Impact of PDS Based kNN Classifiers on Kyoto Dataset

Kailasam Swathi, NRI Institute of Technology, Agiripalli, India
https://orcid.org/0000-0001-7212-078X

Bobba Basaveswara Rao, Acharya Nagarjuna University, Guntur, India

ABSTRACT

This article compares the performance of different Partial Distance Search-based (PDS) kNN classifiers on a benchmark Kyoto 2006+ dataset for Network Intrusion Detection Systems (NIDS). These PDS classifiers are named based on features indexing. They are: i) Simple PDS kNN, the features are not indexed (SPDS), ii) Variance indexing based kNN (VIPDS), the features are indexed by the variance of the features, and iii) Correlation coefficient indexing-based kNN (CIPDS), the features are indexed by the correlation coefficient of the features with a class label. For comparative study between these classifiers, the computational time and accuracy are considered performance measures. After the experimental study, it is observed that the CIPDS gives better performance in terms of computational time whereas VIPDS shows better accuracy, but not much significant difference when compared with CIPDS. The study suggests to adopt CIPDS when class labels were available without any ambiguity, otherwise it suggested the adoption of VIPDS.

KEYWORDS

kNN Classification, Kyoto Dataset, Network Intrusion Detection, Network Security, Partial Distance Search (PDS), Variance Indexing

1. INTRODUCTION

Network Intrusion refers to a number of techniques that allows the malicious users to penetrate into the computer networks and exploit the computing and network resources. Network Intrusion Detection System (NIDS) is a technology that uses network intrusion datasets and identifies the intruders by applying machine learning strategies on these datasets to detect malicious activities. A network intrusion dataset is a collection of network traces i.e., traffic captures from network for a period of time.

The quality and quantity of network datasets will aid machine learning strategies to build heuristic systems for given real-world problems. These heuristic systems will help the decision makers to ever cure risk. Early detection of intrusion helps in control and prevention of malicious activities in a system.

Machine learning algorithms are heuristic approaches to solve complicated problems for which a human designer unable to define the appropriate rules in an explicit form. It is very difficult to construct an efficient real-time NIDS especially for high speed network traffics.

To build such an ideal solution and evaluation of the same, different kinds of datasets are made available for researchers. One such detection system is Kyoto 2006+ which is a real-world data set and is nearer to the current network problems. This dataset is provided with class label hence

DOI: 10.4018/IJRSDA.2019040105

supervised learning algorithms were preferred for attack predictions. In general attack and normal are class labels of these intrusion data sets.

Intrusion detection techniques availing these datasets with class labels and exhibits good results by using machine learning methodologies such as Support Vector Machines (SVM), k-Nearest Neighbor (kNN), Bayes Networks and Decision Tree Inductions etc.

Even though the kNN classifier is a lazy learning algorithm, it is used by huge number of researchers because of its good accuracy rates. Researchers are trying to minimize the classifier complexity as well as classification times of kNN algorithm while maintaining the accuracy rate high.

Partial Distance Search (PDS): is one form of kNN classification approach that makes the classifier faster when compared with general kNN classification algorithm (Basaveswara & Swathi, 2017; Eid et al., 2013). In this approach the distance computation procedure (between a known sample and a test sample/new request) will be terminated at a specific feature value without computing all feature values whenever the distance is larger than the precomputed/stored least k nearest distances, otherwise this distance will be added to the previous k nearest distances by replacing the k^th^ distance. In this approach most of the training samples are discarded quickly that reduced the computational cost of the classifier. Especially when the sample data set is very large such as KDD cup'99 and Kyoto 2006+, PDS approach yields less computational time.

