Diverse Analysis of Data Mining and Machine Learning Algorithms to Secure Computer Network

Neeraj Kumar1 · Upendra Kumar2

Accepted: 14 November 2021 / Published online: 2 December 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract
Network attacks are becoming more complex, making it more difficult to detect intrusions. Various research have been done over the years, employing different categorization techniques of Data Mining (DM) and Machine Learning (ML) inspired hybrid approaches to develop robust IDS. Almost all researchers suggested to improve accuracy in intrusion detection with low computational cost. Authors observed that dissimilar sets of features were picked for different classifiers to get the highest accuracy. This paper is dedicated to a review of relevant research, where an in-depth investigation was carried out with two emphasis points of IDS, which contain distinct pre-processing techniques in the form of feature selection and a diversity of classification algorithms. In addition, this paper presents a comparative algorithmic assessment of the DM and ML techniques applied to create an intelligent IDS. A novel feature selection method based on the CART algorithm has also introduced which provides optimal feature subset of the dataset so perfectly that it has made various existing DM and ML classifying algorithms more performant than earlier, this makes classifiers independent of feature selection. To validate the performance of proposed work, experiments have performed using the ‘Python’ programming language and ‘corrected’ & ‘10_percent’ of ‘Kddcup99’ datasets used as benchmark. As an outcome proposed work, we found that feature reduction and selecting a classifier had a significant influence on the rate of intrusion detection accuracy. The results of simulation and comparison analysis of proposed work with existing DM and ML based classification approaches show that the suggested work is more competent in true prediction and attaining maximum intrusion detection accuracy with minimal computing cost of prediction.

Keywords Intrusion detection · Dimensionality reduction · Clustering · Machine learning · Data mining

Neeraj Kumar
phdcs100009.16@bitmesra.ac.in

Upendra Kumar
upendrakr@bitmesra.ac.in

1 Computer Science and Engineering, Birla Institute of Technology, Mesra, India
2 Department of Computer Science and Engineering, Birla Institute of Technology, Mesra, India
1 Introduction

Because information technology is so prevalent in many parts of human life, safeguarding computers against attacks has become a crucial necessity. The widespread usage of computer technology has resulted in a plethora of vulnerabilities and threats such as viruses, Trojans, Denial of Service (DoS), ransomwares, and so on. Although there are numerous solutions available to protect against these harmful attacks, such as antivirus software, gateways, and firewalls. They are insufficient to deal with the wide range of infiltration attempts, especially when there are botnets with malevolent intent. As a result, the IT community urgently requires an intelligent and dependable tool or process to deal with these vulnerabilities and threats. In contrast to virus prevention, firewalls, and other user access approaches, intrusion detection systems (IDS) are intelligent technologies that identify both known and new threats. As a result, IDS, which utilises a range of categorization techniques, serves as the major source of inspiration for this in-depth analysis.

Most researchers want to make IDS more intelligent so that they can detect new forms of assaults to the maximum extent possible. These systems are broadly classified into three types: signature detection systems, anomaly detection systems, and hybrid detection systems. In signature-based attack detection, a corrupted pattern of network traffic or application data is identified in order to detect malicious intents, whereas in anomaly-based systems, deviations from good or normal behaviour are detected in order to detect intrusions, which typically rely on ML techniques. Hybrid methods combine the advantages of both Signature and Anomaly detection techniques. Although each technology has advantages and limitations, anomaly-based systems offer some significant advantages over other methods. Because anomaly-based systems identify deviations from typical behaviour, they may quickly detect abnormal occurrences as well as previously unknown assaults. There are several new classifying techniques of DM and ML tools available to detect the intrusion.

Another aspect of analysis is dimensionality reduction because most of the time all the dimensions of dataset is not participated or not necessary for the desired goal. It can be achieved in two ways, one of them is feature selection method in which we select the relevant feature and feed it to the system as per the requirement and discard the remaining features. It is used for linear dataset. Feature selection can be done in three steps: ‘Filter’ to find out the information gain, ‘Wrapping’ for accuracy and ‘Embedding’ to determine the errors, which in turn assists in removing or adding the necessary feature. Another method of dimensionality reduction is feature extraction, where feature is selected as per inter dependencies among dimensions; it is used for non-linear dataset. It can be done in many ways. First extracting the row by determining the missing values as it decreases the accuracy. Another way is to eliminate the feature that has low variance i.e. minimum differences between features and has least impact on accuracy. Extraction may be achieved by calculating the high correlation of dimensions. If two dimensions having high correlation, in that case one of the dimensions can be extracted, which will have no impact on accuracy. In the same context, feature extraction may be attained by Principal Component Analysis (PCA) using orthogonal projection, generation of dataset with new dimensions. Backward feature elimination & Forward feature selection works on network training with least error and Random Forest (RF) to prepare the dimension as per given target. As a result, dimensionality reduction is crucial in reducing spatial complexity.

The benefits of each of the aforementioned necessities have prompted us to undertake comprehensive research with this poll. In order for researchers to choose the optimum mix of classification and dimensionality reduction approaches. To obtain maximum attack
detection accuracy rate while maintaining extremely low time and space complexity, this article has discussed different classification approaches of DM and ML as well as dimensionality reduction, as well as the fundamentals connected with them are reviewed and assessed.

2 Related Work

2.1 Pre-Processing and Dimensionality Reduction

Pre-processing is the process of preparing a dataset for use in training different classifiers. This step includes several dataset treatments such as data cleansing, integrating, transformation, reduction, and discretization, among others. Data cleaning includes filling in missing values, finding and removing outliers, and settling disputes. Data integration is the process of combining several datasets. The primary goal of pre-processing is data reduction, and it is also accessible in lightweight form with the same importance. Dimensionality reduction is the technique of producing the same or better analytical findings by reducing the size of the dataset. The most challenging part of the dimensionality reduction is the Feature Selection or Feature Reduction. Selecting the most relevant features or reducing the dimensions improves the performance of algorithm causing the improvement in detection rate. Again these techniques can be classified as: Feature Selection (e.g. filter based, wrapper based etc.), Feature Projection (e.g. PCA, Linear Discriminant Analysis etc.) and Combinatorial Approaches, where researchers try to fuse the features from both former types. Different techniques used in pre-processing are discussed in this section.

2.1.1 Principal Component Analysis

Principle Component Analysis (PCA) is a statistical method that uses orthogonal transformations to turn a set of data into a linear combination of values from linearly non-overlapping variables known as principle components. It is one of the most widely accepted methods for decreasing the dimensionality of datasets.

Dataset—In general, a dataset is a multivariate collection of data that is used to train models/classifiers. The most popular 'NSL-KDD’ & 'Kddcup99' and datasets are associated with NIDS research. Previously, the 'Kddcup99' dataset was widely utilised in research, but subsequently, the 'NSL-KDD' dataset, which is a normalised version of the Kddcup99, became popular.

The authors used PCA to execute their analytical experiment on the kddcup99 dataset with all 41 dimensions for dimensionality reduction by viewing 10 main components with no relationships between them. They attained 99.7 percent accuracy, which was virtually identical to the original dataset due to noisy data and the use of the RF method as a classifier [1]. For feature selection, the researchers used PCA in conjunction with the Fisher Discriminant Ratio. For lowering the original features to an optimum set of reduced number of features and assessing classifier performance on those set of features, Self-Organizing Maps was used as the main classifier, achieving about 90% accuracy with 20 features out of 41 kddcup99 datasets [2].

The authors used the method of combining PCA with Information Gain to reduce dimensionality (IG). According to the experimental data, the ensemble classifiers may reach an accuracy of up to 98.24 percent [3]. One of the authors used the PCA in a novel
method, extracting hidden features to investigate the fitness of encoding categorical characteristics. It was the likelihood of an attack trained on the characteristic in the context of IDS. KNN classifiers were used to find hidden features in numeric form and with 40 different classes in the NSL-KDD dataset, obtaining 98.05 percent detection rate with a 35 percent false alarm rate [4].

PCA is clearly the most dependable approach for dimensionality reduction. It removes associated features, improves classifier performance, decreases overfitting, and improves visualisation. Before using PCA, several measures, such as data standardisation, should be taken. Otherwise, it will be unable to discover the optimal principal components, resulting in the original features becoming Principal Components and no longer being considered as original features.

### 2.1.2 Information and Rough Set Theory

For the aim of feature selection, information-theoretic methods have also been explored and evaluated. Various entropies, including as Conditional Entropy, Relative Entropy, Correlation, Variance, Gini-Index, and so on, as well as the concepts of Information Gain and Correlation Analysis, are used to choose the most relevant features from a multidimensional dataset.

A mutual information-based feature reduction approach was utilised in conjunction with some other pre-processing methods to enhance the efficacy of feature selection. These combinations were found to be better at controlling both linear and non-linear dataset properties. A greedy feature selection-based mutual information approach, as well as a combination of feature mutual information and feature-class mutual information, were used. It was also capable of creating 24 optimal subsets of attributes, which has been demonstrated to be beneficial in minimizing redundancy. It has been observed from the outcomes that the performance of the classifiers like Support Vector Machine (SVM), KNN, RF, DT was improved with best computational complexity [5, 6]. Rough set theory came into existence to provide mathematical solution for imperfect knowledge about data. It has been using in wide variety of fields. Its importance in ML and AI is irreplaceable. It has been used for feature selection from datasets with a large number of dimensions, either in the stock version or after some alteration.

