ATLANTIS PRESS

Advances in Computer Science Research, (ACSR), volume 52

2016 International Conference on Computer Engineering and Information Systems (CEIS-16)

Concept Drift Based on Subspace Learning for Intrusion Detection

Bin Wu, Hai-Zhuo Lin, Lin Feng

School of Innovation and Entrepreneurship, Dalian University of Technology, Dalian, China

E-mail: wubdut@gmail.com

***Abstract*-In recent years, Intrusion Detection System(IDS) thrives and becomes the main approach for detecting and defending internet attack. And network streams are the best data sources for studying network attack. In order to detect intrusions, concept drifting method is applied. What is more, the subspace learning based concept drifting method is fit for dealing with high dimensional data streams. It can not only detect the concept drift, but also reduce the dimensionality at the same time, which makes the detection more efficient. We also design model for judging concept drift, which checks the deviation of the error term of projection variance and the deviation of the error term of projection cosine. The experiment of KDD data set validates that our method is more efficient and accurate.**

Keywords-intrusion detection; concept drift; subspace learning

I. INTRODUCTION

In recent years, with the rapid development of internet and multimedia technology, Cyber security has become a focus of society. As a consequence, Intrusion Detection System(IDS) thrives and becomes the main approach for detecting and defending internet attack. At the beginning, IDS was designed to distinguish the normal and suspicious activities in the network [1]. IDS has three main components, such as data collection, detection and response. The data collection component works for collecting the data from various sources such as audit data, network traffic data, etc. Detection component works for analyzing the collected data to detect intrusions, and if any suspicious activity is detected, the response component reports intrusions. In the literature, misuse based, anomaly based and specification based techniques are the main detection methods [2], [3], [4]. Misuse based detection system detects intrusions according to predefined attack signature. Anomaly based detection technique detects the intrusion on the basis of the system's normal behavior. As the behavior of system changes by time, defining the normal behavior of the system is full of challenge. This method can detect the unknown and new attack, but have high false positive rates. Specification based intrusion detection specifies or defines the set of constraints on a specific protocol and then detects the intrusions at the running time violation of these specifications. Therefore, defining the specification is really a time consuming task. There are generally three basic types of IDS architecture in literature: Stand-alone or local intrusion detection systems, Distributed and Cooperative intrusion detection systems, Hierarchical Intrusion Detection Systems [4].

As network applications are applied more and more generally, the network stream is the best data source for studying network attack. Thus, IDS based on network draws most researchers' attentions and most applied in this field nowadays. Compared with finite stationary databases, data streams are characterized by their concept drifting aspects [5], [6], which means the learned concepts or the underlying data distribution are not stable and may change over time. In addition, data streams bring many challenges to computing systems for limited memory resources (i.e., streams can not be totally stored in memory), and time (i.e., streams should be continuously processed and the learned classification model should be ready at any time to be used for prediction) [7].

In recent years, researchers in the field of data mining has paid an increasing attention to mining concept drifting data streams and a lot of approaches have been developed and deployed in applications [6], [8], [9], [10], [11], [12]. All these approaches have the same main objective consists of tackling the concept drift and maintaining updating the classification model along the continuous flows of data. [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23] have proposed some efficient and effective approaches of concept drifting data streams learning. In order to deal with concept drifting data streams with unlabeled data, [24] proposed a semi-supervised method based on decision tree algorithm---SUM. Based on a concept set, SUN produced concept clusters at leaves to detect a concept drift. It searches the decision tree from bottom to top to evaluate the deviation and distance between history concept and current concept by setting a variable. [25] analyzed the statistical meaning of Bayes formula in data stream applications from statistical angle. [13] presented the state-of-the-art algorithms in the field of data streams and then gave a critical judgment on these existing methods. At the same time, they pointed out that concept drift detection methods should compare the similarities and differences straightly among concepts in different time, while most of the existing drift detecting methods take a indirect route with analyzing the reasons of concept drift and predicting the possible results of concept drift. In [26], based on the assumption that the points in the stream are generated independently, Kifer et al. introduced a novel method to detect and estimate changes in data streams. Lazarescu and Venkateshp [27] used selective memory to track concept drift, which makes the drifting detection more accurate and the data noise filter more effective.