PDS kNN Algorithm:

Compute first k squared distances vector D = (d_1,d_2, …, d_k) among the first k sample vectors (y_1, y_2, …, y_k) in sample set *S* where s is the sample size and *n* is total number of features for each sample belongs to *S* with the input vector *x* for which class label need to be predicted and $d_i = d^2(x, y_i)$, *i*=1,2,3...*k*

i.e., $$d^2\left(x,y_i\right),=\sum_{j=1}^{n}\left(x_j-y_{i_j}\right)^2$$

```
Step 2:   Place these first $k$ distances into vector D in ascending order i.e., $d_1 \le d_2 \le .. \le d_k$.
Step 3:   for $t$ in range of $(k+1, s)$:
Step 3.1:     Calculate the distance $d_t$ between $y_t$ and $x$ as follows:
Step 3.2:     set $d_t = 0$
Step 3.3:     for $p$ in range $(1, n)$:
Step 3.3.1:   Compute $d_t += \left(x_p - y_{t,p}\right)^2$
Step 3.3.2:   If $d_t > d_k$ then go to Step 3.
Step 3.4:     set D by replacing $d_k = d_t$ and reorder the vector D in ascending order.
```

By observing the previous works done by the various researchers, it is noticed that PDS kNN classifier is not applied on Kyoto2006+ data set. Out of our knowledge there is no work done on earlier to apply partial distance search techniques for NIDS with kNN classifiers. Most of the research works were done on increasing the accuracy but a few were concentrated on reducing computational time. This work is an attempt to explore the performance of the various PDS based kNN classifiers for NIDS and measure the computational time along with accuracy.

To achieve this objective, the variance and correlation coefficient-based feature indexing methods applied with PDS kNN classifier and compared with traditional kNN classifier.

The remaining part of this paper is organized as specified below. Section 2 summaries the recent research works tin this area. The detailed description about the Kyoto 2006+, a benchmark dataset is provided in section 3; the experimental methodology is presented in Section 4. In Section 5, results are given. Finally Section 6 presented conclusion and future scope of this work.

2. RELATED WORK

Several researchers are working on network security and intrusion detection systems by applying several machine learning techniques. While some of them are using benchmark data sets, others are generating their own data sets from real-time networks traffics. This section presents recent research works of some of these researchers.

Adel Ammar et al. (2015) has proposed and compared feature reduction-based classification algorithms for network traffic. Authors addressed the improvement in classification accuracy. Hoque (2012) Suggested a genetic algorithm-based IDS to detect network intrusions efficiently. The implementation of proposed genetic algorithm is done on KDD99 benchmark dataset and identified a False Positive rate of 0.3 and detection rate up to 95%.

The web link provided a detailed study on Kyoto 2006+ data set (2006). This article is a complete statistical analysis of the features, values and class labels of the dataset. This will help researches to understand the data set. Wei Chao Lin (2015) introduced and implemented CANN an ensemble method that combined both classification and clustering to increase the accuracy and to achieve higher detection rates too.

Solane Duquea et al. (2015) has described the implementation of nonparametric, semi-supervised learning approaches and compares the performance with other model using feature-based data derived from an operational network that addresses network intrusion problem. Om et al. (2012) has proposed a hybrid NIDS that combines the qualities of both misuse and anomaly detection systems. Further k-means algorithm for clustering was applied to minimize false alarm rate. A combined k-nearest neighbor and naïve Bayesian classifier algorithms were combined as a hybrid classifier for the detection of intrusions. The experiment was done on KDD99 data set and achieved up to 99.0% of accuracy.

Sallay et al. (2013) presented combination of cluster centers and nearest neighbors as a novel feature representation approach for effective and competent intrusion detection (CANN). The performance of this algorithm is better than kNN and SVM classifiers and takes less computational time. CANN failed to detect U2L and R2L attacks effectively. For the implementation of CANN, KDD99 data set was used (Table 1).