The authors attempted quick feature selection using rough set theory; supervised feature selection in conjunction with enhanced harmony search was used to reduce the dataset’s dimensions. Following feature selection, the classifiers achieved an accuracy of 90%. The time required for categorization was also decreased [7]. For data pre-processing and dimensionality reduction, an IDS based on Rough Set Theory and SVM was explored. They decreased the number of variables from 41 to 29 using rough set theory. Finally, the dataset with fewer dimensions was given to the SVM classifier. The results indicated an improvement in false positive rate and accuracy of 92.44 percent [8].

Many studies have been conducted, however they are still missing in the creation of an adaptable model of IDS. This issue was also taken into account by the authors in their study, and they built an adaptive model of IDS capable of detecting new assaults as well. Prior to feature selection, the criterion of information gain ratio was used to produce an ideal attribute using fuzzy rough set theory. For clustering, a global optimum Gaussian mixture model was used. They proposed utilizing such a combination to destroy the fundamental structure of the nodes involved, resulting in a highly identifiable, stable, and
characteristic incursion pattern. The NSL-KDD dataset as a whole contains 125,937 records for training and 22,544 records for testing.

We learned about the idea of rough sets and their capacity to reduce the original dataset into the fewest number of sets with similar information as the original dataset via this research. This method is more successful at discovering hidden data when no further information about the data is available. Hence it is broadly adaptable.

### 2.1.3 Genetic Algorithms

A genetic algorithm (GA) is a heuristic approach in computer science and scientific research that is backed by a mechanism of expected choice from a larger class of evolutionary algorithms. They employ bio-inspired processes like as mutation, crossover, and selection to provide high-quality research solutions. These algorithms have lately acquired traction in the domains of AI and machine learning. The researchers used the same approaches for feature selection as well. It was used to choose the feature, and then classification was performed with the use of DT, with improvements identified in the DT’s classification characteristics for developing robust IDS [9]. For feature selection, an enhanced Non-Dominated Sorting GA-III was used. This method assisted in resolving the imbalance issue as well as reducing the superfluous features. This technique resulted in improved classification accuracy as well as reduced processing complexity [10].

Later, researchers presented the Hyper-Graph-Genetic-Algorithm in support of SVM for parameter and feature selection optimization. Classifiers, such as SVM, rely significantly on parameters to function well. The hyper-clique feature was utilised to do a quick search for the best solution and to avoid traps in the local minima. With the aid of GA, the value of each parameter and feature subset for each chromosome was retrieved by calculating their fitness and deciding whether or not it would be part of the training and testing dataset. This technique increased detection rate while decreasing false alarm rate and provided the necessary number of features [11].

The researcher also attempted to decrease misclassification by utilising the benefits of GA in the development of a new model IDS. A research based on the DT and GA was presented to generate the C4.5, Decision Tree (DT) and GA were used to solve the problems of minor disjunction in the C4.5 on the kddcup99 dataset [12].

### 2.1.4 Other Approaches

Aside from the approaches described above, other experiments were conducted in order to discover even better optimization techniques for feature selection. In order to further refine the feature selection technique, the Cuttlefish optimization algorithm was suggested, which resulted in an improvement in the performance of classifiers employed for detection. The Cuttlefish algorithm essentially transforms the pattern into distinct colours. It is based on the reflected light and visible mechanism for the matching pattern as a subset. DT used fitness to assess the survival of this subset’s characteristic. According to the experimental findings of the kddcup99 dataset, less than 20 features were recommended out of 41 features, and evidently, as the number of features dropped, the Detection Rate rose. The results also showed that classifiers with certain feature subsets improved detection accuracy while lowering false alarm rates [13].

Duplicate and irrelevant data characteristics have a negative impact on classification performance. A study focuses on a filter-based feature reduction approach, which is also
a tried and true method for selecting a subset of features from a larger dimensionality. A mutual information-based feature selection technique was used to increase the efficacy of feature selection. This technique was capable of dealing with both linear and non-linear data properties. This technique was also tested on three different datasets: kddcup99, NSL-KDD, and Kyoto 2006+, producing classifiers with greater accuracy and lower computing cost [14].

Many studies have been published that use Fuzzy, GA-Fuzzy, and Nero-Fuzzy as Soft Computing methods to deal with feature uncertainty. An IDS was created using a wrapper-based feature selection method and a Neuro-Tree model as the classification engine. It achieved a detection rate of 98.4 percent, which was higher than the methods from the DT classifiers with which it was tested. The limitation of this technique is that it is only effective for feature selection and not for classification [15]. The feature selection techniques were put to the test using an IDS that employed a genetic fuzzy rule mining methodology. This was a multi-objective optimization method. The proposed method might potentially be utilised as a genetic feature reduction wrapper to solve the optimal feature problem. The classifiers’ performance has been improved by utilising a minimal number of features. For the creation of an efficient and reliable classifier for IDS, the combined classification technique, ant colony algorithm, and SVM were employed. They were able to choose the 19 most important characteristics from a total of 41 features. In tenfold cross validation, they obtained an accuracy of 98.6249 percent [16]. Along with SVM and SA, DT and SA methods were given in the same region. This combination was used to identify the best-selected attributes. To create detection class identification rules, DT and SA were used. SA additionally altered the optimal settings for both the DT and the SVM. In comparison to the other approaches tested, the recommended strategy outperformed them all [17].

Another hybrid approach, the Feature Vitality Based Reduction Method, was proposed to pick the necessary features from the entire number of available features in the dataset. The process includes eliminating one input feature at a time from the dataset, as well as training and testing the classifier. The process was repeated until the classifier’s performance significantly improved. They used probability-based Naive Bayes as the classifier’s engine. The proposed feature selection approach beat the Correlation-based Gain Ratio and Information Gain techniques. In a hybrid IDS, K-means, KNN, and Naive Bayes were used. To find important features, the entropy-based feature selection technique was utilised. K-means was used for classification, followed by a hybrid mix of KNN and Naive Bayes classifiers. The major goal was to reduce the amount of false alerts [18, 19].

MI-BGSA is a hybrid of the Binary Gravitational Search Algorithm (BGSA) and the Mutual Information Algorithm (MI) (MI). For global search, the BSGA approach was used as a wrapper-based feature selection strategy. Finally, when compared to other common wrapper-based and filter-based feature selection approaches, MI was utilized to increase the BGSA and pick the most relevant features, yielding a better detection rate [15]. Both the standard wrapper-based and filter-based feature selection techniques are outperformed by the proposed strategy. They combined affinity spreading with feature clustering. The studies employed 14 UCI datasets to assess classification accuracy and processing time. Later, an unsupervised feature selection approach was created, which employs the concept of labelling features as irrelevant if they have minimal dependencies on the rest of the features. Experiments were run on a variety of datasets. The findings demonstrated that the proposed method may uncover irrelevant features without limiting the effectiveness of the clustering algorithm [20, 21].

Several combinational research recommendations were employed in feature optimization to get the highest detection rate while having the lowest false detection rate (FDR).
It was discovered that variables that proved beneficial in a tree-based method and other classification models, such as SVM and DT, are more trustworthy. As a result, dataset pre-processing and dimensionality reduction are critical components of the intrusion detection process. It not only improves the performance of classifier algorithms, but it also reduces computing costs by decreasing complexity. In subsequent section, study has been done on the other important aspect of intrusion detection i.e. true classification.

2.2 Data Mining Approaches

DM is a process, which has widely been used by many organizations to turn raw data into meaningful information. It is a computer technique for detecting patterns in large datasets that includes approaches from ML, artificial intelligence (AI), statistical systems. Data mining’s objective is to extract information and patterns from massive volumes of data, not to gather data itself. The honest task of DM is to extract previously undiscovered, intriguing patterns like as clusters of data, odd records i.e. anomaly detection and dependencies (association rule mining) from large data via automatic or semi-automatic analysis. The extracted data in the form of information is utilized for further predictive analysis, ML and so on. It exists as a pool of techniques like Classification, Clustering, Anomaly detection, Association rule learning and Regression. In collaboration with DM’s techniques, we can be able to reduce the dimensions of large datasets as well as attain the accuracy in classification for intrusion detection.

2.2.1 Classification Techniques

Classification is a technique where we classify data sample into given classes. The main objective of a classification issue is to identify the class under which a new data will fall. It works on the basis of supervised learning, which includes training as well as testing data samples for accuracy in predicting the correct class. It may also predict continuous data values using regression analysis. These techniques are used to forecast cluster association for data occurrence. It is a task of considering each attribute of a record and assigning that record to a particular class, which is also known as target. Several techniques have been used to find anomalies. When broadly classified they fall into three categories i.e. DM, ML and Hybrid techniques. DM techniques are used various clustering methods like Naïve Bayes, Fuzzy Logic and Support Vector Machines etc. ML includes Neural Network (NN) and its variants while Hybrid methods tends to take the advantages of both DM and ML techniques. Thus an effort has been made to categorize incoming connection as ‘normal’ or ‘attack’. Studies about some popular works accomplished with various classifier algorithms like DT, RF, Naïve Bayes, K-Nearest Neighbours etc. have been presented in the forthcoming sections.

