature subset representation form can be optimized based on the minimum distance measurement of hypergraph structure and represented as more significant form by the high order relation between the entities in the actual world to achieve data reduction; secondly, the optimal feature subset is identified by hypergraph Helly feature recursion method and the intrusion detection is classified by the residual error-based CNN; finally, the experimental comparison on KDD CUP 1999 benchmark test set shown that the proposed method has better performance on detection precision, recall rate and stability index.

Key words. Hypergraph, Feature selection, Data reduction, Convolutional Neural Network, Intrusion detection

.

1. Introduction

Based on the technical progress of computer and network application program, the security hole problems are gradually increased. As complexity of new hole, the traditional security mechanism, such as user identification, is technologically immature in protecting IT infrastructure from hostile attack by the users [1-2]. Therefore, the Intrusion Detection System (IDS) is designed as the second line of defense to protect the network from suffering threats and abnormal activities to achieve automatic detection and identification of intrusion detection. Generally, IDS can be classified as the signature-based and abnormality-based detection system [3]. The performance of the signature-based IDS depends on frequent update of new

[1]Network And Information Management Center, China West Normal University, Nanchong, Sichuan, 637009

intrusion model; the abnormality-based IDS depend on the appropriate baseline configuration.

With gradually increasing of network flow, the researches on IDS gradually develop towards automation and intelligence based on statistics, machine learning technique and expert system. In the different machine learning techniques, the Support Vector Machine [4] (SVM) and Artificial Neural Network [5] (ANN) have been successfully applied in IDS because its model complexity is not based on the relation between variables. In addition, the incomplete or distorted dataset can be processed by nonlinear data processing and the strong self-learning ability attaches great importance on IDS event prediction. ANN IDS can be classified in two types: supervised and unsupervised. Its disadvantages mainly include the following [6-7]: (1) the unbalanced dataset has lower detection rate due to its lower attach frequency; (2) the artificial neural network structure has higher instability due to local minimum high dimensional dataset traps.

To overcome these challenges, the enhanced edition of CNN based on hypergraph feature selection is proposed to improve the detection rate of IDS. The feature subset is represented as optimizing form based on minimum distance measurement and feature subset. The optimal feature subset is identified by hypergraph Helly feature recursion method and the intrusion detection is classified by CNN to achieve effective detection.

2. Problem description

2.1. convolutional neural network

CNN[8] is designed by Specht and estimated by probability density. The core concept of CNN is the "winner-takes-all strategy" which adopts the multi-element probability estimation and learning competition. It is a classifier edition integrating the Bayes strategy with Parzen window and non-parametric estimation probability density function (PDF). Different from the traditional radial basis function (RBF) and multi-layer feed-forward network, the data can be processed based on the statistics principles by the neural network training. Because CNN is based on PDF estimation, not the iterated function approximation, it has higher training speed and good generalization ability.

The classification is made by Bayes method. The probability class of unknown input vector is classified based on historical data, not the model parameters, such as average value and standard deviation. Bayes classifier can be configured as below:

$$P(C_i|x) = \frac{f(\chi|C_i) * P(C_i)}{f(x)}\,. \tag{1}$$

Where, $P(C_i|x)$ is the posterior probability, representing the probability that input x belongs to class$i$. For any classification problem, the posterior probability of class i is calculated and the input x with maximum $P(C_i|x)$ is classified into class i. $P(C_i|x)$ is calculated based on the prior probability $P(C_i)$ obtained from the historical data. The class condition probability $f(\chi|C_i)$ can be estimated based on the Parzen window training data. The non-parametric estimator based on Gaussian probability density function can be represented as below:

$$f(x|C_i) = \frac{1}{(2\pi)^{n/2}}\left[\frac{1}{m}\sum_{j=1}^{m} e^{-\left(\frac{1}{2\sigma^2}\right)\left[\left(x-x_{c_{ij}}\right)^T\left(x-x_{c_{ij}}\right)\right]}\right]. \tag{2}$$

Where, m represents the pattern quantity in class C_i; $x_{c_{ij}}$ represents j pattern in class C_i; σ represents smoothing parameter.