3. KYOTO 2006+ DATASET

The Kyoto 2006+ (2006) is a real network traffic dataset captured from honeypots. The dataset was obtained from November 2006 to August 2009. The data collected not undergone to any further modifications or removal. It consists of recent network attacks distinguished from normal traffic using honeypots. Kyoto dataset is available with 24 features, among these 24 features 14 were derived from KDDCUP'99 dataset (Song et al., 2011; Tayallaee et al., 2009), and further 10 more features were added that can be helpful in detecting the kind of attacks more effectively in the network (Song et al., 2011).

In the proposed methodology, implementation is taken place only on 18 input features out of 24 features, three of them are prediction labels those are IDS_detection, Malaware_detection and Ashula_detection, these features indicate the type of attack and are redundant to the class label, and IP_source and IP_destination are IP addresses of source and destination machines These two attributes have extremely large number of distinct values and cannot be apply discretization on IP addresses, the start_time is another attribute and is also contains large number of distinct values, due to these reasons these 6 features are discarded. The resultant 18 input features are used in this approach are summarized table 2.

The class label of Kyoto 2006+ dataset is a three valued feature. These three values are "1", "−1" and "-2" representing normal request, known attack, and unknown attack respectively. However, since the presence of the unknown attacks ie., with label "-2" in the database is of 0.7% which is very less

Table 1. A summary of authors' works on NIDS with datasets and classifiers they have used

Author	Dataset used	Technique	Problem domain	Evaluation Method	Feature Selection
Basaveswara (2017)	KDDCUP 99	IKPDS	Anomaly detection	Accuracy, Computational time	No
Basaveswara (2016)	NSL-KDD	IKPDS	Anomaly detection	Accuracy, Fitness value curve, computational time	Yes
Song (2011)	Kyoto 2006+	Statistical Analysis	Anomaly detection	Statistical analysis	No
Om et al. (2012)	KDD CUP 99	Hybrid Model	Anomaly detection	Accuracy	No
Hoque et al. (2012)	KDD CUP 99	Genetic Algorithm	Anomaly detection and reducing false positive rates	Accuracy and False Positive rate	No
Wei Chao Lin et al. (2016)	KDD CUP 99	CANN	Anomaly detection and reducing false positive rates	Accuracy, Detection rate	No
Sallay et al. (2013)	KDD CUP 99	SVM, CANN	Anomaly detection	Accuracy	No

and it is difficult to detect these kind of attacks using a machine learning model. By considering the class label value same for known and unknown attacks, makes the problem a binary classification.

4. EXPERIMENTAL METHODOLOGY

The data set selected for this study is from the first 5 days of August 2009 i.e., nearly 6,35,000 records. This data has undergone a stage of data preprocessing. The data preprocessing part of the methodology contains two phases, data transformation and data normalization. Transformation: A kNN classifier requires all its features of the data set to be in numerical form. Because the feature *flag* in Kyoto 2006+ data set is of categorical, it is transformed into numeric by scalar value, where a distance of zero is assigned if the values are identical; otherwise, the distance is one. For example, the flag value RSTOS0 is converted as value 3.0. The Figure 1 shows the original 3 samples of Kyoto 2006+ dataset, Figure 2 and Figure 3 represent the same 3 samples present in Figure 1 after implementing data transformation of categorical values in to numerical values and data normalization process using min-max normalization respectively.

Normalization: Normalization of NIDS dataset features is necessary to avoid feature influence on distance measures. Kyoto 2006+ dataset features exhibit different characteristics of the network and these values are of both qualitative and quantitative with different ranges. Due to these feature ranges the feature values may influence the classification process especially measuring the distance. The feature with a very high value may suppress the feature with less value. To avoid this kind of dominance among features it is necessary to normalize the features using various scaling techniques.