**Decision Tree**—A tool for decision support as tree-like model used to create a rule-base, which uses further to take decisions and possible outcomes. It is most applicable for classification as well as regression. It makes decision based on previously seen data and classifies the new data to a particular class. Decision is taking on each node of the tree and final class of the instance is decided at the leaf node. Many DT algorithms are available like Classification And Regression Tree (CART), ID3 (Iterative Dichotomise), C4.5 or J48 etc. have been used to create multi-way tree, in deciding and choosing node sequence in greedy fashion to generate the rule.
Hybrid method was contributed to deal with both types of detection methodologies (misuse and discrepancy) with the help of a rule-based decision support system. Where misuse-types of attacks were taking care by J48 DT and through Self-Organizing Map anomaly-type of attack has handled. Overall results of the experiment were found astonishing. The overall detection rate was 99.90% and missed rate was only 0.1% [22]. A proposed Markov blanket model for the feature selection came into picture using Bayesian Network, CART and Ensemble methods for the classification task. They were able to choose 17 of the 41 features in the kddcup99 dataset for classification using the CART algorithm. They got the higher accuracy in CART in comparison to the Bayesian Network. Overall 93.64% accuracy was achieved with DT after feature reduction [23]. Because DT uses Divide and Conquer methods, the main issue is to select the optimal characteristic for each node during the splitting stage that provides the most information gain. GA was also created to tackle the problem of creating DT. During construction the features used were assigned ‘1’ and features which were not used were marked with ‘0’. The fitness of feature characteristics was calculated by applying the idea of genes with a specific threshold frequency selection and DT for the purpose of categorization in order to improve detection rates and decrease false alarm rates [24].

There was also a proposal to increase SVM performance by using DT for feature selection. The major suggestion was to use node information provided by the DT to improve the performance of the SVM. The aim was to simply provide node information as an extra feature to the SVM along with the original characteristics of the dataset for further results. In this work, the experimental results of DT and SVM were compared separately and a hybrid combination of DT and SVM applied as the base classifier for further processing. The results were found to be more fruitful in order to achieve the goals with enhanced accuracy in comparison to any other classifier algorithms and thus it was declared the winner of the kddcup99 contest. A feature selection method as well as intrusion detection was proposed using SVM, DT and simulated angling. Features were chosen through derived decision rules applied on training dataset. As a result of this approach misclassification rates were found very low [9, 19, 25]. Many combinatory approaches comprising of DT, SVM, Genetic and Neuro were proposed by the researchers to improve the detection accuracy and dimensional reduction. A Wrapper based algorithm for feature selection using Neuro-Tree had been proposed in order to achieve better accuracy. Although the suggested technique was evaluated against the family of other six DT (Decision Stump, C4.5, Naive Baye’s, RF, Random Tree and Representative Tree) classifiers, the new algorithm topped the accuracy table when examining different members of the DT algorithms. The proposed algorithm gained the detection rate of 98.38% and error of only 1.62%. IDS with integrated anomaly detection and misuse detection was proposed by an author, in which C4.5 was suggested to generate tree as DT to develop misuse kinds of intrusion detection model and one class SVM was used for anomaly detection. The experiment made use of the NSL KDD dataset. The improvement in detection performance and detection speed demonstrated clearly in the article [26, 27].

Yet there was an issue with accuracy of attack detection, especially for class-type attacks which have very few samples or we can say that entities are less in number in these datasets like Probe, U2R and R2L in case of kddcup99. The DT algorithm using binary split and ID3 algorithm with quad split was used to improve rate of detection for U2R, R2L and Probe type of attacks as a remedy of the above issue [28].

Recently it was considered to look at a privacy property while implementing IDS in a network using DT. To achieve the same, researchers suggested to incorporate a pruning model into DT. So that privacy related concern of IDS, particularly IP addresses should not
be ignored. Since IP address comes under the personal property, all sensitive information from IDS will be “pruned” and hidden. The methodology behind modified DT was that “If at the time of tree generation an IP address is chosen as the splitting attribute then it will examine and identify for similarity or dissimilarity” by comparing it with predetermined sensitive IP addresses. This paper focused on sorting sensitive IP addresses using the sorting method in combination with DT [29].

Most of the researchers aimed in reducing the features using DT, CART and C4.5 causing improvement in accuracy due to which DT is useful in real-time forecasting even for big data. It is notable from the results that the DT classifier outperformed after the feature selection. It is not dependent on any pre-processed data. The advantage is that the missing values within dataset had no effect in developing the tree but its prediction is unstable. Any minor change in the dataset may lead to inaccurate predictions. It is more expensive and prone to overfitting.

**Random Forest**—RF is the most popular and powerful supervised learning ML algorithm. This algorithm can be viewed as an extension of DT based on bagging algorithm. It produces the output by creating many sub data trees, after combining them, to reduce the problem of overfitting. Many researchers for the task of IDS have also used RF. RF can be castoff in par with algorithms like SVM. It also shows RF’s ability to signify the importance of diverse nature of features, by showing different detection rates at different number of selected features. A paper used NN, SVM and RF to predict the importance of feature using rank as the basis. RF discards that irrelevant feature which has low rank, makes RF different from others existing algorithms [30–32].

Rule based IDS was restricted to the detection of known attack but was not able to detect novel attack. As a solution of the same researchers suggested the use of RF along with K-means hybrid approach to build the IDS which was able to detect both the anomaly and misuse detection. RF was used to construct a model for misuse detection whereas K-means was used to handle novel anomaly intrusion. This showed the importance of each feature of the kddcup99 dataset for implementation of IDS. Their findings show that the suggested method has a high detection rate and can detect fresh incursions. According to the findings, an IDS framework based on RF and weighted K-means was able to achieve high detection rates for anomaly detection with a 12.6 FPR. In comparison to the misuse detection rates, which obtained lower detection rates (92.73 percent) but high false positive rates (0.54%) [33]. To deal with the concern of ‘Class Imbalance’, scholars introduced Map Reduce technique for imbalanced data using RF. Although this article was not relevant for the case of IDS but it sheds some light on the real potential and the capabilities of the RF. Therefore this article can be considered as an inspiration for accomplishing IDS tasks using RFs. [34]. Along with these articles many researchers have also carried out IDS either using RF alone or making it hybrid with other algorithms. Many researchers have used RFs so that they can compare the results of their own proposed algorithm with that obtained using RF.

It perceives that RF is capable of performing classification tasks for big data but it performs poorly in regression, handling missing values and maintaining accuracy. Apart from this, it was found capable to handle large dataset with higher dimensionality. It is also capable of predicting high accuracy even when a forest has a number of DTs. It can’t predict beyond the range of the data in case of regression in which the problem of overfitting occurs due to noisy data. This algorithm takes much more training time in comparison to DT.

**Naive Bayes**—Naive Bayes another model of conditional probability classifier. In this model the assumption of independent forecast is almost correct. This algorithm has been used as a classifier in DM. Naive Bayes has been studied extensively since 1950’s. This
classifier has also been frequently utilized in the detection of intrusions. Some studies employed it alone, while others used it as part of a hybrid model with another classifier.

The Pseudo-Bayes estimator approach was employed to increase the capabilities of the anomaly detection system to detect new attack while decreasing the number of false alarms as possible. DT was compared with Naive Bayes using the same kddcup99 dataset. Although the quality of classification was almost the same in both classifications, it was found that Naive Bayes was about 8 times faster than DT [35]. The experiment was carried out using the kddcup99 dataset using a hybrid approach to IDS utilizing K-means Clustering and Naive Bayes. Initially, threats and regular occurrences were divided into various groups using K-means. Later, Naive Bayes was utilized to categorise the attacks even further. This method resulted in a lower false alarm rate and greater accuracy than relying just on K-means or Naive Bayes. Vitality Based Reduction method choose the most relevant features out of the dataset and put the reduced dataset to Naive Bayes for the classification [20]. In addition, for feature selection, a mixture of Proportional K-Interval Discretization (PKID) and Entropy Minimization Discretization (EMD) was employed, while Hidden Naive Bayes was used for classification. It was found that Hidden Naive Bayes performed very well having accuracy of 99.96% as compared to other six classifiers used in the experiment. Moreover like other described classifiers this classifier also has been used extensively, either for classification using itself or for the purpose of comparison with other classifiers [36]. An advanced Naive Bayesian classifier as Relief algorithm was introduced, in which, assign weights to each attribute of the dataset to maintain a relationship between attributes intended for better classification results. The proposed classifier showed better true positive rates and lower false alarm rates during detection [37].

From above Naive Bayes algorithm, it has been learnt that it improves the accuracy in less amount of training time by introducing only sample set of predictors. On the other hand, in this algorithm it is not so easy to obtain a completely independent predictor set that is not mutually independent between attributes.