2.2. CNN structure

CNN has four-layer structure, including the nodes. It can map the input class to discrete class as shown in Fig. 1. The function of each layer structure of CNN is described as below:

Layer 1: input layer and input unit. Any calculation process is not executed in the input layer. The function of input layer is to distribute the input elements to the convolutional layer;

Layer 2: convolutional layer. After the training sample is received from the input layer, the node in convolutional layer can be calculated as below:

$$exp\left[\frac{(x_T * x) - 1}{\sigma^2}\right]. \tag{3}$$

Where, x_T is the training data pattern, x is the unknown pattern of the given class and σ is the smoothing factor.

Layer 3: convergence layer. Each node in this layer can be used to calculate the convergence value of output value in convergence layer. The calculation method is shown as below:

$$S_L = \sum_{i=1}^{C} exp\left[\frac{(x_T * x) - 1}{\sigma^2}\right]. \tag{4}$$

Where, C is the total number of class.

Layer 4: output layer. The node in output layer based on Bayes rules can determine the class of each input pattern x and its form is shown as below:

$$\sum_{i=1}^{C} exp\left[\frac{(x_T * x_i) - 1}{\sigma^2}\right] > \sum_{j=1}^{C} exp\left[\frac{(x_T * x_j) - 1}{\sigma^2}\right]. \tag{5}$$

2.3. Hypergraph structure and features

The hypergraph is the traditional graph theory extension and can be represented as more significant form by the high order relation between the entities in the actual world. In mathematics, the hypergraph can be defined as $H = \{X, E\}$, where is

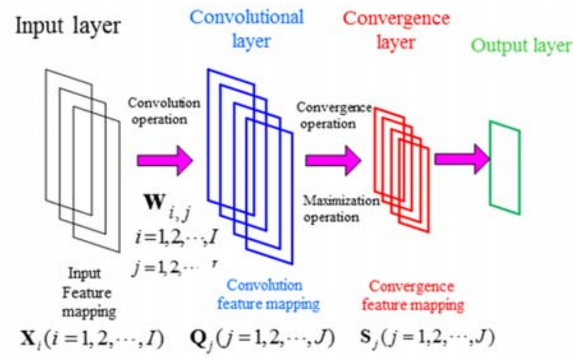

Fig. 1. CNN structure

$x=\{x_1,x_2,\cdots,x_n\}$ the non-empty finite vertex set; $E=\{E_1,E_2,\cdots,E_m\}$ is the non-empty subset of X, called as hypergraph as shown in Fig. 2. In this section, the definition and features of hypergraph are analyzed.

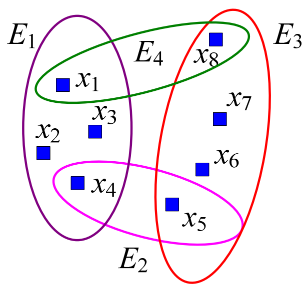

Fig. 2. Hypergraph structure

Definition 1: taking into consideration of hypergraph $H=\{X,E\}$, for each vertex $y\in X$, hyperedge set with Y is called as spot of H $\left(H_y^*\right)$ as shown in Fig. 3. The dimension of D_y is equal to the base number of H_y^*:

$$D_y=\left|H_y^*\right| . \tag{6}$$

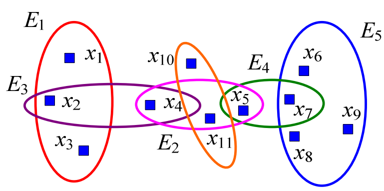

Fig. 3. hypergraph representation model

Where, including the vertex set $x=\{x_1,x_2,\cdots,x_9\}$ and hyperedge set $E=\{E_1,E_2,\cdots,E_5\}$, x_2 is centre-of-gravity position of $H_{y_2}^*=\{E_1,E_2\}$, the dimension of D_{y_2} is $D_{y_2} = 2$, x_5 is centre-of-gravity position of $H^*_{y_5} = \{E_2, E_4\}$ and dimension of D_{y_5} is $D_{y_5} = 2$.