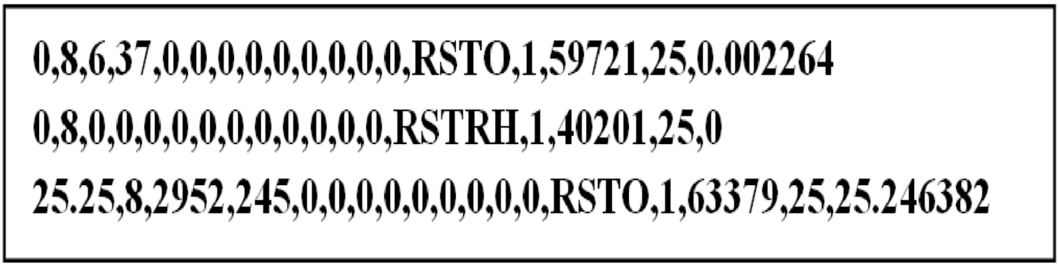
In the present study, to normalize features, the min-max normalization technique is used. The formula for the min-max normalization is given as below.

$$v' = \frac{v - \max A}{\max A - \min A}\left(newMaxA - newMinA\right) + newMinA$$

Table 2. Selected features from Kyoto 2006+ dataset

S.No	Feature #	Feature Name	Feature Description
1	1	Duration	Connection duration in seconds
2	2	Service	Type of service used i.e., http, telnet
3	3	source_bytes	number of bytes of data that is sent by source
4	4	destination_bytes	number of bytes of data that is sent by destination
5	5	Count	The count of connections established among same source and destination
6	6	same_srv_rate	The percentage of the connections that are requesting same service
7	7	serror_rate	The total connections that have "SYN" errors percentage
8	8	srv_serror_rate	The percentage of connections that have "SYN" errors in Srv count
9	9	dst_host_count	Count of IPs whose source and destination are the same to that of the current connection out of last 100 connections
10	10	dst_host_srv_count	Count of IPs whose destination and service type are the same to that of the current connection among last 100 connections.
11	11	dst_host_same_ src_port_rate	Total percentage of connections with source port is same as connection in feature# 9
12	12	dst_host_serror_ rate	The percentage of connections whose feature#9 have "SYN" errors.
13	13	dst_host_srv_ serror_rate	The percentage of connections whose feature#10 have "SYN" errors.
14	14	Flag	Connection State
15	15	Label	'1' normal request, '-1' for known attack, and '-2' for unknown attacks
16	16	source_port_ number	Session's source port number
17	17	destination_port_ number	Session's destination port number
18	18	Duration1	Duration of the session

Figure 1. Three original samples of Kyoto dataset

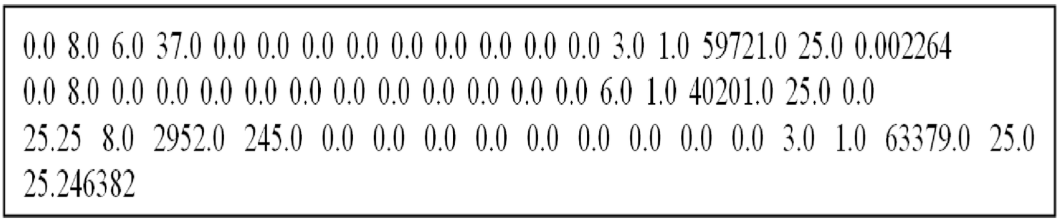


```
0,8,6,37,0,0,0,0,0,0,0,0,RSTO,1,59721,25,0.002264
0,8,0,0,0,0,0,0,0,0,0,0,RSTRH,1,40201,25,0
25.25,8,2952,245,0,0,0,0,0,0,0,0,RSTO,1,63379,25,25.246382
```

Where *A* is an attribute, *v* is a value of attribute *A* that should be normalized, *v'* is the new normalized value of *v*, *minA and maxA* are minimum and maximum values of attribute *A* and *newMinA and newMaxA* are minimum and maximum values into which the value of *v* needs to be normalized.