### 2.2.2 Clustering Techniques

Clustering in DM is the task of grouping related things into clusters so the data point within a cluster is often more related to those in another. A variety of clustering algorithms are included in DM. K-means clustering is among the most often used clustering algorithms in IDS. To model an IDS, first normalize the dataset and then the single-linkage clustering approach was employed for clustering. More labelling of clusters had been planned. This system detected a large number of attacks while having a low false positive rate [38]. It was invented the Y-means clustering heuristic for intrusion detection. It used ‘K-means’ to address the problem of cluster dependence and degeneracy. The work was evaluated using the kddcup99 dataset. Finally, a detection rate of 82.32% with 2.5% of false alarm achieved [24].

A genetic clustering technique for intrusion detection was suggested, which was capable of automatically creating clusters that classified intruders as ‘normal’ or ‘abnormal.’ They performed clustering in the first step, then genetic optimization in the second stage to get the closest ideal detection result. The overall detection rate including all the classes of attacks was about 61%. However, the experiment was feasible and effective for the intrusion detection [39]. Several contributions having combined approach using K-means and NN were proposed for intrusion detection. In this case, K-means has been used to automatically choose an ideal set of samples and the result is then passed to the NN. The experiment was
shown in a paper, they found improvements in terms of time complexity and performance of NNs, as the addition of clustering before feeding the dataset to NNs. The experiment was carried out on the kddcup99 dataset and hierarchical clustering was employed to speed up the SVM training process [40]. The experiment also revealed improvements in training time as well as accuracy of classification and detection rate.

By combining K-means with KNN and Naive Bayes, a hybrid strategy was developed to minimize rate of false alarm of the proposed discrepancy in IDS, in which feature selection was done using an entropy-based feature selection approach to pick relevant characteristics and eliminate irrelevant ones. First, the feature was chosen using the previously predicted feature selection technique. The clustering was done using K-means and the classification was done using a hybrid technique of k-neighbour and Naive Bayes. The results showed an increase in the detection rate along with decrease in the false positive. Using this approach, the detection rate raised to 99.35 percent [41].

Later, a hybrid technique based combined approach K-means with C4.5 algorithm was suggested for detecting network anomalies. K-means was applied in this hybrid technique to split the dataset into clusters for training and testing using Euclidian distance and then rules were created using C4.5 DT for classification. The experiment’s performance measure chart showed that it intelligently outperformed the majority of classifiers, with precision of 95.6 percent and accuracy of 95.8 percent. Another suggestion included a classifier based on Cluster Centre and Nearest Neighbour (CANN). Here they suggested two distances were measured and summed for each data sample from its cluster centre and neighbouring data sample. The researcher used CANN in the KNN classifier. The findings produced using the kddcup99 dataset show a significant improvement in training and testing time as well as accuracy when compared to KNN and SVM. Another researcher presented a novel ensemble building approach based on PSO produced weights. As a meta-optimizer, local unimodal sampling (LUS) was utilized. Experiments on the kddcup99 dataset revealed that it could build superior ensembles that outperformed traditional ensemble techniques [42, 43].

K-means simplifies clusters of various sizes and forms and it is simple to implement for clustering. The K-means method may be performed multiple times to find the initial value ‘k,’ however the difficulty occurs for outlier instances. It does not perform well with large dimension, so it would be preferable to have dimensionality reduction using PCA or any other modified clustering algorithms.

2.3 Machine Learning Approaches

ML is a subfield of AI that use statistical techniques to allow computers to learn from data rather than being model for predicting. ML was extensively used for IDS tasks. Generally, the classification task in IDS is accomplish using the ML techniques. Sometimes, they used ML along with other DM classifiers with an aim to enhance the performance of the existing classifiers [44]. ANN (Artificial Neural Network) also most frequently used algorithms for classification and has been extensively utilized in IDSs. Tons of experiments were conducted utilizing ML approaches, since ML is also one of the trendiest and most developing areas of the previous decade, with numerous advancements.

Support Vector Machine (SVM)—SVM splits the dataset into different classes by determining the centre point decision boundary, which is closer to the opponent class also known as hyperplane. For linear separation via SVM, the hyperplane that has the maximum marginal width of the decision boundary is chosen to avoid misclassification. Therefore,
MMH (Maximal Margin Hyperplane) is the key factor for classification with maximum accuracy. For Non-Linear data classification, Kernel is the key factor which accepts the LD (Low Dimensional) feature space and gives an output with HD (High Dimensional) feature space. It is supervised learning model, so labelled dataset is used for training and has been used heavily in last decade for the misuse type of intrusion detection. SVM is used for classification as well as in regression too.

As far as SVM classification is concerned, researchers have been contributed plenty of articles. Initially contributions were just to get acquainted with the principle of the SVM as it is capable to classify the data-sample into two or multi-classes. Later, SVM was utilized to develop IDS using DARPA 1998 dataset with NN. It was found that the training time for SVM was less than NN. It was also found that SVM has a slightly higher rate of correct detection as compared to NN when used for the same task. However, only the binary classification was possible using SVM, which was a disadvantage for that time. A classification task was performed using SVM and detection rate was used as evaluation criteria. Overall result was better than the winner of the contest of kddcup99. Detection accuracy was found in case of ‘Normal’ 99.3%, ‘DoS’ 91.6%, but lacking in ‘Probe’ 36.65, ‘U2R’ 12%, ‘R2L’ 22%. Poor performance in ‘Probe’, ‘U2R’ and ‘R2L’ was due to less no. of training sample [45–47].

The suggested Rough Set and SVM coupled methods were not only limited to feature selection but were also acceptable for classification. In terms of training and testing time, it was discovered that following feature selection, more accurate results were achieved. Only 29 of the 41 kddcup99 features were chosen, trained and categorised using SVM utilizing the Rough Set. As a result, they achieved accuracy ranging from 86.79 to 89.13% with 29 features compared to 41 features [11]. Some other combinatorial approaches for feature removal method called ‘gradually feature removal method was suggested. In this proposal they used Ant Colony algorithm and SVM. Evaluation was done onto tenfold cross validation to train the network. After reducing feature into each round, dimensions reduced from 41 to 4 and accuracy achieved was 98.6249% [18].

A Multi-Levelled-Hybrid intrusion detection model was suggested and tested on 10% kdd dataset. For pre-processing, initially features ‘protocol_type’, ‘service’ and ‘flag’ were transformed into numeric form. A modified version of K-means algorithm was also presented so that a high-quality training dataset can be used. Subsequently they applied improved K-means on every sub-category (Labelled attack: Normal, DoS, Probe, R2L, U2R) and generated new different number of attributes, which were chosen for different targets of every category. Number of samples of before and after utilizing improved K-means with 10%_Kddcup dataset in this manner was Normal (97,278 to 639), DoS (391,458 to 140), Probe (4107 to 134), R2L (1126 to 51), U2R (52 to 25). Further classification had been performed with this newly generated dataset using SVM and ELM (Extreme Learning Machine). Thereafter testing was done with multi-level model on corrected dataset of kddcup99. The comparative result was found using basic K-means with Modified K-means, with these metric of evaluation: Accuracy (91.88% to 95.75%), DR (92.13to 95.17) and FAR (9.16 to 1.87) [48].

A proposal came into existence in the direction of impulsive protection in the wireless sensor networks to provide secure communication between two sensor nodes. In which clustering had been performed by sampling on the node’s weight. The adaptive chicken swarm optimization method is then executed by the cluster head. Its adaptive nature shortens the time required to pick the optimum cluster head. Because of the increased degree of representation in compared to other sampling approaches, it outperforms the competition. Another approach, Rotated Random Forest (RRF), was used to minimize the features in the
dataset. For classification, SVM & ML two-phase classification approach was applied to reduce the dataset. In the first phase, sensor node predicts about the intrusion in the binary classification and in the second phase malicious sensor nodes were found capable in predicting about their types. In this work, researchers concluded that the RRF performed feature selection with higher detection accuracy and comparatively less time in comparison to normal RF. After that, by taking the advantage of this reduced feature the SVM was used as classifier resulting in improved accuracy of above 90% for DoS, Normal and Probe types of attack for detection of kddcup99 dataset with minimal FDR. At the same time, computational cost using these approaches were observed to be higher and it also performed poor in detecting classes like R2L and U2R of kdcup99 dataset as lesser number of samples was there to train. An author presented a combinatory approach of SVM-GA-ANN to deal with this problem [49].

To solve the issue of inadequate sample data for each feature to train any system, the ML-based approach was presented. This proposal describes a unique hybrid method to feature selection and intrusion detection. For feature selection, a wrapper technique GA with multi-parent crossover and multi-parent mutation (MGA) with SVM, dubbed MGA-SVM, was employed. To train the classifier, a hybrid gravity search (HGS), a particle swarm optimization (PSO) and ANN methods were employed in combination with MGA-SVM-HGS-PSO-ANN. Using only four of the 43 features in the NSL-KDD dataset, performance was compared to other common methods such as Chi-SVM, gradient descent (GD-ANN) and DT, GA-ANN and PSO (GSPSO-ANN) and a maximum detection accuracy of 99.3% was achieved [50].

From this analysis clears that most of the researchers adopted SVM to improve detection efficiency. At the same time disadvantage associated with SVM was that training & testing time and detection accuracy performance decreases for big-data. Hence it is recommendable for high dimensional space but would not be suggested for big-data without dimensionality reduction.