Definition 2: for the given hypergraph $H = \{X, E\}$, one intersecting cluster of H is cluster of hypergraph $E \subseteq X$ and is the non-empty intersection set.

Definition 3: taking into account of hypergraph H with hyperedge set $\{E_1, E_2, \cdots, E_m\}$, if the intersection set of E_i and E_j is not empty, $i, j \in k$, $k = \{1, 2, \cdots, m\}$. The pair-wise intersecting edges of H includes the following two cases:

Case 1: for pair-wise intersection with edge of normal intersection as shown in Fig. 4a, the pair-wise intersecting edge $\{E_1, E_2, E_3\}$ causes the vertex x_3 serve as the common intersection point.

Case 2: for pair-wise intersection without common intersection set as shown in Fig. 4b, the pair-wise intersecting edge $\{E_1, E_2, E_3\}$ has no common intersection point.

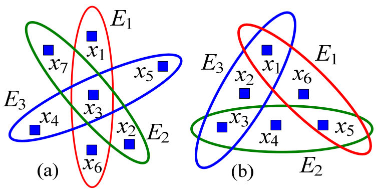

Fig. 4. Classification of pair-wise intersecting edge ((a) with common intersection point, (b) without common intersection point)

3. Hypergraph-based CNN

The main objective of any feature selection technique is to reduce data dimension and maintain good classification precision. For many pattern recognition problems, the text, spectrum, topology, geometry and statistical characteristics are used to achieve the learning model training. Under the imbalanced dataset, the generalization error of learning model is increased due to existence of redundancy features. To overcome the above defects, the hypergraph data representation tool is used to minimize the time complexity and achieve the optimal feature subset identification.

The hypergraph-based feature selection algorithm (algorithm 1) proposed includes two stages: (1) hypergraph representation; (2) Helly feature application. In the first stage, the edge of hypergraph can be obtained by the sample topology and geometrical relationship. The edge and vertex of hypergraph respectively correspond to sample and feature of dataset. For the supervised learning, the Euclidean distance measurement method based on the minimum distance algorithm is used to construct the edge of each class:

$$E_d(x, y) = ||x - y|| \,. \tag{7}$$

In the next stage, for optimal reduction identification, the hypergraph features can be used for the intersecting edge by recursion method. In the processing procedures, the non-intersecting edge can be ignored. For unsupervised learning, the distance between clusters can be maximized and the distance in the cluster can be minimized by the city block distance (CBD) distance measurement method.

$$CB_d = \max(|x_1 - x_2|, |y_1 - y_2|)\,. \tag{8}$$

Based on the Helly features of hypergraph, the complexity generated from optimal time reduction is the minimum. The feature selection based on hypergraph Helly features and the CNN recognition process are respectively shown in algorithm 1 and 2.

Algorithm 1: feature selection based on hypergraph Helly features

Input:$f \leftarrow \{f_1, f_2, \cdots, f_m\}$, // m features of given dataset

$S = \{S_1, S_2, \cdots, S_n\}$, // n features of given dataset

$C = \{C_1, C_2, \cdots, C_k\}$; // k features of given dataset

Output:f_s ←optimal feature subset

$HG\,(f, s, c, f_s)$

1. for each $i \leftarrow 1$:k do
2. $\quad Hyperedge\,[i] \leftarrow \min\,[E_d\,(f_i, f_j)]$;
3. end
4. for each $i \leftarrow 1$:k begin
5. $\quad f_s \leftarrow \bigcap_{j\in i} Hyperedge\,[j]$;
6. end