In IDS, coping with high dimensional data streams is also an tough problem which concept drifting method should be faced with. In order not to affect the detection of new arrived data stream, IDS usually need to have a quick reaction to the

current data stream, which requires efficiency in concept drifting detection. The curse of dimensionality makes the traditional machine learning method not appropriate for this task. In [28], Dasu et al. proposed one classical concept drift detecting method. They apply an information-theoretic approach to the high dimensional data to detect concept changes. This method measures the difference between two given distributions by Kullback-Leibler (KL) distance. However, there are some limitations while utilizing KL distance: 1) it needs discretization to calculate probability density; 2) it can only handle concept drift between two classes and multiclass can only be decided based on results of two classes; 3) the process of bootstrap and discretization is time consuming. In [30], Sries and Rckert presented three novel testing methods based on statistics for drifting detection and evaluated performances of several different methods that are applied to concept drifting detection.

In concept drifting detection, the above method are all time consuming. Concept drifting detection is always an independent process without considering effects on classification of data streams. Subspace learning is a popular topic in machine learning, such as classical manifold learning methods [29], [30], [31], [32], [33]. In order to solve classification problem, some researchers proposed supervised dimensional reduction methods, such as Linear Discriminant Analysis (LDA). LDA [34] aims to minimize within-class scatter and maximize between-class scatter. Feng et al. proposed a subspace learning based concept drifting method---ARLDA [35] which uses projectional matrix to estimate if concept drift occurs, and reduces the dimensionality of data at the same time.

This paper uses LDA to calculate the projectional matrix for obtaining projection variance which describes the distribution of variance after projection and projection cosine which describes the distribution of angel after projection. We also design the concept drifting conditions to judge if concept drift occurs, based on the projection variance and projection cosine of normal and suspicious samples, respectively. For the data set, we treat every 100 samples as a data stream, and then conduct our training and testing experiments. The subspace based concept drifting method performs more accurate than No-Detection method and more efficient than KL-Measure method.

II. A BRIEF OF LDA

The dataset is denoted as $X = [x_1, \cdots, x_N] \in R^{D \times N}$, which satisfies $N = \sum_{i=1}^{c} N_i$. where N_i is the sample size of class i , and c is the number of classes. Linear projection is denoted as $W = [w_1, w_2, \cdots, w_d] \in R^{D \times d}$. Where d is the dimensionality of the target subspace. The optimal process can be described as follow:

$$\begin{aligned} &\max C(W) = \frac{trace(W^T S_b W)}{trace(W^T S_w W)} \\ &s.t. \left\| W^T S_w W \right\| = I \end{aligned} \tag{1}$$

Where $S_b = X_b X_b^T$ is the within-class scatter matrix, $X_b = [\mu_1 - \mu, \mu_2 - \mu, \cdots, \mu_c - \mu]$, μ_i is the mean vector of class i , μ is the global mean vector; $S_w = X_w X_w^T$, $X_w = X(I - \frac{1}{N} 11^T)$, $1 = [1,1, \cdots ,1]_{1 \times N}^T$.

Equation (1) equals to:

$$S_b W = S_w \Lambda W \tag{2}$$

If S_w is reversible, the above equation can translate to:

$$S_w^{-1} S_b W = \Lambda W \tag{3}$$

With eigen-decomposition or singular value decomposition (SVD), the orthogonal matrix is composted of eigenvectors corresponding to the top d largest eigenvalues of Λ .

III. MODEL FOR JUDGING CONCEPT DRIFT

In this section, we use a concept drifting approach to detect network intrusions. We judge if concept drift occurs by the subspace projectional matrix based on LDA. Namely, when the subspace changes a lot, we assume concept drift occurs. In this case, data stream needs to be relearnt to obtain new projectional matrix for detecting new coming data streams.

We use projection variance which is used to analyze distribution of variance after projection and projection cosine which is used to analyze distribution of angel after projection to detect concept drift. The projection variance can only reflects the dispersion among data while the projection cosine only reflects the orientation of data. As both variance and orientation are necessary for analyzing real-world data which may have noise, We use both to detect concept drift.