**Neural Network**—Neural Network (NN) is parallel computing device which works as human brain and can take decision with fast computation. In order to better identify intrusions, it can be effectively applied to deal with the upstairs issues. NN has just been utilized to take care of numerous issues related to pattern recognition, DM and complete AI. The challenge is so far an identification of new types of intrusion for which there is no prior knowledge. An NN modelled IDS was introduced to overcome this challenge along with very less FDR. Through the experiment, they obtained 96% DR with 7% FDR. This article, on the other hand, proved to be a source of inspiration for many future works in the field of IDS, identifying the problem of high false alarm rates and low detection rates of new threats employing NN [51]. To deal with the new attack, they had suggested a combination of discriminatory training and general keywords approach to minimize such problems. New keywords were added, which detect actions that were common to many attacks and used simple NN discriminant training to produce output. The improved system received a detection rate of approximately 80% per day at a low false alarm rate of approximately one false alarm [52].

It was noticed that the sample, which was small in number, had a lower intrusion detection rate over any supervised learning classifier. They had less participation in training so clearly perform poorer during testing. Many researchers proposed their issues when a small number of input feature were served for training. A contribution to deal with this issue, they had proposed a model of SVM-NN approach on kdcup99 dataset. They were able to identify 23 different class features using NN and SVM for a low-cost, real-time IDS. They used strategies such as removing one characteristic at a time, doing experiments and
ranking the relevance of each input feature. This process was repeated for all features individually and a set of features was obtained based on their rank. As an effect of the above dimensional reduction procedure, the results were mostly notable in terms of training time [53, 54]. A study on intrusion detection adopting hierarchical NN-based IDs was suggested to properly and adaptively identify abuse and anomalous attacks. In order to enhance the effectiveness of serial hierarchical IDS, parallel hierarchical IDS was introduced. They acquired an 89 percent detection rate and a 1.6% false positive rate during experiments. The paper also concludes how parallel hierarchical IDS was superior to a serial hierarchical NN that uses the hierarchical identity model PCA-NN. They attained PCA-NN based IDS suitable for adaptive online computing for misuse detection and anomaly detection. The paper shows ameliorated results over existing similar works of that time using its proposed method. However, in this paper they did not consider a class whose samples were less in numbers and dealt with anomaly type of attacks only [55]. In sight of above concerns, a researcher proposed a hybrid system called Artificial Immune System. In that paper Kohen Self Organizing Map (SOM) method adopted for the network intrusion detection. This approach found capable of handling both anomaly as well as misuse detection attack. An artificial immune system was used to detect anomalous network connections at first. Initially, it was proposed to use SOMs to identify abnormal connections for categorization. Later, those characteristics with higher-level information in the form of clusters were deleted. These experiments were carried out on the well-known kddcup99 dataset. Hence, it is observed that their experimental results had improved in comparison to other results obtained on similar tasks at the time [56].

Later on, hybrid soft computing was developed. The Fuzzy-ANN method was proposed to address primarily two issues: poorer detection precision for low frequency attacks and a lack of detection accuracy. This hybrid model employs a three-stage categorization system. They began by creating several training subsets using the fuzzy classification approach. Then, in the second stage, several ANNs were trained and finally in the final step a metalearn and fuzzy aggregation module were added to learn again and aggregate the outputs of the many ANNs. The concept behind this divide and conquer approach was adopted. The results were in fact productive and appeared in the context of the proposal. The system also improved the detection rate of less frequent attacks such as R2L and U2R of the kddcup99 dataset [57]. In this context another proposal was also made to simplify the same target using mutual information feature selection method, coupled with multilayer NN. They compared it to MLP and Radial Basis Function networks in their experimental evaluation. When compared to previous ideas, the suggested model was found to be more accurate [58, 59].

Through the ML based examines, we perceived that Anomaly-based search is an identification technique whereby IDS searches for issues using user-defined criteria rather than signatures currently stored in IDS. This form of identification generally adopts AI to differentiate between regular and abnormal traffic.

### 2.4 Hybrid Approaches

These approaches comprise of combined algorithms of DM or ML techniques to create or to implement a new algorithm from an existing algorithm. The goal is to improve either the performance or the efficiency of existing algorithms or to create a new algorithm to complete a task. In their quest to create the most efficient ID, researchers performed many tasks by combining different algorithms of DM or ML. Some of the works were described
in the above sections of this paper. During the period 2000 to 2007 the study focused on single and hybrid classifiers. After analyzing many articles, it was suggested to adopt ML to develop robust and computationally profitable IDS [60].

A research team proposed a novel intelligent hybrid approach to classify useful and useless features meeting the feature reduction through the first ranking method. This was achieved by combining both ranks which were derived from information gain and correlation. These reduced features were then input into forward NNs, which were trained and tested on the kddcup99 dataset. They then evaluated the proposed technique on five distinct test datasets and compared it to the absence of with and without feature selection using various assessment criteria [61]. Furthermore, this review approach was extended using C4.5, Naive Bayes and RF classification algorithms. The review researcher focused on cases of oversampling and undersampling features like U2R, R2L of the kddcup99 dataset. Because one class on oversampling can weaken the performance of another class. They came to the conclusion that sampling the class was more effective than monitoring the relevant occurrences for the U2R and R2L ranges. For solution U2R and probe orbit attacks, they found the Naive Bayes classification to be comparatively appropriate [62].

Hybrid approaches were found to be more suitable for both in detecting misuse and mismatch attacks. Misuse Intrusion Detection (MID) is a static type of approach that typically deals with well-known attacks using a set of rules. MID discovers a known attack with false positives. MID related issues are not useful in detecting new attacks. Whereas Anomaly Intrusion Detection (AID) is a dynamic approach which can detect unusual activity on the network. For such attacks first we need to know about the normal traffic to detect any discrepancy. It is capable of responding the new attacks due to dynamical change in network traffic other than normal traffic but at the same time caution should be taken as all abnormal traffic is not malicious. Therefore in case of AID we have to handle more alarms that are positive. While blocking attempts that matches the rule we can be more specific with MID and it is better to use alerts in AID. Therefore researchers can decode the right approach that should be taken as per their requirements. Most of the researchers preferred hybrid approach to develop an IDS model because the amalgamation of these techniques can do a superior job by countering their own flaws.

3 Comparative Discussion and Experimental Analysis

The aforesaid survey draws a lot of conclusions. Figure 1 shows the year-wise distribution of review articles since 1992. It also reflects the trends in approaches used by researchers for single or hybrid solutions to implement robust IDS for intrusion detection. From Fig. 1, we have seen that hybrid approaches were becoming popular with researchers proposing work for the past decade. This journey of algorithmic analysis yielded better results for handling both types of attack.

3.1 Research Gap as Motivation

The above survey on dimensional reduction for designing efficient IDS using various techniques for classification concludes that dimensional reduction plays a significant role in designing any IDS. It was found that reducing the dimensions, especially in case of huge dimensions, the dataset is complex and time consuming. Following are the major observations that were found as a research gap:
(i) For different classifier, dissimilar set of features was chosen for the same dataset, which infer that dimension reduction is classifier dependent.

(ii) Different features were used for the detection of different attacks within a classifier.

(iii) Suffering with False Alarm rate.

(iv) Average detection rate.

(v) Involve high computation cost.

(vi) Less performant in case of Big-data.

Various approaches are prevailing for the dimensionality reduction as stated above. Some problems noticed when reducing the dimensions using most of the existing techniques are:

- Same dataset has to be manipulated differently for different classifiers.
- Different number of dimensions (within the same dataset) are used for the detection of different entities. It makes dataset difficult to be fed into various advanced Machine Learning algorithms without making them fit for the algorithm first. So, most of the techniques becomes classifier dependent. These problems are one of the biggest limitations of the existing techniques.

### 3.2 Proposed Work

To overcome the above drawbacks, a novel method for the Feature Selection using Classification And Regression Tress (CART) algorithm is offered in this paper. This method of feature selection reduces dimensions of the dataset to make it fit for most of the classifier algorithms present in DM, ML as well as at the intersection of both.

This dimension reduction technique is classifier independent. Any dataset once fed is returned with the reduced dimensions which can be fed to most of the classifiers. The main advantage of this classifier is the ease with which it can be used and the consistency that it maintains. Thus, this hassle-free method can be executed on any dataset and the reduced dataset can be returned before feeding the dataset to the classifier.
3.3 Performance of Comparative Experiments

To explain the information collected from the survey, we carried out an experiment using assessment metrics on different DM and ML based popular classifiers such as DT, MLP, RF, K-means, KNN, and SVM. For the experiment, the ‘Python’ programming language is employed. The benchmark dataset kddcup99 was chosen, which has 41 features, but only five of them (duration, protocol type, flag, diff_srv_rate, and dst_host_rate) were picked after using the suggested feature reduction approach. Tables 1 and 2 demonstrate the ‘detection accuracy’ achieved by various classifiers before and after feature reduction.

