IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 7, 2022, accepted February 8, 2022, date of publication February 18, 2022, date of current version March 16, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3152963

Practically Implementation of Information Loss: Sensitivity, Risk by Different Feature Selection Techniques

AMJAD ALI[1], FAIZ JILLANI[1], RABBIAH ZAHEER[2], AHMAD KARIM[1], YASSER OBAID ALHARBI[3], MOHAMMAD ALSAFFAR[3], AND KHALID ALHAMAZANI[3]

[1]Department of Information Technology, Bahauddin Zakariya University, Multan, Punjab 60000, Pakistan
[2]Department of Physics, Bahauddin Zakariya University, Multan, Punjab 60000, Pakistan
[3]College of Computer Science and Engineering, University of Hail, Ha'il 55255, Saudi Arabia

Corresponding author: Ahmad Karim (ahmadkarim@bzu.edu.pk)

ABSTRACT The modern perspective to deal with bulky data generations schemes of latest technologies in terms of dimensionality and sample size to extract meaningful information also to support automated knowledge discovery and pattern recognition process form datasets a lot of Data Mining (DM) and Machine Learning (ML) techniques developed. In each dataset features are the key factors for machine learning task. In modern research mindset classification algorithms are focused to get high accuracy by taking in account prior features and less focus on features having low characteristic values. In this paper we focused on those features which are usually ignored in selection phase as low scale features which may decrease model performance but in future for the most sensitive scenarios will focus on minor information which will alert about performance fluctuation in practical implementation of that model. For practical verification of our concept, we implemented rule-based classification algorithms and different features selection techniques with 3 search methods using WEKA data mining tool. The experimental results show that the fewer selected features provide high accuracy i.e > 90% in some cases having less focus on specificity.

INDEX TERMS Features, features selection techniques, rule-based classification algorithms, minor information, machine learning.

I. INTRODUCTION

Feature selection an emerging pane is under prior consideration for learning and implementation paradigms in the vast domains of data mining (DM), machine learning (ML) and probabilistic statistics (PS). As with the robust pace for development in ML the interest for features selection (FS) research has gained much importance because of its practical involvement in many applications broadening day by day. The core purpose of features selection involves dimensionality reduction by means of irrelevant, redundant and imbalanced values removal to attain appropriate features set [1], [2]. The availability of data tends to focus on its features which have capability for its considerable representation and this capability of features distinguish them from each other. The process of features selection is dependent to computational tasks which are the challenging phase for any selection criterion. Despite the complex computation most of the researchers focus on features not on datasets. Features selection have instant impact on algorithm's efficiency, accuracy and extending model feasibility but the problem is still there that how to highlight these effective features among the all [3] and this problem further leads the idea which follows that the features selection criterion is still confusing paradigm. The data mining methods are dependent to features selection pro-phase before any further implication and that phase is pre-processing aiming with dimensionality reduction for data on the basis of features relevance and priorities of described model. The success story of features selection is based on nature of selection metrics regarding to our utilization keeping in mind that either selected approach fits to acquired standards and also understandability of core knowledge about considered domain. The adoptability of optimal features selection technique assists in redundancy and irrelevance removal in terms of features

The associate editor coordinating the review of this manuscript and approving it for publication was Nazar Zaki.

to obtain excellent features for high performing phases. The selection out comes helps in improvement of learning rate and accuracy [4], complexity level and learning period reduces also simple understandable results representation only if optimal features selection carried out [5], [6]. Dimensionality reduction implemented by two different ways one is features extraction and other is features selection [7]–[9]. The reduction is an essential part for learning algorithms to perform its best by removal of noisy data. In supervised learning the main problem is the identification of these data values inside the datasets, measurement of degree of relevance of each attribute with its class as it is in some context representor of that particular class. Dimensionality reduction removes noise and imbalance problems and data cleaned from problematic features which diminish the model's performance but after that reduction model have the availability of pure features and it can perform well as expected. Now all the irrelevant features have been removed and datasets only comprised of relevant features. In the recent advancement in features selection domain different approaches for selection procedure are followed as selection of features according to selected classification algorithms [10]. The other features selection perspective is selection of features according to prescribed problem. The core objective for the features involves understandability of prior factors that on predictive analysis of problem requirements fits to selection criterion to build up a model in terms of classifiable metrics. Model is generated using that selected features and provides problem solution. In selection two factors about features are brought into consideration which are sensitivity and specificity [11]. The split approach for features adopted and to carry out this split a difference metrics is defined in terms of threshold. The threshold is defined on the basis of probability scale of feature values ranging from 0 to 1. Keeping in mind such paradigms there exists a strong chance for some features in a formal way may not fulfill selection standards but these features with minor sensitive information are much valuable for any classification task.