The overall framework of detection is illustrated in Fig. 1. We can detect concept drift in data streams via utilizing projection variance and projection cosine. The whole sample set is $X = \bigcup_{i=1}^{t} X_i$, where X_i is the sample set of the ith data stream. $X_i = X_{i1} \bigcup X_{i2}$, where X_{i1} and X_{i2} represent the normal sample set and the suspicious sample set of the ith data stream, respectively. The

corresponding sample size of X_i, X_{i1}, and X_{i2} are N_i, N_{i1} and N_{i2}. Thus, $N_i = N_{i1} + N_{i2}$, and $N = \bigcup_{i=1}^{t} N_i$.

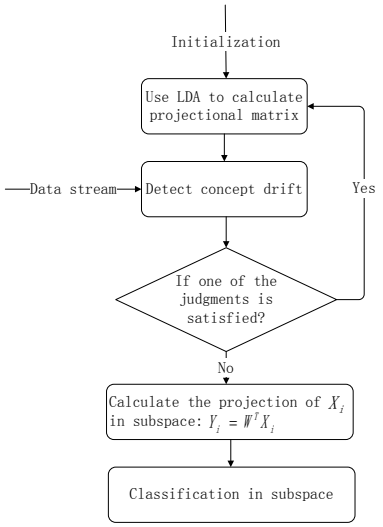

Figure 1. The overall framework of detection.

W_i is the projectional matrix, Y_i is the projection of X_i in subspace:

$$Y_i = X_i W_i \tag{4}$$

The center of the ith data stream is:

$$\mu_i = \frac{1}{N_i} \sum_{x \in X_i} x \tag{5}$$

$\tilde{\mu}_i$ is the projection of μ_i in subspace:

$$\tilde{\mu}_i = \mu_i W_i \tag{6}$$

We use the first two data streams X_1 and X_2 to initialize the low-dimensional subspace projectional matrixes W_1 and W_2, then determine if concept drift occurs when the ith data stream comes.

When we deal with the ith data stream, we calculate projection variance (7) and projection cosine (11) at first. Then we obtain error term of projection variance (8) and error term of projection cosine (12). Finally, we design judgments by using error term of projection variance and error term of projection cosine.

The projection variance is a two-dimensional vector. The first dimension indicates the variance of normal data, and the second dimension indicates the variance of suspicious data:

$$V_i = \left(\frac{trace((Y_{i1} - \tilde{\mu}_i)(Y_{i1} - \tilde{\mu}_i)^T)}{N_{i1}}, \frac{trace((Y_{i2} - \tilde{\mu}_i)(Y_{i2} - \tilde{\mu}_i)^T)}{N_{i2}}\right)^T \tag{7}$$

Then, we can get error term of projection variance:

$$\Delta V_i = \left\| V_{i+1} - V_i \right\| \tag{8}$$

Assuming that x is a sample, where $x \in X_i$. The projection cosine can be:

$$\cos\beta = \frac{\left\|(x - \mu_i) W_i\right\|}{\left\|x - \mu_i\right\| \cdot \left\|W_i\right\|} \tag{9}$$

Then,

$$\cos^2\beta = \frac{[(x - \mu_i)W_i][(x - \mu_i)W_i]^T}{(x - \mu_i)(x - \mu_i)^T} = \frac{(y - \tilde{\mu}_i)(y - \tilde{\mu}_i)^T}{(x - \mu_i)(x - \mu_i)^T} \tag{10}$$

The projection cosine is also a two-dimensional vector. The first dimension indicates the mean value of normal data's projection cosine, and the second dimension indicates the mean value of suspicious data's projection cosine:

$$E_i = \left(\frac{\sum \frac{(y - \tilde{\mu}_i)(y - \tilde{\mu}_i)^T}{(x - \mu_i)(x - \mu_i)^T}}{N_{i1}}, \frac{\sum_{x \in X_{i2}} \frac{(y - \tilde{\mu}_i)(y - \tilde{\mu}_i)^T}{(x - \mu_i)(x - \mu_i)^T}}{N_{i2}}\right)^T \tag{11}$$