Table 1 Comparison of Accuracy Detection (%) before and after Feature Selection using Kddcup99 corrected_dataset for training and full_dataset for testing

| Accuracy on difference classifiers | Decision Tree | MLP | Random Forest | K-Means | KNN | SVM |
|-----------------------------------|--------------|-----|---------------|---------|-----|-----|
| Before feature selection          | 85.009       | 94.93 | 96.805         | 1.472   | 91.425 | 0   |
| After feature selection           | 95.53        | 93.277 | 92.868         | 96.7    | 96.9 | 92.62 |

Table 2 Comparison of Accuracy Detection (%) before and after Feature Selection using Kddcup99 10_percent_dataset for training and full_dataset for testing

| Accuracy on difference classifiers | Decision Tree | MLP | Random Forest | K-Means | KNN | SVM |
|-----------------------------------|--------------|-----|---------------|---------|-----|-----|
| Before feature selection          | 92.089       | 94.85 | 97.305         | 87.053  | 87.053 | 0   |
| After feature selection           | 96.682       | 97.014 | 95.707         | 97.494  | 97.494 | 94.836 |
Comparison of Accuracy Detection Ratio (%) before and after Feature Selection on corrected dataset of Kddcup99

(a) Comparison of Accuracy Detection Ratio (%) before and after Feature Selection on corrected dataset of Kddcup99

(b) Comparison of Accuracy Detection Ratio (%) before and after Feature Selection on 10Percent dataset of Kddcup99

Fig. 2 a Comparison of Accuracy Detection Ratio (%) before and after Feature Selection on corrected dataset of Kddcup99. b Comparison of Accuracy Detection Ratio (%) before and after Feature Selection on 10Percent dataset of Kddcup99
Graphical representation of the above result has been exhibited in Fig. 2 a and b which shows the comparison of Accuracy Detection Ratio (%) before and after feature selection among the classifiers.

Table 5 shows the effect of feature selection and classifier selection on prediction accuracy. Where we acquired the results or influence on accuracy by comparing different techniques for feature selection given or utilised by researchers before. We also evaluated the performance of classifiers using our suggested feature selection approach on other ML-based classifiers. Experiment findings show that almost all classifiers used the same set of features and achieved increased or comparable intrusion detection accuracy.

3.4 Outcomes

The following experimental results are represented in tables and graphs above by examining two sets of Kddcup99 dataset, one is ‘corrected’ and the other one is ‘10_percent’. In the first case, training was done on a ‘corrected_dataset,’ while in the second case, a ‘10_percent’ dataset was used for training, but for testing we used the ‘full dataset’ of Kddcup99 in both cases for true classification in our proposed work. Table 1 and Fig. 2 a demonstrate the intrusion detection accuracy for the first case whereas Table 2 and Fig. 2 b for the second.

3.5 Various Conclusions Based on Experimental Results

1. Performance Accuracy of several classifiers obtained in First Case:
   Through Table 1 and Fig. 2a the effect of dimensionality reduction on DT & KNN and for true classification enhanced MLP results were obtained. At the same time in case of RF and MLP the result was at par.

2. Performance Accuracy of several classifiers Second Case:
   Through Table 2 and Fig. 2b in this case of classifiers like K-Means and SVM, rate of intrusion detection was found to be significantly improved. Furthermore, we observed that SVM do not performed well due to the large dataset prior to feature selection.
3. **Precision, Recall and F-score Graphs:**

Figures 3, 4 and 5 show that almost all of the classifier algorithms share the same personality, as demonstrated by the shape of the curves. The fact that the dataset has truly acquired for the majority of the algorithms while reducing such a large number of attributes is an excellent sign. Furthermore, the similarity of the classifiers demonstrates that none of them are under- or over-fitted.

4. **Fall of bar into the Figs. 3, 4 and 5 in both of the cases:**

The decline of the curve towards zero is visible in the case of R2L and U2R due to their low frequency (R2L is only 3.011% of the whole training set, while U2R is just 0.050% of the total training set) within the training and testing datasets.

5. **Feature Subset and Computational Cost**

As per Table 5, this specific suggested method FS-IDS model gives the least and most optimum subset with number of features '5' among different present approaches for Feature Selection. In intrusion detection systems, feature selection is essential for decreasing processing costs, expanding storage space, and enhancing data knowledge.

---

**Fig. 4** Recall (%) of Different Methods

**Fig. 5** F-Score (%) of Different Methods
6. **Overall Accuracy**

In terms of overall accuracy, almost all classifiers performed well on the datasets with reduced features. In the first scenario KNN achieved the greatest detection accuracy of 96.9% in this example. As indicated in Table 1, K-Means, RF and MLP have detection accuracy of 96.70%, 92.87%, and 93.28%, respectively. Table 2, on the other hand, shows the detection accuracy of several classifiers in the second example, where K-Means and KNN achieved the highest detection accuracy of 97.494%. While DT and MLP also fared well, scoring 96.682% and 97.014%, respectively. We are sacrificing RF accuracy in this case, but we are also saving training and testing time, which benefits us in terms of computing cost. When we talk about performance of SVM, we could able to predict after feature selection even for huge dataset.

7. **Impact of feature selection and choosing a classifier on detection accuracy:**

The impact of feature selection on detection accuracy has been empirically examined and the performance accuracy of our suggested approach has been compared to other researchers’ suggestions, as shown in Table 5. Table 5 shows the percentage of accuracy attained after feature selection for several classifiers. Table 5 shows that after removing a large number of features using our suggested feature selection approach, the accuracy of most classifiers has either enhanced or remained almost constant with both datasets when compared to the prior suggestion of numerous researchers.

8. **Training Time**

Table 3 compares the training times of several classifiers. The table clearly shows that our suggested feature selection approach reduces training time for all classifiers by more than half. SVM could not be trained before feature selection, but it could be trained after. Furthermore, SVM took the longest to train of all the classifiers in both situations. Only the K-Means classifier took a little longer, but it performed well in terms of intrusion detection accuracy after feature selection.

9. **Testing Time**

The testing periods of several classifiers are compared in Table 4. It is clear that after feature selection, almost all classifiers’ testing duration is reduced. K-Means was slower

| Table 3 | Time Taken in Training (in Seconds) on Different Classifiers on both Kdd data types |
|---------|---------------------------------|--------------------------------|-----------------|---------------|-------------|-------------|
| Outcomes on classifiers | Dataset | Decision Tree | MLP | Random Forest | K-Means | KNN | SVM |
| Before feature selection | Corrected | 0.780 | 10.187 | 1.015 | 3.612 | 10.545 | - |
| | 10_percent | 1.049 | 25.483 | 9.387 | 1.016 | - | - |
| After feature selection | Corrected | 0.032 | 5.514 | 0.325 | 5.216 | 4.256 | 95.596 |
| | 10_percent | 0.047 | 15.575 | 3.062 | 0.22 | 141.369 | - |

| Table 4 | Time taken in testing (in seconds) on different classifiers on both kdd data types |
|---------|---------------------------------|-----------------|---------------|-------------|-------------|
| Outcomes on classifiers | Dataset | Decision Tree | MLP | Random Forest | K-Means | KNN | SVM |
| Before feature selection | Corrected | 0.311 | 2.525 | 2.020 | 0.908 | 1014.176 | - |
| | 10_percent | 0.193 | 2.267 | 10.022 | 176.238 | 2789 | - |
| After feature selection | Corrected | 0.111 | 2.029 | 1.753 | 579.253 | 584.273 | 287.393 |
| | 10_percent | 0.064 | 1.643 | 8.306 | 2763 | 514.087 | 1410.522 |
Table 5  Comparison between several feature selection algorithms using Kddcup99 in terms of accuracy

| Approach                  | References                     | No. of Feature Selected | KddCup99 dataset          | Classifier used | Accuracy (in %) |
|---------------------------|--------------------------------|-------------------------|---------------------------|-----------------|-----------------|
|                           |                                |                         | Training                  | Testing         |                 |
| LSSVM                     | Ambusaidi et al., [63]         | 6                       | “10%”                     | “10%”           | DT              | 92.80           |
| Cuttlefish                | Eesa et al., [13]              | 5                       | “10%”                     | “10%”           | DT              | 91.99           |
| GA                        | Aslahi-Shahri et al., [64]     | 10                      | “Corrected”               | “Corrected”     | DT              | 94.00           |
| SVM                       | Keshtgary et al., [65]         | 10                      | “Corrected”               | “Corrected”     | DT              | 94.80           |
| Cuttlefish                | Mohammadi et al., [66]         | 10                      | “10%”                     | “Corrected”     | DT              | 95.05           |
| FGLCC                     | Mohammadi et al., [66]         | 16                      | “10%”                     | “Corrected”     | DT              | 92.59           |
| SigmoidPIO                | Hadeel Alazzam et al., [67]    | 10                      | “10%”                     | “10%”           | DT              | 94.70           |
| CosinePIO                 | Hadeel Alazzam et al., [67]    | 7                       | “10%”                     | “10%”           | DT              | 96.00           |
| Proposed Feature Selection method | Proposed approach | 5                       | “Corrected”               | “Full dataset”  | DT              | 95.53           |
|                           |                                |                         | Training                  | Testing         | MLP             | 93.28           |
|                           |                                |                         |                           |                 | Random Forest   | 92.87           |
|                           |                                |                         |                           |                 | K-Means          | 96.70           |
|                           |                                |                         |                           |                 | KNN             | 96.90           |
|                           |                                |                         |                           |                 | SVM             | 92.62           |
|                           |                                |                         |                           |                 | DT              | 96.682          |
|                           |                                |                         |                           |                 | MLP             | 97.014          |
|                           |                                |                         |                           |                 | Random Forest   | 95.707          |
|                           |                                |                         |                           |                 | K-Means          | 97.494          |
|                           |                                |                         |                           |                 | KNN             | 97.494          |
|                           |                                |                         |                           |                 | SVM             | 94.836          |
to compute at first, but it improved substantially after feature selection. Following feature selection. After feature selection, it was discovered that SVM could be trained and tested.