Algorithm 2: Residual error-based CNN recognition process

Input : total number of class:Tot_{class}, S_{Train}, f_s, σ;

Output: $Classify$ ←classification precision of test vectors

CNN (Tot_{class}, S_{Train}, f_s, σ)

1. Initialize the parameter value:$Largest = 0$, $Sum = 0$, $Classify = -1$, $\varepsilon = 0.1$;
2. Calculate the training residual error parameter:

$Train_{Data}\,[i, j] \leftarrow \sum_{k=1}^{f_s} \left[\left(\frac{f_k}{\varepsilon}\right) - \left\lfloor \frac{f_k}{\varepsilon} \right\rfloor\right]$

$\forall i = \{1, \cdots, S_{Train}\}\,, j = \{1, 2, \cdots, f_s\}$

3. Calculate the testing residual error parameter:

$Test_{Data}\,[i] \leftarrow \sum_{k=1}^{f_s} \left[\left(\frac{f_k}{\varepsilon}\right) - \left\lfloor \frac{f_k}{\varepsilon} \right\rfloor\right]$

$\forall i = \{1, 2, \cdots, f_s\}$

4. for each $k = 1 : Tot_{class}$ begin
5. $\quad Sum\,[k] \leftarrow 0$;
6. for each $i \leftarrow 1$:S_{Train} begin
7. $\quad p \leftarrow 0$;
8. for each $j \leftarrow 1$ begin
9. $\quad p \leftarrow p + (Test_{dataset}\,[j] \times Train_{dataset}\,[j]\,[j])$

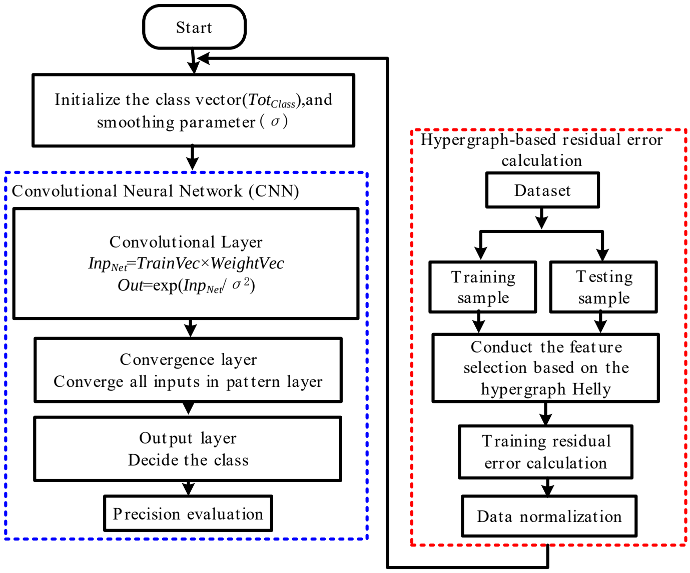

Fig. 5. Hypergraph-based CNN algorithm flow

4. Experimental analysis

4.1. Experiment settings

The network intrusion dataset with unbalanced standards (KDD CUP 1999 test set) is used in this experiment. As shown in Table 1, KDD CUP 1999 intrusion data includes 22 types of attacks, namely, DOS, U2R, R2L and Probe. It has typical unbalance and is used to verify the proposed algorithm. In each type, the sample is not uniformly distributed. For example, DoS attack includes a large number of samples, but probe, U2R and R2L have less samples.