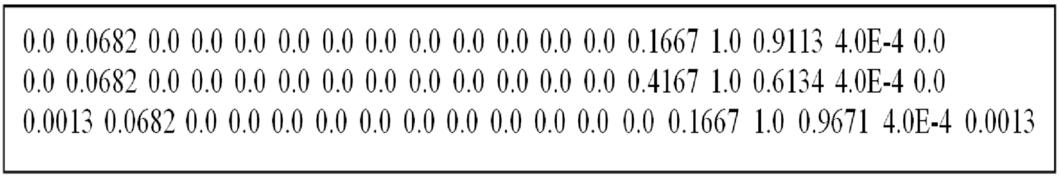
After preprocessing phase, to apply these PDS kNN classifiers the feature is rearranged as without indexed for SPDS, variance indexed for VIPDS and correlation indexed for CIPDS.

Figure 2. Three data samples after transformation



```
0.0 8.0 6.0 37.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 3.0 1.0 59721.0 25.0 0.002264
0.0 8.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 6.0 1.0 40201.0 25.0 0.0
25.25 8.0 2952.0 245.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 3.0 1.0 63379.0 25.0
25.246382
```

Figure 3. Three data samples after scaling



```
0.0 0.0682 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.1667 1.0 0.9113 4.0E-4 0.0
0.0 0.0682 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.4167 1.0 0.6134 4.0E-4 0.0
0.0013 0.0682 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.1667 1.0 0.9671 4.0E-4 0.0013
```

Variance Indexing: Indexed based kNN classification using PDS method (Basaveswara & Swathi, 2017; Oiao et al., 2004) is an improvement on PDS algorithm where all these features are ordered based on their variance indexing. Highest indexed feature will be placed first (Song et al., 2011). In this approach, highest variance feature be computed for the distance measure which further reduces the computational time of the classification as highly varied feature will increases the distance which will become greater than the k nearest distances so that the training sample can be discarded early. In this paper, the following variance *(v)* formula is used.

$$v = \frac{\sum_{i=1}^{n}\left(x_{i} - \overline{x}\right)^{2}}{n-1}$$

Where x_i is the i^{th} value of the feature *x*, $\overline{x}$ is the mean of the feature *x* and n is the number of samples of dataset. Table 3 shows the list of features of their variance ordering along with variance values.

Correlation Coefficient indexing: this method uses correlation coefficient between each feature with the class label of the data set. This technique is class label dependent whereas variance indexing is a class label independent.

To measure the linear dependence between any two attributes correlation coefficient measure is the basic and most popular linear correlation methods (Amiri et al., 2011; Nsl-kdd data set, 2009). This measure is used in many research areas because of its simplicity and ease of estimation. For any two random variables, their correlation coefficient specifies the magnitude of the relationship between them. In this paper, correlation coefficient is implemented to measure the relationship between each feature with the class label feature. All these features are then ordered according to their correlation coefficient values in descending order. The following is the correlation coefficient formula implemented.

$$r = \frac{n\left(\sum XY\right) - \left(\sum X\right)\left(\sum Y\right)}{\sqrt{\left[n\sum X^2 - \left(X\right)^2\right]\left[n\sum Y^2 - \left(\sum Y\right)^2\right]}}$$

Where Y is the class label and X is one of the features of the given dataset and n is the number of data samples that are available. Table 4 shows the features after their correlation coefficient ordering along with correlation coefficient values.

The preprocessed dataset is rearranged by their feature indexing based on its feature variances in variance indexing and by their feature indexing based on its correlation values in correlation indexing. After indexing, partial distance kNN classifier is applied on these feature-indexed reordered data sets. Results are discussed along with traditional kNN classification as well as SPDS kNN classification in the following section.

5. RESULTS

All these four kNN classifiers are implemented on processor-Intel core i5 with 4 GB RAM, in eclipse IDE with Java 1.6. In this experimental procedure, the value of k of kNN classifier is taken as 10. The performance measure, accuracy is calculated on three PDS based classifiers and traditional kNN classifier also. The following is the formula for measuring accuracy.