4 Conclusion and Future Work

In this paper, we emphasized the findings as a research need and addressed their solution. After analysis of the survey, it was found that most of the methods are inefficient in accurate classification along with reduced time overhead for intrusion detection. This research was intent towards solution of existing issues by proposing an optimal feature selection method using CART. The outcomes of our experimental works indicate how the trend in the use of various algorithms for the creation of IDS has changed from pure DT to ML and other hybrid approaches. It has also been observed that dimensionality reduction is strongly connected to time complexity and improved true classification accuracy. The proposed work presents a novel approach for optimum feature selection, through which we obtained an optimal collection of features, which is then passed over to numerous classifiers and employed to identify various attacks within the classes. The benefits of this proposed work include that feature selection is no longer dependent on classifiers. It is evident from the experimental results analysis that the outcomes of our work were determined to be acceptable for the majority of the DM, ML and hybrid-based classifiers. A classifier like K-Means performed badly before to feature selection, but its performance has substantially improved after feature selection. Some classifiers, such as SVM, were unable to run on such huge datasets prior to feature selection but the classifier is now able to run and perform better after feature selection. From Table 5, we observed that performance through our proposed Feature Selection method on almost every DM and ML classifiers techniques found at par or enhanced as compared with others research proposals. Assessment through the comparative analysis and simulation’s outcome concludes that ML as well as hybrid approaches play a vital role in design of robust IDS to secure computer network.

This proposal’s major advantage is its ease of use and consistency in performance. As a consequence of this simple proposed method, further work on any dataset can be accomplished and the reduced dataset may be returned before feeding the dataset to the classifier to address other many difficulties in improving the performance of classifying algorithms. In addition, it was assumed that real-time intrusion detection at a low cost would be a demand for IDS to safeguard computer networks.

Acknowledgements The authors are grateful to the learned anonymous reviewers whose insightful remarks helped us to improve the quality of this paper.

Funding The author(s) received no financial support for the research, authorship, and/or publication of this paper.

Availability of Data and Material The data that support the findings of this study are available at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

Declarations

Conflicts of interests The author(s) declares that there is no conflicts of interest with respect to the research, authorship and/or publication of this paper anywhere.
References

1. Keerthi Vasan, K., & Surendiran, B. (2016). Dimensionality reduction using principal component analysis for network intrusion detection. *Perspectives on Science, 8*, 510–512. https://doi.org/10.1016/j.pisc.2016.05.010

2. Kuang, F., Zhang, S., Jin, Z., & Xu, W. (2015). A novel SVM by combining kernel principal component analysis and improved chaotic particle swarm optimization for intrusion detection. *Soft Computing, 19*(5), 1187–1199. https://doi.org/10.1007/s00500-014-1332-7

3. Salo, F., Nassif, A. B., & Essex, A. (2019). Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection. *Computing Networks, 148*, 164–175. https://doi.org/10.1016/j.comnet.2018.11.010

4. Sameera, N., & Shashi, M. (2020). Encoding approach for intrusion detection using PCA and KNN classifier. In *Proceedings of the Third International Conference on Computational Intelligence and Informatics*, pp. 187–199.

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

6. Hoque, N., Bhattacharyya, D. K., & Kalita, J. K. (2014). MIFS-ND: A mutual information-based feature selection method. *Expert Systems with Applications, 41*(14), 6371–6385. https://doi.org/10.1016/j.eswa.2014.04.019

7. Inbarani, H. H., Bagyamathi, M., & Azar, A. T. (2015). A novel hybrid feature selection method based on rough set and improved harmony search. *Neural Computing and Applications, 26*(8), 1859–1880. https://doi.org/10.1007/s00521-014-1840-0

8. Chen, R.C., Cheng, K.F., Chen, Y.H., & Hsieh, C.F. (2009). Using rough set and support vector machine for network intrusion detection system. In *Proceedings—2009 1st Asian Conference on Intelligent Information and Database Systems, ACIIDS 2009*, pp. 465–470. https://doi.org/10.1109/ACIIDS.2009.59.

9. Stein, G., Chen, B., Wu, A. S., & Hua, K. A. (2005). Decision tree classifier for network intrusion detection with GA-based feature selection. *Proceedings of Annual Southeast Conference, 2*, 2136–2141. https://doi.org/10.1145/1167253.1167288

10. Zhu, Y., Liang, J., Chen, J., & Ming, Z. (2017). An improved NSGA-III algorithm for feature selection used in intrusion detection. *Knowledge-Based System, 116*, 74–85. https://doi.org/10.1016/j.knosys.2016.10.030

11. Gauthama Raman, M. R., Somu, N., Kirthivasan, K., Liscano, R., & Shankar Sriram, V. S. (2017). An efficient intrusion detection system based on hypergraph - Genetic algorithm for parameter optimization and feature selection in support vector machine. *Knowledge-Based System, 134*, 1–12. https://doi.org/10.1016/j.knosys.2017.07.005

12. Azad, C. & Jha, V.K. (2019). Decision tree and genetic algorithm based intrusion detection system. In *Proceeding of the Second International Conference on Microelectronics, Computing & Communication Systems (MCCS 2017)*, pp. 141–152. https://doi.org/10.1007/978-981-10-8234-4_13.

13. Eesa, A. S., Orman, Z., & Brifcani, A. M. A. (2015). A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems. *Expert Systems with Applications, 42*(5), 2670–2679. https://doi.org/10.1016/j.eswa.2014.11.009

14. Ambusaidi, M. A., He, X., Nanda, P., & Tan, Z. (2016). Building an intrusion detection system using a filter-based feature selection algorithm. *IEEE Transactions on Computers, 65*(10), 2986–2998. https://doi.org/10.1109/TC.2016.2519914

15. Bostani, H., & Sheikhhan, M. (2017). Hybrid of binary gravitational search algorithm and mutual information for feature selection in intrusion detection systems. *Soft Computing, 21*(9), 2307–2324. https://doi.org/10.1007/s00500-015-1942-8

16. Li, Y., Xia, J., Zhang, S., Yan, J., Ai, X., & Dai, K. (2012). An efficient intrusion detection system based on support vector machines and gradually feature removal method. *Expert Systems with Applications, 39*(1), 424–430. https://doi.org/10.1016/j.eswa.2011.07.032

17. Lin, S. W., Ying, K. C., Lee, C. Y., & Lee, Z. J. (2012). An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection. *Applied Soft Computing Journal, 12*(10), 3285–3290. https://doi.org/10.1016/j.asoc.2012.05.004

18. Mukherjee, S., & Sharma, N. (2012). Intrusion detection using naive bayes classifier with feature reduction. *Procedia Technology, 4*, 119–128. https://doi.org/10.1016/j.protcy.2012.05.017

