**ISSN** 1746-7659, England, UK
**Journal of Information and Computing Science**
Vol. 9, No. 1, 2014, *pp*. 037-048

# A Novel Data Mining based Hybrid Intrusion Detection Framework

Mradul Dhakar[+] and Akhilesh Tiwari

Department of CSE & IT, Madhav Institute of Technology and Science,
Gwalior (M.P.), India

(*Received June 23, 2012, accepted October 12, 2013*)

**Abstract.** The prosperity of technology worldwide has made the concerns of security tend to increase rapidly. The enormous usage of internetworking has raised the need of protecting system(s) as well as network(s) from the unauthorized access (intrusion). To tackle the intrusive activities, several countermeasures have been found in literature viz. firewall, antivirus and currently widely preferred Intrusion detection System (IDS). IDS, is a detection mechanism for detecting the intrusive activities hidden among the normal activities. The revolutionary establishment of IDS has attracted analysts to work dedicatedly enabling the system to deal with technological advancements. Hence in this regard, various beneficial schemes and models have been proposed in order to achieve enhanced IDS. This paper proposes a novel hybrid model for intrusion detection. The proposed framework in this paper may be expected as another step towards advancement of IDS. The framework utilizes the crucial data mining classification algorithms beneficial for intrusion detection. The Hybrid framework would henceforth, will lead to effective, adaptive and intelligent intrusion detection.

**Keywords:** Data Mining, Intrusion Detection, Classification, K2, TAN, REP, KDDCup'99

## 1. Introduction

The progressive use of intrusion detection system for handling the abnormalities on web has caused multiple efforts laid by the analysts. The intrusions have been found dominating the internet which may be assumed as a threat to the security of genuine users. In order to meet the advancement of changing technological world, IDS has been through various alterations where it has been competent in detecting these intrusions more precisely.

Though IDS itself is a standalone definition but in order to make the system cope with the recent technological developments and increased intrusions strategies, it has gone through several revisions where it has beneficially used the various research fields such neural network, statistics and recently data mining. The key ideas are to use data mining techniques to discover consistent and useful patterns of system features that describe program and user behavior, and use the set of relevant system features to compute (inductively learned) classifies that can recognize anomalies and known intrusions [1]. This thought has made IDS with data mining serve as the most promising intrusion detection scheme where Data Mining identifies trends within data that go beyond simple analysis [2].

Generally IDSs are deployed to monitor a system or a network in search of any abnormal condition. In this surveillance if any kind of intrusive attempt is detected, the monitoring system i.e. IDS sets up an alarm which is an indication of the presence of intrusion. In order to detect intrusions in an efficient manner, various appreciable models have registered their presence in the literature. The presently available models involve usage of various novel algorithms which are likely to detect these intrusions distinguishably.

Among these, algorithms based on data mining have been a point of attraction for researchers because of their extensive feasibility in detecting intrusions. These algorithms aid in improving accuracy of the system along with effective detection rate and less false alarm rate. The algorithms loyal for classification are the

[+] Corresponding author.
*E-mail address*: mraduliitm@gmail.com.

*Published by World Academic Press, World Academic Union*

most desirable algorithms for detection.

In the data mining classification techniques, Tree Augmented Naïve Bayes (TAN) and Reduced Error Pruning (REP) algorithms have come out as the most significant detection algorithms in IDS. Hence this paper presents an intelligent effort for intrusion detection which proposes a framework named Hybrid Intrusion Detection Model. This model is a combinational scheme which aims at surmounting the shortcomings faced by two algorithms individually with interestingly increased accuracy of the detection.

The paper consists of following sections: Section II is a brief description about intrusion detection system, section III discusses the intrusion detection processes involved, section IV studies about the intrusion detection approaches, section V describes about the attacks detected by IDS, section VI elaborates the proposed methodology and section VII consists the experimental analysis performed on proposed hybrid model.

## 2. Intrusion Detection System

The adverseness of abnormalities (generally referred as intrusions) on web has brought up the security concerns leading to the successful implementation of abnormalities detection system named as Intrusion Detection System (IDS). Intrusions may be defined as the unauthorized attempt for gaining access on a secured system or network. Intrusion detection is the course of action to detect suspicious activity on the network or a device. Intrusion Detection System (IDS) is an important detection used as a countermeasure to preserve data integrity and system availability from attacks [3].

The IDS has been a renowned aspect for detecting intrusions adequately. The IDS is assumed as hardware or software or combination of both that allows monitoring of the network traffic in search of intrusions. An intrusion detection system (IDS) inspects all inbound and outbound network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system [4]. It has advantageously helped the analysts to learn about the various possible attacks.

### 2.1. Intrusion Detection Process

Intrusion detection on the basis of their detection process are categorize into Misuse / Signature-based intrusion detection and Anomaly-based intrusion detection.

### *2.1.1. Misuse Detection*

Misuse detection compares the user activities to the known intruder activities on web. The idea of misuse detection is to represent attacks in the form of a pattern or a signature so that the same attack can be detected and prevented in future [5]. The IDS searches for defined signatures and if a match is found, the system generates an alarm indicting the presence of intrusion. Since it works on the basis of predefined signatures, it is unable to detect new or previously unknown intrusions.