In this paper we worked on deselected features during selection process only because of their low relevance to under consideration selection criterion where only high relevance features treated for model generation. These mishandled features are not the features which are missing or belong to some imbalance class but actually such features belongs to the data repository and best fits in class label due to the presence of high valued features they remained untouched for any learning process. To highlight such features we selected two important datasets one from UCI [12] namely spam-base and other DARPA evaluation [13] dataset namely an extension of KDD-99 as NSL-KDD to support our practical approach. We applied three most important features selectin techniques CFS-subset-Eval with two search method 1) Bes First, 2) Greedy Stepwise, second features selection technique applied is Info-Gain Attribute with Ranker search method and third used filter supervised attribute selection. To evaluate the performance of these selection techniques we use two rule-based classifiers Decision Table and PART. We considered different evaluation metrics including classifier's accuracy, Root Mean Square Error (RMSE), True Positive rate (TRP) and F-measures. Then provided the classifier different number of features as well as full features set to be used for obtaining model performance. The experimental results show that when model is provided with different features for model generation of the same dataset its performance metrics changed that is due to unavailability of same features in each phase and such happens due to minor loss of information that unselected features contains.

The rest of the paper is organized as follows: Section II represents related work, Section III and IV discussed various feature selection techniques and algorithms, finally the paper is concluded in section V.

II. RELATED WORK

In the recent advancements in machine learning environment there are a lot methods tools introduced to deal with simple as well as complex problems, so the datasets with usual arrangements and managements are not a case of intentions because with the availability of automated process to handle such data. This eased the manual efforts and broadened the thinking towards a new concept where main focus about data utilization for future trends and it is not about data understandability because this phase shifted to automated tools processing. In the same context handling the features is not a big deal for a new learner with a little knowledge about tools and data valuation because technology advancements have made knowledge access an easy task in terms of written, published, video tutorials etc. A lot of features selection techniques have been introduced so far and everyone who intended to deal with features and their implementation getting hand rubbed with these techniques as considering them an authentic source without taking any confusion mind. Every techniques as it is developed performing its best with the provision of optimal features to machine learning algorithms. There exists different selection parameters for each selection technique each select some different features from the same dataset under different circumstances. That area of difference is still confusing as for prior knowledge some features adopted and some balanced features dropped. All these paradigms are brought into implementation to obtain higher accuracy and every factor that can improve accuracy considered at high priority. Same happens in the case of features only those selected which can uplift accuracy so some important with minor information carriers remained unconsidered.

According to [14] Al-Janabi there exists a lot of methods to handle such minor information in any datasets by taking design as core concept and introduced different decomposition levels for data representation also for retrieval introduced revised key structures. He mainly focused on dimensional view of attributes so that they in some dimension applicable for model efficiency. He also focused on data gathering algorithms which used for information retrieval from datasets and intended to improve retrieval code so that every information from each aspect can be collected and then no further processing required to retain information. Though this was a good idea to handle such minor features values but t lacks proficiency. This model proposal lacks of some important selection parameters for information which are out of proportional bound because this was based on proportionality constructors which involve average leading parameters whose performance usually doesn't adopt below average values. Bruggeman *et al.* [15] carried out an efficient solution for web contents that unintentionally left unchecked by webserver and in prior phase of information retention those unchecked information kept unseen, so the effective information obtaining doesn't compete there. Though it can fulfill the query information and up to some content provide understandability about topic. To overcome this problem they suggested webserver namely phylopars on the basis of statistics consistency in which they adopted merge technology for one dataset to a general dataset to overcome its deficiencies in values proficiency by adjusting estimated weight. In [16] authors made the same approach for forecast prediction to replace such information with new one of high priority by taking post-processing phase for this addition implementation. Though this was good idea to make minor information at higher scale by some updating original ones but due to improper explanation for implementation no further improvement can be made. In [17] authors followed a model which on similarities base at classification phase measure that either those features pair interact with each other not. They constructed a random Forest model by utilizing information about features both in direct a well as indirect pane by considering training examples. Thus by this similarity comparison idea they used maximum information as compared to any other model and their model accuracy also improved.