19. Siddique, K., Akhtar, Z., Khan, M. A., Jung, Y. H., & Kim, Y. (2018). Developing an intrusion detection framework for high-speed big data networks: A comprehensive approach. *KSI Transactions on Internet and Information Systems, 12*(8), 4021–4037. https://doi.org/10.3837/itiis.2018.08.026
20. Zhang, Y., Li, Y., Zhang, T., Gadosey, P. K., & Liu, Z. (2018). Feature clustering dimensionality reduction based on affinity propagation. Intelligent Data Analysis, 22(2), 309–323. https://doi.org/10.3233/IDA-163337
21. Talavera, L. (2000). Dependency-based feature selection for clustering symbolic data. Intelligent Data Analysis, 4(1), 19–28. https://doi.org/10.3233/ida-2000-4103
22. Depren, O., Topallar, M., Anarim, E., & Çiliz, M. K. (2005). An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. Expert Systems with Applications, 29(4), 713–722. https://doi.org/10.1016/j.eswa.2005.05.002
23. Chebrolu, S., Abraham, A., & Thomas, J. P. (2005). Feature deduction and ensemble design of intrusion detection systems. Computers & Security, 24(4), 295–307. https://doi.org/10.1016/j.cose.2004.09.008
24. Guan, Y., Ghorbani, A. A., & Belacel, N. (2003). Y-means: A clustering method for intrusion detection. Can. Conf. Electr. Comput. Eng., 2, 1083–1086. https://doi.org/10.1109/ccee.2003.1226084
25. Peddabachigari, S., Abraham, A., Grosan, C., & Thomas, J. (2007). Modeling intrusion detection system using hybrid intelligent systems. Journal of Network and Computer Applications, 30(1), 114–132. https://doi.org/10.1016/j.jnca.2005.06.003
26. Sithatha Sindhu, S. S., Geetha, S., & Kannan, A. (2012). Decision tree based light weight intrusion detection using a wrapper approach. Expert System Application, 39(1), 129–141. https://doi.org/10.1016/j.eswa.2011.06.013
27. Kim, G., Lee, S., & Kim, S. (2014). A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. Expert System Application, 41(4), 1690–1700. https://doi.org/10.1016/j.eswa.2013.08.066
28. Puthran, S., & Shah, K. (2016). Intrusion detection using improved decision tree algorithm with binary and quad split. Communications in Computer and Information Science, 625, 427–438. https://doi.org/10.1007/978-981-10-2738-3_37
29. Chew, Y. J., Ooi, S. Y., Wong, K.-S., & Pang, Y. H. (2020). Decision tree with sensitive pruning in network-based intrusion detection system. Computer Science and Technology. https://doi.org/10.1007/978-981-15-0058-9_1
30. Ho, T.K. (1995). Random decision forests. in Proceedings of 3rd international conference on document analysis and recognition, vol. 1, pp. 278–282. https://doi.org/10.1109/ICDAR.1995.598994.
31. Kim, D. S., Lee, S. M., & Park, J. S. (2006). Building lightweight intrusion detection system based on random forest. International Symposium on Neural Networks, 3973, 224–230. https://doi.org/10.1007/11760191_33
32. Zhang, J. & Zulkernine, M. (2006). A hybrid network intrusion detection technique using random forests. in First International Conference on Availability, Reliability and Security (ARES’06), pp. 262–269. https://doi.org/10.1109/ARES.2006.7
33. Elbasiony, R. M., Sallam, E. A., Eltobely, T. E., & Fahmy, M. M. (2013). A hybrid network intrusion detection framework based on random forests and weighted k-means. Ain Shams Engineering and Journal, 4(4), 753–762. https://doi.org/10.1016/j.aesej.2013.01.003
34. Del Río, S., López, V., Benítez, J. M., & Herrera, F. (2014). On the use of MapReduce for imbalanced big data using Random Forest. Information Science (Ny), 285(1), 112–137. https://doi.org/10.1016/j.ins.2014.03.043
35. Barbara, D., Wu, N. & Jajodia, S. (2001). Detecting novel network intrusions using bayes estimators. in Proceedings of the 2001 SIAM International Conference on Data Mining, pp. 1–17.
36. Koc, L., Mazzuchi, T. A., & Sarkani, S. (2012). A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier. Expert Systems with Applications, 39(18), 13492–13500. https://doi.org/10.1016/j.eswa.2012.07.009
37. Wang, Y., et al. (2018). A novel intrusion detection system based on advanced Naive Bayesian classification. Lecture Notes in Institute for Computer Science and Social Telecommunication Engineering LNICST, 211, 581–588.
38. Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. American Statistician, 46(3), 175–185. https://doi.org/10.1080/00031305.1992.10475879
39. Liu, Y., Chen, K., Liao, X., & Zhang, W. (2004). A genetic clustering method for intrusion detection. Pattern Recognition, 37(5), 927–942. https://doi.org/10.1016/j.patcog.2003.09.011
40. Khan, L., Awad, M., & Thuraisingham, B. (2007). A new intrusion detection system using support vector machines and hierarchical clustering. The VLDB Journal, 16(4), 507–521. https://doi.org/10.1007/s00778-006-0002-5
41. Om, H. & Kundu, A. (2012). A hybrid system for reducing the false alarm rate of anomaly intrusion detection system. 2012 1st International Conference on Recent Advance Information Technology RAIT-2012, pp. 131–136. https://doi.org/10.1109/RAIT.2012.6194493.
42. 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 System, 78(1), 13–21. https://doi.org/10.1016/j.knosys.2015.01.009

43. Nguyen, T. T. T., & Armitage, G. (2008). A survey of techniques for internet traffic classification using machine learning. IEEE Communication Survey Tutorials, 10(4), 56–76. https://doi.org/10.1109/SURV.2008.080406

44. Cortes, C., & Vapnik, V. (1995). Support-vector networks.

45. Mukkamala, S., Janoski, G., & Sung, A. (2002). Intrusion detection using neural networks and support vector machines. Proceedings of International Joint Conference Neural Networks, 2, 1702–1707. https://doi.org/10.1109/ijcnn.2002.1007774

46. Al-Yaseen, W. L., Othman, Z. A., & Nazri, M. Z. A. (2017). Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. Expert Systems with Applications, 67, 296–303. https://doi.org/10.1016/j.eswa.2016.09.041

47. Borkar, G. M., Patil, L. H., Dalgade, D., & Hutke, A. (2019). A novel clustering approach and adaptive SVM classifier for intrusion detection in WSN: A data mining concept. Sustainable Computing: Informatics and Systems, 23, 120–135. https://doi.org/10.1016/j.suscom.2019.06.002

48. Lippmann, R. P., & Cunningham, R. K. (2000). Improving intrusion detection performance using keyword selection and neural networks. Computer Networks, 34(4), 597–603. https://doi.org/10.1016/S1389-1286(00)00140-7

49. Ryan, J., Lin, M.J., & Miikkulainen, R. (1998). lntrusion detection with neural networks. Advance Neural Information Processing System, pp. 943–949.

50. Powers, S. T., & He, J. (2008). A hybrid artificial immune system and Self Organising Map for network intrusion detection. Information Science (Ny), 178(15), 3024–3042. https://doi.org/10.1016/j.ins.2007.11.028

51. Zhang, C., Jiang, J., & Kamel, M. (2005). Intrusion detection using hierarchical neural networks. Pattern Recognition Letter, 26(6), 779–791. https://doi.org/10.1016/j.patrec.2004.09.045

52. Liu, G., Yi, Z., & Yang, S. (2007). A hierarchical intrusion detection model based on the PCA neural networks. Neurocomputing, 70(7–9), 1561–1568. https://doi.org/10.1016/j.neucom.2006.10.146

53. Sung, A.H., & Mukkamala, S. (2003). Identifying important features for intrusion detection using support vector machines and neural networks department of computer science new Mexico institute of mining and technology. Symposium A Q. J. Model Foreign Lit., pp. 3–10.

54. Wang, G., Hao, J., Mah, J., & Huang, L. (2010). A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering. Expert Systems with Applications, 37(9), 6225–6232. https://doi.org/10.1016/j.eswa.2010.02.102

55. Vancha, M.R., Prakash Reddy, I.R., & Reddy, K.A. (2019). Intrusion detection system using multilayer neural networks, vol. 11, no. 5. Springer, Singapore.

56. Akashdeep, I. M., & Kumar, N. (2017). A feature reduced intrusion detection system using ANN classifier. Expert System Application, 88, 249–257. https://doi.org/10.1016/j.eswa.2017.07.005

57. Chandak, T., Shukla, S., & Wadhvani, R. (2017). An analysis of ‘A feature reduced intrusion detection system using ANN classifier’ by Akashdeep et al. expert systems with applications. Expert Systems with Applications, 130, 79–83. https://doi.org/10.1016/j.eswa.2019.04.017

58. Ambusaidi, M.A., He, X., Tan, Z., Nanda, P., Lu, L.F., & Nagar, U.T. (2015). A novel feature selection approach for intrusion detection data classification. Proceedings—2014 IEEE 13th International Conference Trust Security Private Computing Communication Trust 2014, pp. 82–89. https://doi.org/10.1109/TrustCom.2014.15.
64. Aslahi-Shahri, B. M., et al. (2016). A hybrid method consisting of GA and SVM for intrusion detection system. *Neural Computing and Applications*, 27(6), 1669–1676. https://doi.org/10.1007/s00521-015-1964-2

65. Khalvati, L., Keshtgary, M., & Rikhtegar, N. (2018). Intrusion detection based on a novel hybrid learning approach. *Journal of AI data Mining*, 6(1), 157–162. https://doi.org/10.22044/JADM.2017.979

66. Mohammadi, S., Mirvaziri, H., Ghazizadeh-Ahsae, M., & Karimipour, H. (2019). Cyber intrusion detection by combined feature selection algorithm. *Journal of Information Security Application*, 44, 80–88. https://doi.org/10.1016/j.jisa.2018.11.007

67. Alazzam, H., Sharieh, A., & Sabri, K. E. (2020). A feature selection algorithm for intrusion detection system based on Pigeon Inspired Optimizer. *Expert System Application*. https://doi.org/10.1016/j.eswa.2020.113249

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Neeraj Kumar** is research scholar at Birla Institute of Technology mesra - Extension Centre Patna, India and presently working as a Principal Incharge, Purnea College of Engineering, Purnea, Govt. of Bihar, under Department of Science & Technology, Patna, India. His research work involves using Data Mining, Network Security, Machine Learning techniques to improve the performance of any algorithm in Computer Vision. Contributed papers in the field of Data Mining, Network Security, Machine Learning, Distributed System, MANET and Soft Computing. He had nineteen years of experience in teaching and research and life time member of Computer Society of India.

**Dr. Upendra Kumar** received his Master in Technology and Doctor in Philosophy from Birla Institute of Technology, Mesra Ranchi in the Department of Computer Science and Engineering respectively. He is currently working as an Assistant Professor in the same Department from last 10 years. His research field is related with Computer Network Security and Data Mining. He has published more than 25 research papers in Journal of repute and Conferences and apart of this he is also reviewer of several Journals of Computer Science and member of International Conference of ACM, IEEE and Springer.