### *2.1.2. Anomaly Detection*

Anomaly intrusion detection identifies deviations from the normal usage behavior patterns to identify the intrusion [6]. It is a technique which is based on the revealing of traffic anomalies. It estimates the deviation of a user activity from the normal behavior and if the deviation goes beyond a preset threshold, it considers that activity as an intrusion. It is because of this threshold concept anomaly can detect new intrusions in addition to the previously known intrusions. However anomaly is able to detect new intrusion but the compulsion for involvement of limiting factor results in high percentage of false positive rate.

### 2.2. Intrusion Detection Approaches

On the basis of the data analyzed and stored it is classified into Host-based Intrusion Detection System (HIDS) and Network-based Intrusion Detection System (NIDS).

### *2.2.1. Host-based Intrusion Detection System*

Host-based IDS analyze host-bound audit sources such as operating system audit trails, system logs, and application logs [7]. It is a software application which is installed onto a system in order to protect it from intruders. The audit data which is to be analyzed is collected from the host (or system) in the network. HIDS are OS dependent and thus require some prior planning before implementation and are efficient in detecting buffer overflow attacks.

### *2.2.2. Network-based Intrusion Detection System*

Network-based IDS analyze network packets that are captured on a network [7]. In NIDS, detection software is installed in a network in order to detect intrusions. NIDS collects data directly from the network in form of packets and are analyzed for detecting intrusions. It is OS independent and cannot scan protocol or content if the traffic in encrypted. It provides better security against denial of service attacks.

## 2.3. Attacks Detected by IDS

Following are the four types of attacks on ground being detected by IDS:

### *2.3.1. Denials-of-Service (DoS)*

Denials-of Service attacks have the goal of limiting or denying services provided to the user, computer or network. Attacker tries to prevent legitimate users from using a service [8]. It is usually done by making the resources either too busy or overflow, which as a consequence results in the disobedience of services requested by the legitimate users.

### *2.3.2. Probing or Surveillance*

Probing or Surveillance attacks have the goal of gaining knowledge of the existence or configuration of a computer system or network [9]. The attacker thereafter tries to harm or retrieve information about the resources of the victim network.

### *2.3.3. User-to-Root (U2R)*

User-to-Root attacks are attempts by a non-privileged user to gain administrative privileges. In the attack, users take advantage of system leak to get access to legal purview or administrator’s purview, such as: Buffer Overflow is among them [10]. The goal of gaining root or super-user access on a particular computer or a system (on which the attacker previously had user level access) is to compromise the vulnerabilities of the system.

### *2.3.4. Remote-to-Local (R2L)*

Remote-to-Local attack is the kind of intrusion attack where the remote intruder consistently sends packets to a local machine with motive to expose the machine vulnerabilities and exploit privileges which a local user would have on the computer.

# 3. Proposed Methodology

The proposed system (shown in figure 1) is a hybrid intrusion detection framework based on the combination of two classifiers i.e. Tree Augmented Naïve Bayes (TAN) and Reduced Error Pruning (REP). The TAN classifier is used as a base classifier while the REP classifier is used as a Meta classifier. The Meta classification is the learning technique which learns from the Meta data and judge the correctness of the classification of each instance by base classifier. The judgment from each classifier for each class is treated as a feature, and then builds another classifier, i.e. a meta-classifier, to make the final decision [11]. Hence it can be said that the Meta-classification re-classifies the classification judgments made by classifiers.

The working of hybrid framework can be understood in following algorithmic steps:

Step 1: Input dataset
Step 2: Perform preprocessing of the dataset
Step 3: Select TAN as the base classification algorithm
Step 4: Choose REP algorithm for Meta classification
Step 5: Perform classification on base classifier for Meta Rules

DOI https://doi.org/2024-JICS-22595 | Generated on 2025-04-05 22:12:03
OPEN ACCESS

Step 6: Set the obtained Meta rules as input for Meta classification
Step 7: Perform re-classification using Meta classifier

The main idea of using this technique is to improve the overall classification performance resulting in better outcomes than any other existing technique. The two classifiers indulged in the proposed system can be understood as:

### 3.1. Tree Augmented Naïve Bayes Algorithm

The Tree Augmented Naïve Bayes (TAN) [12, 13] is a Bayesian Network learning technique and it is the extension to simple Naïve Bayes classifier. Naive Bayes is probabilistic classifier structure based on Bayes theorem having naive (strong) independence assumptions. This structure encodes the strong conditional independence assumption among attributes i.e. the class node is the parent node for each and every attribute node with no parent node defined for it. Thus the joint probability as represented by:

$$p(c, v_1, v_2, \ldots, v_n) = p(c) \prod_i p(v_i|c)$$

The TAN network is an enhancement on Naive Bayes that relaxes the strong conditional independence assumption. It allows additional edges between the attributes of the network in order to capture correlations among them [12]. Similar to Naive Bayes, each attribute can have class node called augmenting edge pointing to it. The augmenting edges encode statistical dependencies between attributes. Thus, the joint probability in TAN depends on probabilities conditioned not only on class in-fact also on an attribute parent node $\mathrm{pa}_{v_i}$ as well.