Then, we can get error term of projectional cosine:

$$\Delta E_i = \left\| E_{i+1} - E_i \right\| \tag{12}$$

At the end, we get concept drifting judgments.
Use projection variance to judge if concept drift occurs:

$$\left|\Delta V_{i-1} - \Delta V_{i-2}\right| > \frac{1}{\lfloor (i-1)/2 \rfloor} \sum_{k=2}^{\lfloor (i-1)/2 \rfloor} \left|\Delta V_k - \Delta V_{k-1}\right| \tag{13}$$

Use projection cosine to judge if concept drift occurs:

$$\left|\Delta E_{i-1} - \Delta E_{i-2}\right| > \frac{1}{\lfloor (i-1)/2 \rfloor} \sum_{k=2}^{\lfloor (i-1)/2 \rfloor} \left|\Delta E_k - \Delta E_{k-1}\right| \tag{14}$$

When one of (13) and (14) is satisfied, we assume that concept drift occurs in the ith data stream. In this case, the subspace projectional matrix W_i need to be updated to adjust the new data stream. Otherwise, we assume that concept drift does not occur, and $W_i = W_{i-1}$. In addition, the subspace projectional matrix which is calculated by using LDA, not only plays the role in judging if concept drift occurs, but also reduces the dimensionality greatly, which makes data

processing more efficient. In this paper, we use Support Vector Machine (SVM) to classify the data.

Concept drift based on subspace learning detects if concept drift occurs in a new data stream by using projection variance and projection cosine, which saves data processing time and avoids the separation between the concept drifting detection and data analysis.

IV. EXPERIMENT

In this section, we use KDD data set to implement our experiments. To illustrate the advantage of concept drift based on subspace learning in aspects of accuracy and efficiency, we compare it with No-Detection method and concept drifting method which use KL distance to judge if concept drift occurs.

KDD data set contain 4 kinds of attacks: Prob, DoS, U2R and R2L. KDD data set is classified to 2 classes, one is labeled , and the other is unlabeled. Every labeled data has 41 properties and 1 tag. Tag indicates the name of attack. Every property is consecutive, discrete or string. Most classifiers can not deal with these kinds of data. Therefore, preprocessing of data is necessary for classification. The preprocessing includes 2 steps: first, translate the value of property to number; second, describe the data. We classify all data to 2 classes for convenience. 1 indicates "Normal", while 2 indicates "Attack" (e.g., Buffer-Overflow, guess-passwd and so on). Namely, we class all the kinds of attacks as "Attack". Then we get the problem of binary classification. Symbol features, such as protocol type(3 different symbols), service(70 different symbols), identifier(11 different symbols) translate to N numbers whose range is from 0 to $N-1$. N is the number of symbols. we divide the data set to many data stream windows. Each window has 1000 samples.

This paper uses support vector machine (SVM) [36] as the classifier to train and test the data set. For the No-Detection method, we treat the first data stream as training set to test all the following data streams. Fig.1(a) presents the accuracies of No-Detection method and Subspace-Based method. Fig.1(b) describes the accuracies of No-Detection method and KL-Measure method. Obviously, concept drift increase the testing accuracy.

We also analyze the mean value of accuracy and the mean value of consuming time. The results are shown in TABLE I. The subspace based concept drifting detection and the method which uses KL distance to measure concept drift promote the accuracy almost at the same level, but the former show a great advantage of consuming time, which is due to that the subspace based method not only use the projectional matrix to detect concept drift, but also reduce the dimensionality greatly. The low dimensional data improve the efficiency of classifier.

V. CONCLUSION

In this paper, we use the subspace based concept drifting method to detect the intrusion in the network, and design the judgments for this problem. Compared with other methods, subspace based one shows advantages of accuracy and consuming time for network intrusion detection. Subspace learning method detects concept drift and reduces the dimensionality at the same time, which makes it work more efficient. It is more fit for these tasks, such as IDS.

TABLE I. ACCURACY AND CONSUMING TIME

	Accuracy	*Time*
No-Detection	95.0190%	0.0868s
Subspace-Based	96.9270%	0.0219s
KL-Measure	97.0350%	0.2299s

REFERENCES

[1] Y. Zhang and W. Lee, “Intrusion detection in wirelessad hoc networks,” in Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom’00), pp. 275–283, 2000.