Table 1. KDD CUP 1999 test set attack list or address

No.	Type	Attack
1	DoS	Neptune, Smurf, tear drop, land, back, pod
2	Probe	satan, Ipsweep, portsweep, nmap
3	R2L	Guesspassword, Ftpwrite, Spy, Phf, Warezclient, Imap, Mutlihop, Warezmaster
4	U2R	Buffer, Perl, overflow, Rootkit, Loadmodule

Algorithm running environment: Matlab2012a, processor i7-7700HQ, memory 8GB ddr3-1600, system win7 flagship version. The quantity of DoS, Probe, R2L and U2R training sample (testing sample) is 8435(123166), 3000(4011), 1126(16189) and 52(288)respectively. The detection precision index, recall rate index and detection stability index are selected as the evaluation index:

$$Precision\,(P_{value}) = \frac{T_P}{T_P + F_P}\,. \tag{9}$$

$$Recall\,(R_{value}) = \frac{T_P}{T_P + F_N}\,. \tag{10}$$

$$Stability\,(\%\ ) = \frac{\text{Successful training sample quantity}}{\text{Total training sample quantity}}\,. \tag{11}$$

Where: T_P represents the measurement of malicious act which is deemed as attack, T_N represents the measurement of act which is determined as normal act, F_P represents the measurement of act which is misjudged as attack and F_N represents the measurement of act which is misjudged as normal act.

4.2. Experiment result analysis

The naive Bayes algorithm, decision tree algorithm and random prediction algorithm are selected as comparison algorithm to verify the algorithm effectiveness. The experiment results are shown in Table 2.

Table 2. Experiment result comparison (%)

Attack type	Index	Bayes	Decision tree	Random prediction	Proposed algorithm
DoS	Precision	96.82	97.52	91.26	98.32
	Recall	95.21	96.13	90.58	97.64
	Stablity	92.68	93.84	89.64	94.56
Probe	Precision	48.76	47.53	41.62	82.16
	Recall	50.76	51.69	48.75	81.28
	Stablity	51.26	52.76	46.21	79.46
R2L	Precision	33.21	34.27	30.69	83.64
	Recall	36.38	35.19	32.84	82.93
	Stablity	33.27	34.68	30.42	81.69
U2R	Precision	23.16	29.16	28.94	72.59
	Recall	23.82	30.67	27.63	73.64
	Stablity	25.39	28.91	24.18	71.82

According to the experiment data in Table 2, in DoS attack, the prediction precision, recall rate index and stability index of these algorithms are higher and can reach over 90%, which shows the simplicity of DoS attack prediction. The proposed algorithm is superior to the selected comparison methods in terms of the above indexes. In U2R attack, the prediction precision, recall rate index and stability index are the minimum. The index values of these algorithms are kept at 20%-30% and the recognition effects are poor. However, the index value of the proposed algorithm can reach above 70%, which shows good abnormal attack recognition performance.

The comparison of training convergence time for the selected algorithms is shown in Fig. 6.

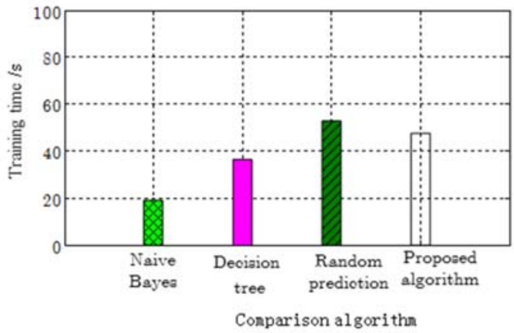

Fig. 6. Algorithm running time comparison

According to the results in Fig. 6, as for the running time index, the running time of the proposed algorithm is longer than that of naïve Bayes algorithm and decision tree algorithm, but shorter than that of random prediction algorithm. As the off-line training mode is adopted for the above algorithms in attack prediction, the running time difference is acceptable.

5. Conclusion

The design method of intrusion detection system based on Convolutional Neural Network (CNN) of hypergraph feature reduction is proposed in this paper. The feature subset representation form can be optimized based on the minimum distance measurement of hypergraph structure. The intrusion detection is classified by the residual error-based CNN. The experiment results show that the proposed algorithm has prediction precision advantages. Although the hypergraph data reduction is adopted, the computing time is relatively longer and shall be further optimized in the subsequent researches.

Acknowledgement

Excellent talents foundation of china west normal university(NO:17YC497).