$$p(c, v_1, v_2, \ldots, v_n) = p(c) \prod_{i=1}^{n} p(v_i|pa_{v_i}, c)$$

From these network structures and corresponding joint distributions, we can compute class predictions $\hat{C}(V)$

$$\hat{C}(V) = argmax_c P(C|V) \propto P(C) \prod_i P(V|C)$$

In Naive Bayes, the network structure is known beforehand while TAN network structures needs to be learned. The learning is done by calculating conditional mutual information function between two attributes. This function by finding the maximal weighted spanning tree in a graph is used in view to construct the maximum-likelihood tree. The following are the steps for this procedure as discussed in [13]:

1) Compute the conditional mutual information given C between each pair of distinct variables,

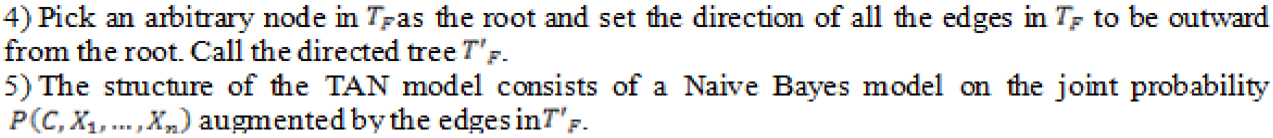
$$I(X_i;X_j|C) = \sum_{x_i, x_j, c} \tilde{P}(x_i, x_j, c) \log \frac{\tilde{P}(x_i, x_j|c)}{\tilde{P}(x_i|c)\tilde{P}(x_j|c)}$$

where $\tilde{P}(\cdot)$ is an empirical distribution (computed using the training data). Intuitively, this quantity represents the gain in information of adding $X_i$ as a parent of $X_j$ given that C is already a parent of $X_j$

2) Build a complete undirected graph on the features$X_1, \ldots, X_n$ where the weight of the edge between $X_i$ and $X_j$ is $I(X_i;X_j|C)$. Call this graph $G_F$.

3) Using Kruskal or Prim's algorithm, find a maximum weighted spanning tree on$G_F$. Call it$T_F$.

DOI https://doi.org/2024-JICS-22595 | Generated on 2025-04-05 22:12:03



4) Pick an arbitrary node in $T_F$ as the root and set the direction of all the edges in $T_F$ to be outward from the root. Call the directed tree $T'_F$.
5) The structure of the TAN model consists of a Naive Bayes model on the joint probability $P(C, X_1, \dots, X_n)$ augmented by the edges in $T'_F$.

### 3.2. Reduced Error Pruning Algorithm

REP [14, 15] is a fast decision tree learner classifier of data mining technique. It uses a validation data set for estimating generalization error. The error is pruned in the technique for each node in the tree. Essentially the node with the highest reduced error rate is pruned.

Pruning can be understood as a technique whose objective is to reduce the size of decision tree by removing parts of a tree that allow better classification of instances. Pruning therefore reduces the classification complexity with increased accuracy. The accuracy is improved by the reduction of overfitting and by removal of the parts of the tree classifier that may be based on noisy or erroneous data. Hence, a pruning of a tree is a sub tree of the original tree with just zero, one or more internal nodes changed into leaves [14].

The pruning in REP is always done at leaves where each node is replaced with its most popular class. First, the training data are split into two subsets: a growing set (usually 2/3) and a pruning set (1/3) [15]. The growing phase is used to grow the rules for constructing classification tree while pruning phase performs pruning. Next an error rate is calculated for each node, estimated as the number of instances that are misclassified on a validation (pruning) set by propagating errors upward from the leaf nodes. The difference in error rate is determined by replacing the most common class resulting from a node. Finally, if the difference is a reduction in error then the sub-tree below the node can be considered for pruning. In reference to the discussion of REP, following are the algorithmic steps undertaken.

Step 1: Select dataset
Step 2: Split the input dataset into two subsets, growing set and validation set.
Step 3: Repeat the pruning phase i.e. step 4 and 5 for every node in the tree
Step 3: Evaluate the impact on the validation set i.e. error rate for each node.
Step 5: Remove the node which maximally improves the accuracy of the validation set i.e. the node with highest reduced error rate.

The leading benefits of this classification technique are its simplicity and the speed for decision learning. However the construction of tree requires larger amount of data for pruning, REP is still assumed as more accurate classification algorithm.

## 4. Detailed Description of the Hybrid IDS Framework

This section describes about all the modules incorporated in the Hybrid IDS framework shown in fig. 1. Following is the brief discussion about each module:

### 4.1. KddCup'99 Dataset

The kddcup'99 dataset [16] is a benchmark dataset which is originated by processing the tcpdump segment of DARPA 1998 evaluation dataset. The KDDCup'99 dataset was originated by processing the tcpdump segment of DARPA 1998 evaluation dataset. The data set consists of 41 features and a separate feature (42$^{nd}$ feature) that labels the connection as 'normal' or a type of attack. The data set contains a total of 24 attack types that fall into 4 major categories (DoS, Probe, R2L and U2R) that are already discussed.