[2] A. Chaudhary, A. Kumar, and V. N. Tiwari, “A reliable solution against packet dropping attack due to malicious nodes using fuzzy logic in MANETs,” in International Conference on Optimization, Reliabilty, and Information Technology (ICROIT’14), pp. 178–181, 2014.

[3] A. Chaudhary, V. N. Tiwari, and A. Kumar, “Analysis of fuzzy logic based intrusion detection systems in mobile ad hoc networks,” BVICAM’s International Journal of Information Technology, vol. 6, no. 1, pp. 690–696, 2014.

[4] S. Sen and J. A. Clark, Guide to Wireless Ad Hoc Networks: Chap. 17, Intrusion Detection in Mobile Ad Hoc Networks, pp. 427–454, Springer, 2009.

[5] A. Tsymbal, “The problem of concept drift: Definitions and related work,” Technical Report TCD-CS-2004-15, Department of Computer Science, Trinity College Dublin, Ireland, 2004.

[6] G. Widmer and M. Kubat, “Learning in the presence of concept drift and hidden contexts,” Machine Learning, vol. 23(1), pp.69–101, 1996.

[7] H. Borchani, P. Larranaga, J. Gama and C. Bielza, “Mining multi-dimensional concept-drifting data streams using Bayesian network classifiers,” Intelligent Data Analysis, vol. 20.2, pp. 257-280, 2006.

[8] C.C. Aggarwal, “Data Streams: Models and Algorithms,” Springer, 2007.

[9] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby and R. Gavalda, “New ensemble methods for evolving data streams,” in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 139–148, 2009.

[10] M.M. Gaber, “Advances in data stream mining,” Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 2(1), pp. 79–85, 2012.

[11] J. Gama, “Knowledge Discovery from Data Streams,” Data Mining and Knowledge Discovery Series, Chapman & Hall CRC, 2010.

[12] J. Gao, B. Ding, W. Fan, J. Han and P.S. Yu, “Classifying data streams with skewed class distributions and concept drifts,” IEEE Internet Computing, vol. 12(6), pp. 37–49, 2008.

[13] Z. Ouyang, Y. Gao, Z. Zhao, and T. Wang, “Study on the classification of data streams with concept drift,” In: 2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), vol. 3, pp. 1673–1677, 2011.

[14] J. Gama and G. Castillo, “Learning with local drift detection,” Advanced Data Mining and Applications 4093, pp. 42–55, 2006.

[15] E. Ikonomovska, J. Gama, R. Sebasti, and D. Gjorgjevik, "Regression trees from data streams with drift detection," Discovery Science 5808, pp. 121–135, 2009.

[16] K.O. Stanley, "Learning concept drift with a committee of decision trees," Informe tcnico: UT-AI-TR-03-302, Department of Computer Sciences. University of Texas at Austin, USA , 2003.

[17] R. Klinkenberg, "Using labeled and unlabeled data to learn drifting concepts," In: Workshop notes of the IJCAI-01 Workshop on Learning from Temporal and Spatial Data, pp. 16–24, 2001.

[18] P. Lindstrom, S.J. Delany and B.M. Namee, "Handling concept drift in a text data stream constrained by high labelling cost," In: FLAIRS Conference, 2010.

[19] A. Dries, and U. Ruckert, "Adaptive Concept Drift Detection," Analysis and Data Mining, Vol. 2(5-6), pp. 311–327, 2009.

[20] G.J. Ross, N.M. Adams, D.K. Tasoulisand and D.J. Hand, "Exponentially weighted moving average charts for detecting concept drift," Pattern Recognition Letters, Vol. 33(2), pp. 91–198, 2012.

[21] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," Advances in Artificial Intelligence(SBIA) 3171, pp. 286–295, 2004.

[22] M. Baena-Garcira, J. del Campo-Avila, R. Fidalgo, A. Bifet, R. Gavalda, and R. Morales-Bueno, "Early drift detection method," Knowledge Discovery from Data Streams, 2006.