References

[1] Y. Chen, F. J. Tang, B. Yi, T. Yan, G. Chen: *A Fe-C coated long period fiber grating sensor for corrosion induced mass loss measurement.* Optics letters *41* (2016), 2306–2309.

[2] N. Arunkumar, S. Jayalalitha, S. Dinesh, A. Venugopal, D. Sekar: *Sample entropy based ayurvedic pulse diagnosis for diabetics.* IEEE-International Confer-

ence on Advances in Engineering, Science and Management, ICAESM-2012, art. No. 6215973 (2012), 61–62.

[3] Y. J. Zhao, Y. H. Hu, J. J. Liu: *Random Triggering-Based Sub-Nyquist Sampling System for Sparse Multiband Signal.* IEEE Transactions on Instrumentation and Measurement *66* (2017), NO. 7, 1789–1797.

[4] X. Du, L. Chen, D. Huang, Z. Peng, C. Zhao, Y. Zhang, Y. Zhu, Z. Wang, X. Li, G. Liu: *Elevated Apoptosis in the Liver of Dairy Cows with Ketosis.* Cellular Physiology & Biochemistry *43* (2017b), No. 2, 568–578.

[5] N. Arunkumar, R. K. Kumar, V. Venkataraman: *Automatic detection of epileptic seizures using permutation entropy, Tsallis entropy and Kolmogorov complexity.* Journal of Medical Imaging and Health Informatics *6* (2016), No. 2, 526–531.

[6] Y. Y. Zhang, A. Algburi, N. Wang, V. Kholodovych, D. O. Oh, M. Chikindas, and K. E. Uhrich: *Self-assembled Cationic Amphiphiles as Antimicrobial Peptides Mimics: Role of Hydrophobicity, Linkage Type, and Assembly State*, Nanomedicine: Nanotechnology, Biology and Medicine *13* (2017), No. 2, 343–352.

[7] Y. Song, N. Li, J. Gu, S. Fu, Z. Peng, C. Zhao, Y. Zhang, X. Li, Z. Wang, X. Li: *β-Hydroxybutyrate induces bovine hepatocyte apoptosis via an ROS-p38 signaling pathway.* Journal of Dairy Science *99* (2016), No. 11, 9184–9198.

[8] N. Arunkumar, K. R. Kumar, V. Venkataraman: *Automatic detection of epileptic seizures using new entropy measures.* Journal of Medical Imaging and Health Informatics, *6* (2016), No. 3, 724–730.

[9] R. Hamza, K. Muhammad, N. Arunkumar, G. R. González: *Hash based Encryption for Keyframes of Diagnostic Hysteroscopy*, IEEE Access, https: //doi.org/10.1109/ACCESS.2017.2762405 (2017).

[10] D. S. Abdelhamid, Y. Y. Zhang, D. R. Lewis, P. V. Moghe, W. J. Welsh, and K. E. Uhrich: *Tartaric Acid-based Amphiphilic Macromolecules with Ether Linkages Exhibit Enhanced Repression of Oxidized Low Density Lipoprotein Uptake*, Biomaterials, *53* (2015) , 32–39.

[11] W. Pan, S. Z. Chen, Z. Y. Feng: *Automatic Clustering of Social Tag using Community Detection.* Applied Mathematics & Information Sciences *7* (2013), No. 2, 675–681.

[12] Y. Y. Zhang, E. Mintzer, and K. E. Uhrich: *Synthesis and Characterization of PEGylated Bolaamphiphiles with Enhanced Retention in Liposomes*, Journal of Colloid and Interface Science *482* (2016), 19–26.

[13] N. Arunkumar, K. M. Mohamed Sirajudeen: *Approximate Entropy based ayurvedic pulse diagnosis for diabetics - A case study.* TISC 2011 - Proceedings of the 3rd International Conference on Trendz in Information Sciences and Computing, art. no. 6169099 (2011), 133–135.

Received May 7, 2017