Many researchers tried to work with the same dataset KDD-99 that we selected and they too faced these selection problems for features. In [18] Sung *et al.* only enabled themselves to select 19 features from full feature set with the help of trial and error based approaches. In the same context Chebrolu *et al.* [19] implemented BN (Bayesian Network) and only managed to identify 17 features and all other remained undetected for this process. In [19] they also applied a classifier namely CART and managed to identify only 12 features less than the BN classifier. Though they got enough features for each class but one of the class still remained missing with its features too but in the same time with selected features accuracy improved [20]. There still minor information about unselected features remained aside which certainly can increase model's effectiveness. In [21], [22] authors Alexandar *et al.* implemented Genetic Algorithm (GA) for Radial Basis Function (RBF) used network to select features and optimize its architecture. The main concern for this implementation was to provide network specific features and also select those features which are necessary the architecture of network. So there comes the ranking problem that each network work on different features and same features can't be used for each network either these are neural networks (NNs) or support vector machine (SVM). This method also acquired more time and resources for training section as well as for cross-validation.

III. FEATURES SELECTION TECHNIQUES AND CLASSIFICATION ALGORITHMS

The rapid advancements in the field of machine learning where benefited a lot there also generated large amount of data with enormous features which may cause overfitting and machine learning model performance degrades. There available high dimensional data where it is needed to reduce its dimensionality and various techniques for this task introduced which also gained the researchers attraction. Features selection is the broad concept for dimensionality reduction to select features from originally available data attributes by following some specified rules based on variable selection metrics for evaluation. These if truly selected can improve performance in terms of leaning accuracy, lessens processing budget and upgrades model effectiveness.

IV. FEATURES SELECTION TECHNIQUES

A. CFS-SUBSET-EVAL

Correlation based Features Selection (CFS) is purely automated technique to select features. The interaction for evaluation function is quite simple that it selects those features that are at high instance correlated with the described class and at high instance uncorrelated with each other which means that their class relevance is tested and inter relevance removed. The features for an instance considered selected by CFS if it predicts a unique correlation value which is not yet predicted by any other feature. The selected features are compared to the merit features K as in first equation where coefficient namely Pearson is pass through estimation phase for training data samples with correlation calculation in Xi and Xj as features of same class or different as

$$R_{x(i)x(j)} = \frac{\sum_k^m (x_k^i - \bar{x^i})(x_k^j - \bar{x^j})}{(m-1)S_iS_j}$$

$$M_s = \frac{k\bar{r_{cf}}}{\sqrt{k + k(k-1)\bar{r_{ff}}}}$$

where Ms is a sort of merit with S features having k samples in it. In the equation Krcf represents an average correlation for feature class and rff is also a correlation which represents average for features with features within the same class [31]–[33]. The upper portion for equation second represents the class prediction about features values and the lower part represents redundancy within the features. The CFS follows coupling mechanism in which features are defined on the basis of tightly or loosely coupled with class and with other features. If a features is latterly assigned to a class its dominance will be higher than other features and class as it lacks correlation attributes.

B. INFORMATION GAIN

This method follows threshold scale for features evaluation and all those features who satisfy threshold merits are further considered for classification [23]. It is based in the division of dataset values regarding to available variable random value or the entropy reduction is calculated. In the general estimated value about information gain (IG) contains routine values for desired variable of (X) and its adequate variable in independent phase as (A) where the variable A shows performance diminishing paradigm for entropy value according to variable X [24]. This technique may involve the risk of loss for minor information packets within the features. IG calculation is brought into consideration with X and Y attributes and the addition foe data values X according to Y to find out its value. In the same context the vulnerability of Y is accessed on X basis where cognitive resemblance for Y is X, H (Y|X).

$$I(Y:X) = H(Y) - H(Y/X)$$

where

$$X = \{x_i \ldots x_k\} \& Y = \{y_i \ldots y_k\}$$

To calculate the entropy values

$$\mathrm{H}(Y) = -\sum_{t=1}^{t=k} (\mathrm{Y} = \mathrm{yt})\, log2\mathrm{P}(\mathrm{Y} = \mathrm{yt})$$

Entropy dependency for Y in terms of X is as

$$\mathrm{H}(Y|X) = -\sum_{j=1}^{i} \mathrm{P}(\mathrm{X} = \mathrm{xj})\, H(\mathrm{Y}|\mathrm{X} = \mathrm{xj})$$

So the IG is obtained as:

$$I(Y:X) = H(Y) + H(X) - H(Y|X)$$

After that mixed entropy applied to find out combined factor for IG for both values X and Y is

$$(Y:X) = argmax_{x0} I(Y:X)$$

If the entropy values of mean entropy values is differentiated from the original entropy values in all branches then we can obtain improved results for IG.

$$IG(X, y) = H(X) - H(X|y)$$

In this section the *IG (S, y)* represents information for X dataset *with the involvement of simple or random variable values*, *H(X)* the entropy representation according to changed values for dataset also $H(X|y)$ is termed as entropy for conditional values in which variable Y as dataset representation. In this statement gained X values are shown related to y. For the further proceedings and shifting these values within the dataset stored.