Table 3. Ordered features based on variance indexing with rank

S. No	Feature #	Variance Rank	Feature Name
1	6	0.245	same_srv_rate
2	2	0.210	service
3	10	0.227	dst_host_srv_count
4	8	0.164	srv_serror_rate
5	13	0.101	dst_host_srv_serror_rate
6	16	0.090	source_port_number
7	14	0.085	flag
8	11	0.037	dst_host_same_src_port_rate
9	9	0.015	dst_host_count
10	5	0.009	count
11	17	0.008	destination_port_number
12	7	0.004	serror_rate
13	12	0.001	dst_host_serror_rate
14	4	2.73E-5	destination_bytes
15	3	1.80E-5	source_bytes
16	18	1.77E-5	Duration1
17	1	1.77E-5	Duration

Table 4. Ordered features based on correlation coefficient indexing with rank

S. No	Feature #	Correlation coefficient Rank	Feature Name
1	10	0.7459	dst_host_srv_count
2	6	0.4406	same_srv_rate
3	5	0.3664	count
	16	0.3108	source_port_number
4	14	0.3640	flag
5	11	0.0787	dst_host_same_src_port_rate
6	1	0.1167	duration
7	18	0.1167	duration1
8	4	0.0035	destination_bytes
9	3	0.0040	source_bytes
10	12	-0.0360	dst_host_serror_rate
11	7	-0.0447	serror_rate
12	17	-0.1603	destination_port_number
13	9	-0.3688	dst_host_count
14	13	-0.2264	dst_host_srv_serror_rate
15	8	-0.4850	srv_serror_rate
16	2	-0.8438	Service

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Where TP=True Positive (Total number of normal test samples predicted as normal)

TN= True Negative (Total number of attack test samples predicted as attack)
FP= False Positive (Total number of attack test samples predicted as normal)
FN= False Negative (Total number of normal test samples predicted as attack)

The computational time of these four classifiers is also measured. The following Table 5, Figure 4 and Figure 5 present the differences between these classifiers in terms of accuracy and computational time.

The accuracy for traditional kNN is 99.21, the same accuracy is exhibited by SPDS and CIPDS whereas the VIPDS exhibits little bit of more accuracy i.e., 0.07%. The traditional kNN classifier takes more computational time than PDS kNN classifiers. Subsequently it is observed that

SPDS takes highest computational time than two-feature indexing based classifiers VIPDS and CIPDS. The computational time difference between VIPDS and CIPDS is 7 minutes and the accuracy difference is 0.07%.

Table 5. Shows accuracies, Classification time (min)for four classifiers

Classifier Measures	kNN	SPDS	VIPDS	CIPDS
Accuracy (%)	99.21	99.21	99.28	99.21
Computational time (min)	306.41	238.87	202.67	195.05