For the training and testing of the proposed framework the 10% of the KddCup'99 dataset is used as the full KddCup'99 dataset consists of 5 million instances many of them are redundant. The 10% of the KddCup'99 dataset consists of 494021 instances. In which 97278 are 'Normal' instances and remaining 396743 are belongs to any one type of attack.

### 4.2. Preprocessing

In the preprocessing module the class label presents in the 42nd feature of KddCup'99 dataset is recast into five major categories for the sake of decreasing complexity of performance evaluation of the proposed model. As the original KddCup'99 dataset having 22 types of attack labels, it was very inconvenient to assess the performance of the classification model. Hence the attack labels are modified to their respective categories for the ease of analysis. Finally five major classes are formed as the class label i.e. DoS, Probe, R2L, U2R and Normal.

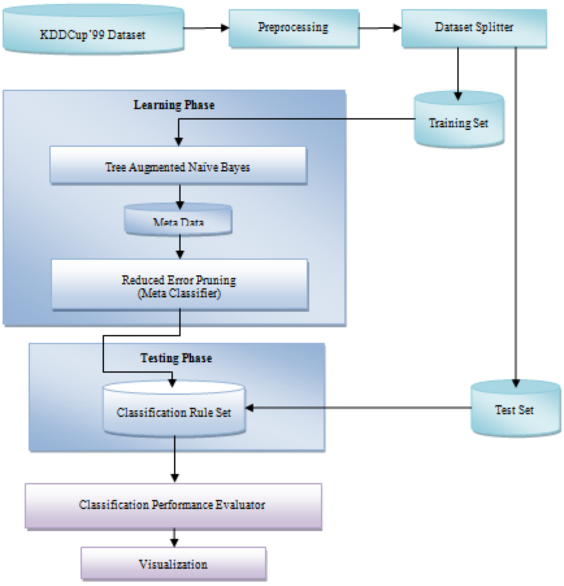


Fig. 1: Hybrid Intrusion Detection Framework

## 4.3. Dataset Splitter

The Dataset Splitter module partitions the dataset into two parts received from the preprocessing module. To partition the dataset into two parts a method named holdout is used. In this method, the given data are randomly partitioned into two independent sets, a training set and a test set [17]. The 66% of the data is allocated to the training set and the remaining 44% of the dataset is allocated to the testing set. The training set is used to derive the proposed framework while the test set is used to assess the accuracy of the derived model. When the KddCup'99 dataset passed through the data splitting module then it gets divided into the training set which consists of 326054 instances and the testing set which consists of 167967 instances.

## 4.4. Learning Phase

The learning phase involves two steps for generating the classification rules. In the first step, the learning of base classifier i.e. TAN using the training dataset is achieved. The outcome of this base classifier is assumed as the input data (known as Meta data) for the second step. This meta-level training set is composed by using the base classifiers' predictions on the validation set as attribute values, and the true class as the target [18]. From these predictions, the meta-learner adapts the characteristics and performance of the base classifier and computes a meta-classifier which is a model of the original training data set. This meta-classifier in second step fetches the predictions from the base classifier for classifying an unlabeled instance, and then makes the final classification decision.

## 4.5. Testing Phase

The classification rules that are generated in Learning Phase are stored for the performance evaluation of hybrid intrusion detection framework. In this phase, the Testing Set generated in Data Splitting module is used as input to assess the performance. The outcomes of this module is further forwarded to next module i.e. Classifier Performance Evaluator module.

## 4.6. Classifier Performance Evaluator

The Classifier Performance Evaluator module calculates the various classification performance measures in order to judge the accuracy of the Hybrid IDS framework. These measures are as follows:

- True Positive Rate (TPR):

$$TPR = \frac{TP}{TP + FN}$$

- False Positive Rate (FPR):

$$FPR = \frac{FP}{TN + FP}$$

Where TP (True Positive), FN (False Negative), FP (False Positive) and TN (True Negative) can be defined as follows [17]:

- True Negative (TN): These are the negative tuples that were correctly labeled by the classifier.
- True Positive (TP): These refer the positive tuples that were correctly labeled by the classifier.
- False Positive (FP): These are the negative tuples that were incorrectly labeled as positive.
- False Negative (FN): These are the positive tuples that were mislabeled as negative.

These terms can also be understood by the confusion matrix shown in Table 1, where a confusion matrix is a tabular visualization of the performance of an algorithm. The column in the matrix represents the instances of a prediction class while the row represents the instances of an actual class.

Table 1: Confusion Matrix for TN, TP, FN and FP

| | **Correctly Classified** | **Incorrectly Classified** |
|---|---|---|
| **Valid Record** | True Negative (TN) | False Positive (FP) |
| **Attack Record** | True Positive (TP) | False Negative (FN) |

### 4.7. Visualization

The result generated in the Performance Evaluation phase can be visualized in the visualization module. These results can be in the form of text or graph etc.

## 5. Experimental Analysis

This section describes the experimental outcomes of the developed hybrid intrusion detection framework and its comparison with various other techniques present in the scenario. It has been noticed that the outcomes of the hybrid IDS framework excelled most of the algorithms in respect of performance (prominently accuracy).