[23] P. Sobhani and H. Beigy, "New drift detection method for data streams," Adaptive and Intelligent Systems 6943, pp. 88–97, 2011.

[24] X. Wu, P. Li, X. Hu, "Learning from concept drifting data streams with unlabeled data," Neurocomputing, Vol. 92, pp. 145–155, 2012.

[25] I. Zliobaite, "Change with delayed labeling when is it detectable," In: Data Mining Workshops (ICDMW), pp. 843–850, 2010.

[26] D. Kifer, S. Ben-David and J. Gehrke, "Detecting change in data streams," Proceedings of the Thirtieth international conference on Very large data bases-Volume 30. VLDB Endowment, pp. 180-191, 2004.

[27] M.M. Lazarescu and S. Venkatesh, "Using selective memory to track concept drift effectively," Intelligent Systems and Control, pp. 388, 2003.

[28] T. Dasu, S. Krishnan, S. Venkatasubramanian, and K. Yi, "An information-theoretic approach to detecting changes in multi-dimensional data streams," In: Proc. Symp. on the Interface of Statistics, Computing Science, and Applications, 2006.

[29] S.T. Roweis, L.K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," Science, Vol. 290, pp. 2323–2326, 2000).

[30] Z. Zhang, and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," SIAM Journal of Scientific Computing, Vol. 26(1), pp. 313–338, 2004.

[31] X. He, D. Cai, S. Yan, and H. Zhang, "Neighborhood preserving embedding," In: 2010 The 2nd IEEE International Conference on Information Management and Engineering (ICIME), pp. 1208–1213, 2005.

[32] Y. Pang, L. Zhang, Z. Liu, N. Yu, and H. Li, "Neighborhood preserving projections(NPP): a novel linear dimension reduction method," Advances in Intelligent Computing 3644, pp. 117–125, 2005.

[33] W. Min, K. Lu, and X. He, "Locality pursuit embedding," Pattern Recognition vol. 37(4), pp. 781–788, 2004.

[34] W. Zheng, L. Zhao and C. Zou, "An efficient algorithm to solve the small sample size problem for LDA," Pattern Recognition, vol. 37(5), pp. 1077–1079, 2004.

[35] L. Feng, S. Liu, Y. Xiao, and J. Wang, "Subspace Detection on Concept Drifting Data Stream." Proceedings of ELM-2014, Springer International Publishing, Vol. 1, pp. 51-59, 2015.

[36] C. Cortes and V. Vapnik, "Support vector networks," Machine Learning, Vol. 20(3), pp. 273–297, 1995.

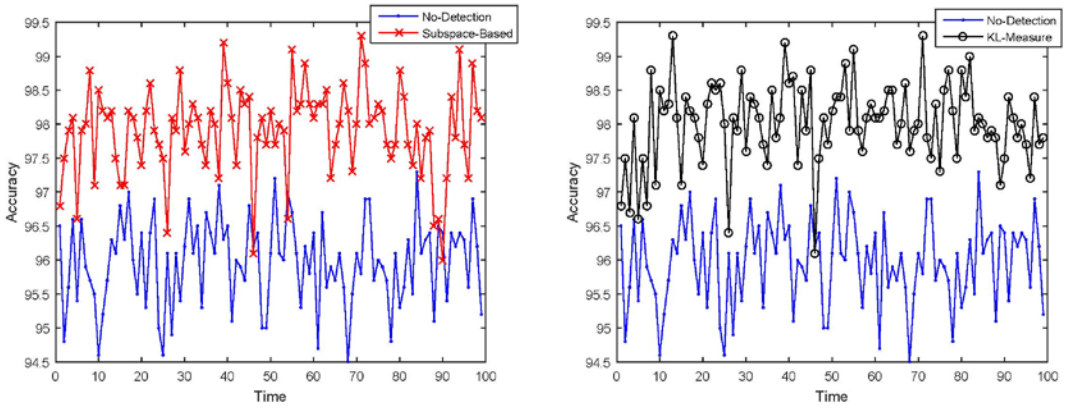

Figure 2. Experimental results of concept drift: (a) Subspace-Based method and No-Detection method. (b) KL-Measure method and No-Detection method.