Figure 4. Computational time (min) for four classifiers

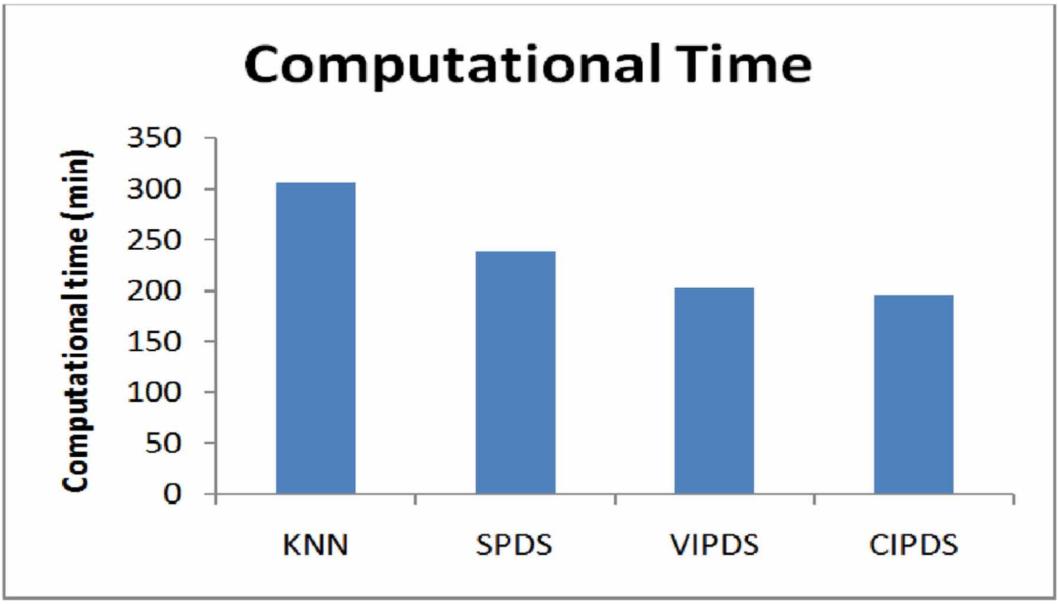
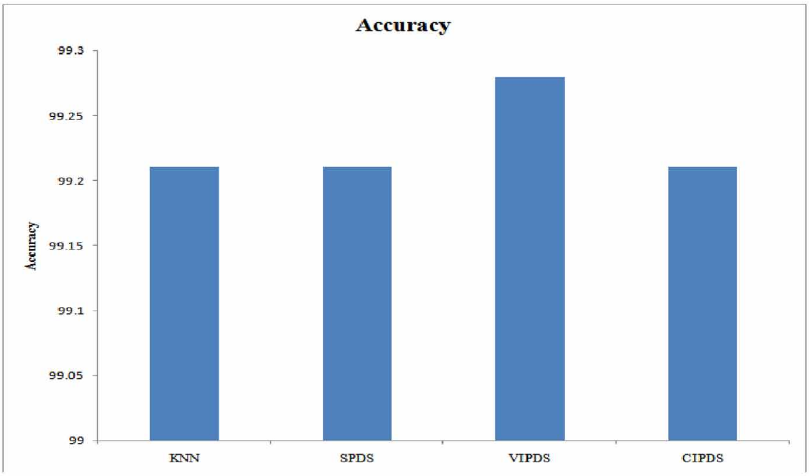


Figure 5. Accuracies for four classifiers



6. CONCLUSION

This paper investigated three PDS based kNN classifiers with the traditional kNN classifier with accuracy and computational time as performance measures on Kyoto 2006+ dataset.

Any novel defense mechanism suggests for NIDS, need to show substantial improvement in accuracy and simultaneously with minimal computational time. After examining these results authors suggest both feature indexing PDS methods VIPDS and CIPDS as suitable defense mechanisms for NIDS.

It is observed that CIPDS consumes less computational time with high accuracy. These results are encouraging PDS methods, for better classification time with some preorder methodologies. Since early detection of intrusion is more important parameter for a good defense mechanism. From this study, it is concluded to adopt CIPDS when class labels are available without any ambiguity, otherwise, it is suggested to adopt VIPDS.

The future scope for this work is to apply other feature indexing procedures. The feature selection may be done with these feature indexing procedures.

REFERENCES

Aissa, N. B., & Guerroumi, M. (2016). Semi-supervised Statistical Approach for Network Anomaly Detection. *Procedia Computer Science*, *83*, 1090–1095. doi:10.1016/j.procs.2016.04.228

Amiri, F., Rezaei Yousefi, M. M., Lucas, C., Shakery, A., & Yazdani, N. (2011). Mutual information-based feature selection for intrusion detection systems. *Journal of Network and Computer Applications*, *34*(4), 1184–1199. doi:10.1016/j.jnca.2011.01.002

Ammar, A. (2015). comparison of feature reduction techniques for the binomial classification of network traffic. *Journal of Data Analysis and Information Processing*, *3*(02), 11–19. doi:10.4236/jdaip.2015.32002

Basaveswara, R., & Swathi, K. (2016). Variance-Index Based Feature Selection Algorithm for Network Intrusion Detection. *International Journal of Scientific Research-Journal of Computer Engineering (IOSR-JCE), 18*(4), 1-11.