Following Table 2 and 3 is the comparison of the two algorithms i.e. TAN and REP utilized in the hybrid IDS framework with respect to the frequently preferred bayes net based K2 algorithm.

Table 2: Performance Comparison of TAN and K2

| **Class** | **TAN** | | **K2** | |
|---|---|---|---|---|
| | ***TPR*** | ***FPR*** | ***TPR*** | ***FPR*** |
| DoS | 0.999 | 0.000 | 0.988 | 0.000 |
| Probe | 0.986 | 0.000 | 0.978 | 0.005 |
| R2L | 0.967 | 0.000 | 0.959 | 0.001 |
| U2R | 0.857 | 0.000 | 0.810 | 0.005 |
| Normal | 0.999 | 0.001 | 0.985 | 0.002 |
| | | | | |

Table 3: Performance Comparison of REP and K2

| **Class** | **REP** | | **K2** | |
|---|---|---|---|---|
| | ***TPR*** | ***FPR*** | ***TPR*** | ***FPR*** |
| DoS | 1.000 | 0.001 | 0.988 | 0.000 |
| Probe | 0.978 | 0.000 | 0.978 | 0.005 |
| R2L | 0.982 | 0.000 | 0.959 | 0.001 |
| U2R | 0.667 | 0.000 | 0.810 | 0.005 |
| Normal | 0.999 | 0.000 | 0.985 | 0.002 |

Next the Table 4 shows the comparison of the developed framework with the K2 algorithms proving its effectiveness with improved results in case of each type of attacks.

Table 4: Performance Comparison of Hybrid and K2

| **Class** | **Hybrid** | | **K2** | |
|---|---|---|---|---|
| | ***TPR*** | ***FPR*** | ***TPR*** | ***FPR*** |
| DoS | 1.000 | 0.001 | 0.988 | 0.000 |
| Probe | 0.987 | 0.000 | 0.978 | 0.005 |
| R2L | 0.971 | 0.000 | 0.959 | 0.001 |
| U2R | 0.833 | 0.000 | 0.810 | 0.005 |
| Normal | 0.999 | 0.000 | 0985 | 0.002 |

When the developed Hybrid framework is compared with the two eminent algorithms, the obtained outcomes shows the benefit of the framework in handling the shortcomings noticed in case of TAN and REP, hence making the system intelligent enough to detect the intrusions to a remarkable extent. This can be visualized through graphs (fig. 3 and fig. 4) showing the number correctly classified instances and number of incorrectly classified instances.

The following figures (fig. 5-7) show the class-wise Accuracy comparison among K2, TAN, REP and the developed Hybrid framework.

The Effectiveness of the developed hybrid framework can be understood by the following Figure (fig. 8-9). It can be noticed that ranging from K2 to Hybrid Model, there is a rapid increase in the true positive rate and decrease in false positive rate.

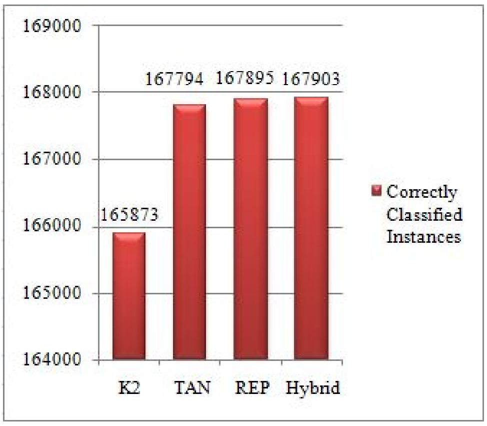


Fig. 2: Numbers of Correctly Classified Instances in K2, TAN, REP and Hybrid Models

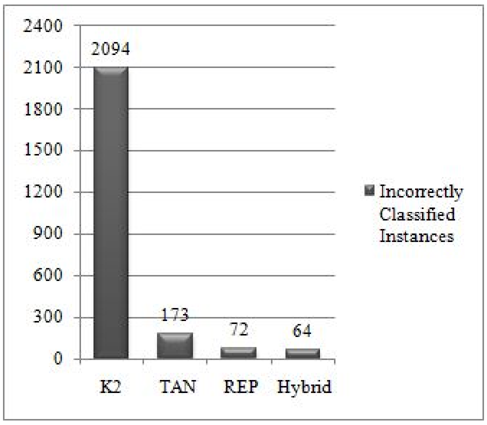


Fig. 3: Numbers of Incorrectly Classified Instances in K2, TAN, REP and Hybrid Models