Basaveswara, R., & Swathi, K. (2017). Fast kNN Classifiers for Network Intrusion Detection System. *Indian Journal of Science and Technology*, *10*(14), 1–10.

Duque, S., & bin Omar, M. N. (2015). Using Data Mining Algorithms for Developing a Model for Intrusion Detection System (IDS). Procedia Computer Science, 61, 46–51. 10.1016/j.procs.2015.09.145

Eid, H. F., Hassanien, A. E., Kim, T.-h., & Banerjee, S. (2013). Linear Correlation-Based Feature Selection for Network Intrusion Detection Model. In *Advances in Security of Information and Communication Networks* (pp. 240–248). Springer; . doi:10.1007/978-3-642-40597-6_21

Hoque, M. S., Mukit, M., Bikas, M., & Naser, A. (2012). An implementation of intrusion detection system using genetic algorithm.

Hwang, W. J., & Wen, K. W. (1998). Fast kNN classification algorithm based on partial distance search. *Electronics Letters*, *34*(21), 2062–2063. doi:10.1049/el:19981427

Kang, S.-H., & Kim, K. J. (2016). A feature selection approach to find optimal feature subsets for the network intrusion detection system. *Cluster Computing*, *19*(1), 325–333. doi:10.1007/s10586-015-0527-8

Kyoto 2006+ dataset. (n.d.). Retrieved from http://www.takakura.com/Kyoto_data/

Lin, W.-C., Ke, S.-W., & Tsai, C.-F. (2015). CANN: An intrusion detection system based on combining cluster centers and nearest neighbors. *Knowledge-Based Systems*, *78*, 13–21. doi:10.1016/j.knosys.2015.01.009

Om, H., & Kundu, A. (2012). A hybrid system for reducing the false alarm rate of anomaly intrusion detection system. In *Proceedings of the 2012 1st International Conference on Recent Advances in Information Technology (RAIT)*. IEEE; . doi:10.1109/RAIT.2012.6194493

Qiao, Y. L., Pan, J. S., & Sun, S. H. Improved partial distance search for k nearest-neighbor classification. In *Proceedings of the* 2004 *IEEE International Conference on Multimedia and Expo ICME'04* (pp. 1275-1278). IEEE.

Sallay, H., Ammar, A., Saad, M. B., & Bourouis, S. (2013, August). A real time adaptive intrusion detection alert classifier for high speed networks. In *Proceedings of the 2013 IEEE 12th International Symposium on Network Computing and Applications* (pp. 73-80). IEEE. 10.1109/NCA.2013.16

Shazzad, K. M., & Park, J. S. (2005). Optimization of intrusion detection through fast hybrid feature selection. In *Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies PDCAT 2005*. IEEE; . doi:10.1109/PDCAT.2005.181

Song, J., Takakura, H., Okabe, Y., Eto, M., Inoue, D., & Nakao, K. (2011). Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation. In *Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security*. ACM; . doi:10.1145/1978672.1978676

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A Detailed Analysis of the KDD CUP 99 Data Set. In *Proceedings of the 2009 IEEE symposium on computational intelligence in Security and Defense Applications (CISDA 2009)*. IEEE; Retrieved from http://nsl.cs.unb.ca/KDD/NSLKDD.html

Wei, C. L., Ke, S. W., & Tsai, C. F. (2015). CANN: An intrusion detection system based on combining cluster centers and nearest neighbours. *Knowledge-Based Systems*, *78*, 13–21. doi:10.1016/j.knosys.2015.01.009

Bobba B Rao has completed his Ph.D. at ANU. He is a Research Director at Acharya Nagarjuna University, Guntur, India, and guiding about 10 scholars, and from among them 3 were awarded.