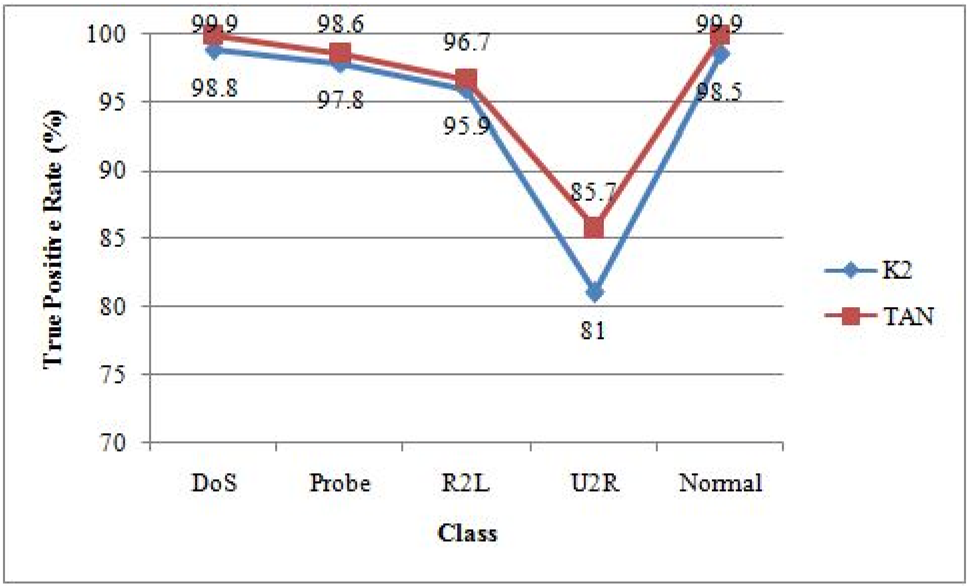


Fig. 4: Class-wise comparison of accuracy in K2 and TAN

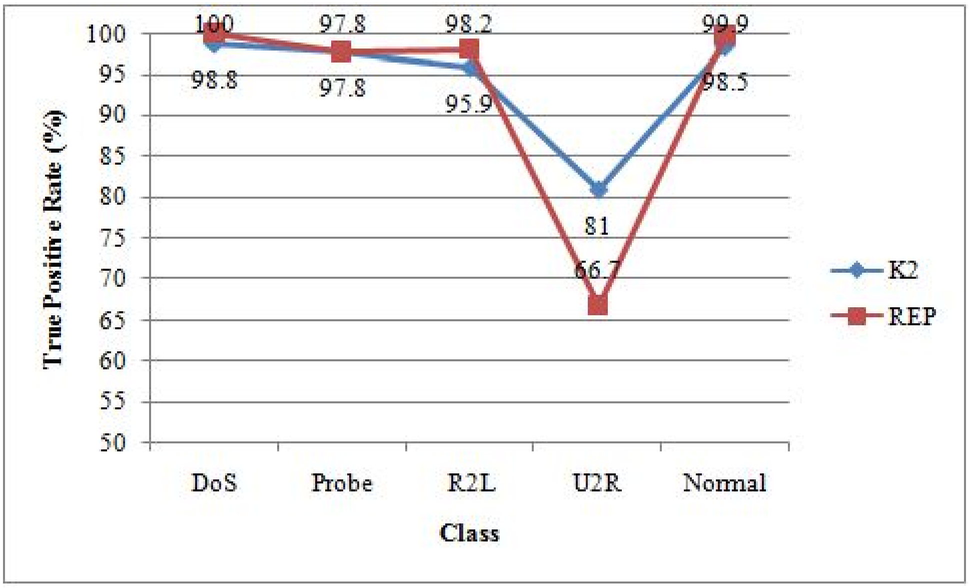


Fig. 5: Class-wise Comparison of accuracy in K2 and REP

DOI https://doi.org/2024-JICS-22595 | Generated on 2025-04-05 22:12:03

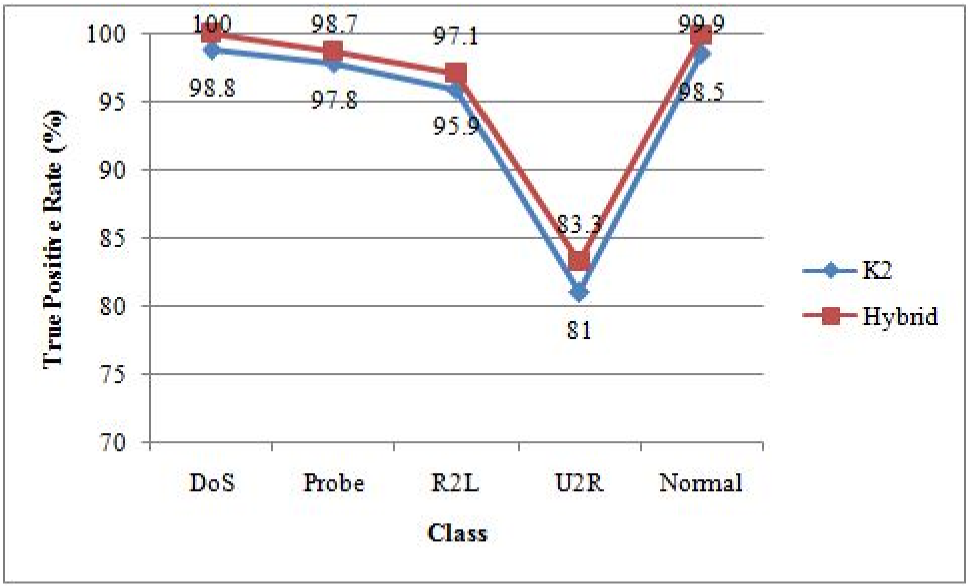


Fig. 6: Class-wise comparison of accuracy in K2 and Hybrid

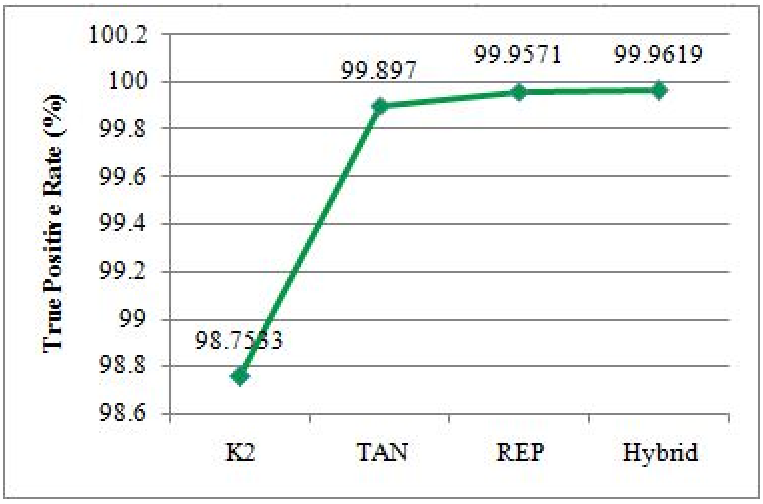


Fig. 7: Comparison of TPR among K2, TAN, REP and Hybrid

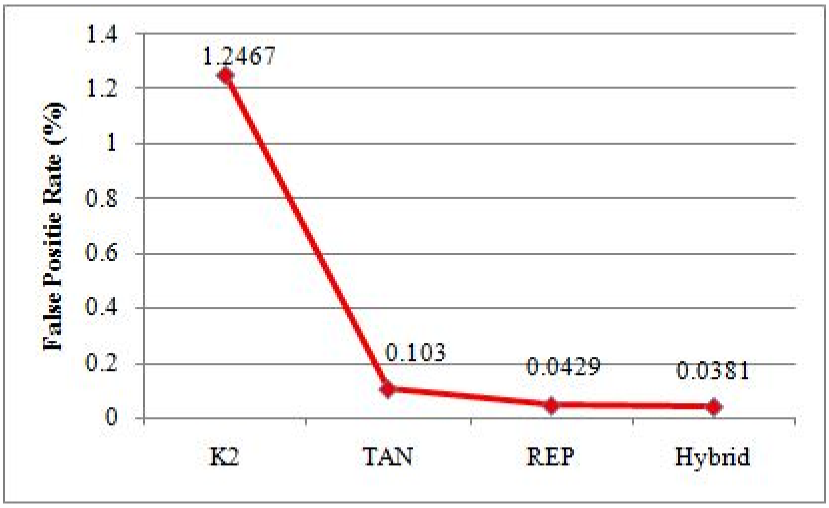


Fig. 8: Comparison of FPR among K2, TAN, REP and Hybrid

DOI https://doi.org/2024-JICS-22595 | Generated on 2025-04-05 22:12:03

When the developed framework is compared with the respective various available data mining techniques for intrusion detection, the resultant obtained shows the favorable opinion to opt as the hybrid technique. The lead may be understood from the following comparison graph:

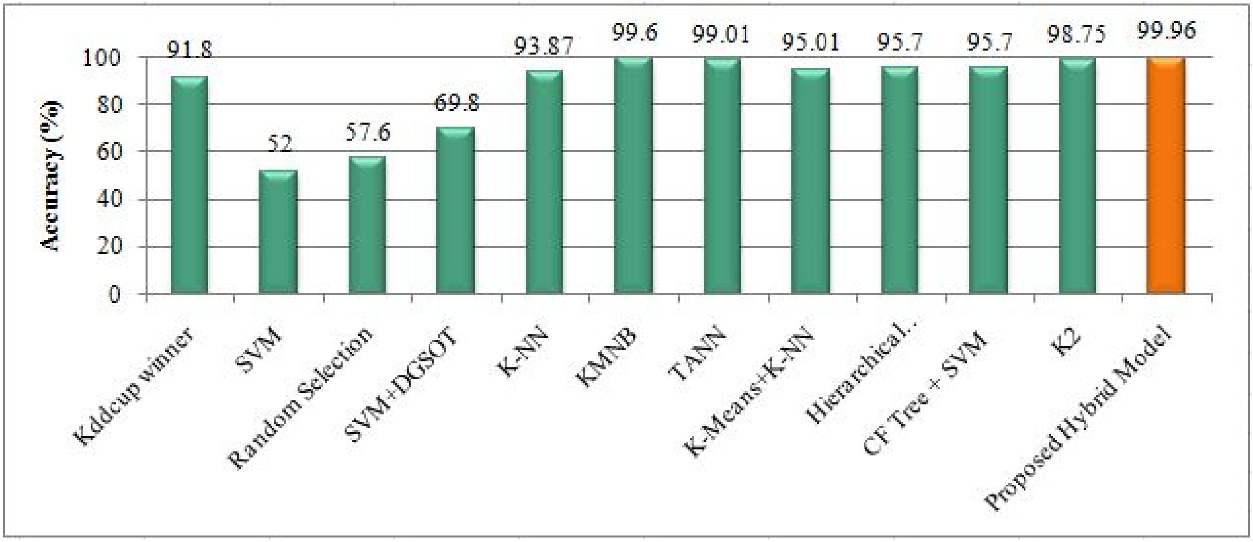


Fig. 9: Accuracy comparisons of various data mining-based IDS Models

## 6. Conclusion and Future Aspect

This paper proposes an envisioning framework for intrusion detection i.e. Hybrid Intrusion Detection System. The developed framework is an intelligent, adaptive and effective intrusion detection framework. The experimental analysis is performed on the developed IDS framework and is compared with other techniques present in the scenario. The resultants obtained convey that the developed hybrid framework is highly effective to overcome the deficiencies found in previous work. As the framework uses two data mining techniques (i.e. TAN and REP) to breed the classification rules, it can be effortlessly implemented in real time and is able to detect and adapt new types of intrusive activities. Also experimental assessment shows that the developed framework has reduced the false alarm rate and increased the accuracy up to noteworthy extend which is a major concern in case of intrusion detection mechanism. In addition to this, the framework is able to detect U2R and R2L attacks more efficiently than previous findings, boosting up the detection process. In future, some more work can be made in order to detect U2R and R2L attacks more accurately which may tend to further enhance the system efficiency.

## 7. References

[1] Wenke Lee and Salvatore J.Stolfo, *Data Mining Approaches for Intrusion Detection*, Proceedings of the 7th USENIX Security Symposium San Antonio, Texas, January 26-29, 1998.

[2] Sattarova Feruza Yusufovna, *Integrating Intrusion Detection System and Data Mining*, Proceedings of International Symposium on Ubiquitous Multimedia Computing, IEEE (2008), pp. 256-259.

[3] Zillur Rehman, S S Ahmedur Rehman, Lawangeen Khan, *Survey Reports on Four Selected Research Papers on Data Mining Based Intrusion Detection System*, University of West Indies at Mona, 2009.

[4] Parekh S.P., Madan B.S. And Tugnayat R.M., *Approach For Intrusion Detection System Using Data Mining*, Journal of Data Mining and Knowledge Discovery, ISSN: 2229–6662 & ISSN: 2229–6670, Volume 3, Issue 2 (2012), pp.-83-87.

[5] Srinivas Mukkamala, Andrew H. Sung, Ajith Abraham, *Intrusion detection using an ensemble of intelligent paradigms*, Elsevier, Journal of Network and Computer Applications 28 (2005) pp.-167–182.

[6] Sandhya Peddabachigari, Ajith Abraham,Crina Grosan, Johnson Thomas, *Modeling intrusion detection system using hybrid intelligent systems*, Elsevier, Journal of Network and Computer Applications 30 (2007), pp.114-132.

[7] Daejoon Joo, Taeho Hong, Ingoo Han, *The neural network models for IDS based on the asymmetric costs of false*

DOI https://doi.org/2024-JICS-22595 | Generated on 2025-04-05 22:12:03

*negative errors and false positive errors*, Elsevier, Expert Systems with Applications 25 (2003), pp.- 69–75.

[8] Farah Jemili, Dr. Montaceur Zaghdoud, Pr. Mohamed Ben Ahmed, *A Framework for an Adaptive Intrusion Detection System using Bayesian Network*, Intelligence and Security Informatics, IEEE (2007).

[9] Mrutyunjaya Panda and Manas Ranjan Patra, *Network Intrusion Detection Using Naïve Bayes*, IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.12, December (2007).

[10] Su-Yun Wu, Ester Yen, *Data mining-based intrusion detectors*, Elsevier, Expert Systems with Applications 36 (2009), pp. 5605-5612.

[11] Wei-Hao Lin and Alexander Hauptmann, *Meta-classification: Combining Multimodal Classifiers*, Springer, Mining Multimedia and Complex Data, LNAI 2797 (2003) pp. 217–231.

[12] Alexandra M. Carvalho, Arlindo L. Oliveira and Marie-France Sagot, *Efficient learning of Bayesian network classifiers: An extension to the TAN classifier*, Proceedings of Advances in Artificial Intelligence, Springer, Volume 4830, (2007), pp 16-25.

[13] Ajit Singh, *Tree-augmented naive bayes*, Homework 2 Problem 7 of Probabilistic Graphical Models, Fall 2006.

[14] Tapio Elomaa and Matti Kääriäinen, An analysis of Reduced Error Pruning, Journal of Atificial Intelligence Research 15 (2001), pp. 163-187

[15] Johannes Fürnkranz, *Pruning Algorithms for Rule Learning*, Machine Learning, 27 Kluwer Academic Publishers (1997), pp. 139-172.

[16] KddCup99 dataset, available at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, 1999.

[17] Jiawei Han & Micheline Kamber, *Data Mining: Concepts and Techniques*, Second Edition, Morgan Kaufmann Publishers, 2006.

[18] Andreas L., Prodromidis and Salvatore J. Stolfo, *A Comparative Evaluation of Meta-Learning Strategies over Large and Distributed Data Sets*, In workshop on Meta Learning, Sixteenth Intl. Conf. Machine Learning, 1999.