C. FILTER SUPERVISED ATTRIBUTE SELECTION

This model has too much flexibility that prioritize it from other usual features selection algorithms. This method provides the combining feasibility of multiple features selection techniques and implementation of that combined approach as a single unit. This evaluates the attributes with the user provided configuration for any classifier. These configurational adaptability enables this method to work with user specified parameters and highlight only those features that fulfill the required performance scale. This algorithm carries multiple capabilities including attributes handling, class balancing and instance selection. In attributes it deals with binary attributes, empty nominal attributes, date attributes, missing values, numeric attributes, and unary attributes. In case of class it involves binary class, date class, missing class, nominal class and numeric class. In terms of instances it deals with very low number of instances often consider 0 instances. IR Class-Value – The class label, or 1-based index of the class label, to use when evaluating subsets with an IR metric (such as f-measure or AUC). Leaving this unset will result in the class frequency weighted average of the metric being used. It uses combinational approach to evaluate any measure.

D. SEARCH METHODS

There exists three commonly imposed search methods namely Best first, Greedy stepwise and ranker to search the features on their described patterns where each use different search strategy.

Best First: Searches the space of attribute subsets by greedy hill-climbing augmented with a backtracking facility. Setting the number of consecutive non-improving nodes allowed controls the level of backtracking done. Best first may start with the empty set of attributes and search forward, or start with the full set of attributes and search backward, or start at any point and search in both directions (by considering all possible single attribute additions and deletions at a given point). Set the maximum size of the lookup cache of evaluated subsets. This is expressed as a multiplier of the number of attributes in the data set. (default = 1).

Greedy Stepwise: Performs a greedy forward or backward search through the space of attribute subsets. May start with no/all attributes or from an arbitrary point in the space. Stops when the addition/deletion of any remaining attributes results in a decrease in evaluation. Can also produce a ranked list of attributes by traversing the space from one side to the other and recording the order that attributes are selected. Specify the number of attributes to retain. The default value (−1) indicates that all attributes are to be retained. Use either this option or a threshold to reduce the attribute set.

Ranker: Ranks attributes by their individual evaluations. Use in conjunction with attribute evaluators (ReliefF, Gain Ratio, Entropy etc.). Specify the number of attributes to retain. The default value (−1) indicates that all attributes are to be retained. Use either this option or a threshold to reduce the attribute set.

1) CLASSIFIERS

The availability of multiple classifiers acquires the deep knowledge about classifiers working mechanism understandability because each one performs differently according to the problem constraints. The top classifiers with different approaches [25] are given as

1. Bayesian Classifiers
2. Tree based classifiers

TABLE 1. Spam based dataset attributes description.

Sr. No	Type of values	Description
1	48 continuous real [0,100] attributes of type word_freq_WORD	= percentage of words in the e-mail that match WORD, i.e. 100 * (number of times the WORD appears in the e-mail) / total number of words in e-mail. A "word" in this case is any string of alphanumeric characters bounded by non-alphanumeric characters or end-of-string.
2	6 continuous real [0,100] attributes of type char_freq_CHAR]	= percentage of characters in the e-mail that match CHAR, i.e. 100 * (number of CHAR occurences) / total characters in e-mail
3	1 continuous real [1,...] attribute of type capital_run_length_average	= average length of uninterrupted sequences of capital letters
4	1 continuous integer [1,...] attribute of type capital_run_length_longest	= length of longest uninterrupted sequence of capital letter
5	1 continuous integer [1,...] attribute of type capital_run_length_total	= sum of length of uninterrupted sequences of capital letters = total number of capital letters in the e-mail
6	1 nominal {0,1} class attribute of type spam	= denotes whether the e-mail was considered spam (1) or not (0), i.e. unsolicited commercial e-mail.

3. Rule based classifiers
4. Function based classifiers
5. Lazy classifiers
6. Multi-instance classifiers
7. Ensemble classifiers as PostScript (PS),

a: RULE BASED CLASSIFIERS

The rule of IF-THEN logic enabled rule based classifiers strong recommendation in classification tasks. The ease of generation for IF-THEN rules as well as their simple interpretation gained much importance rather than any other classifiers. The main advantage for rule base classifiers is that they reduce errors ratio especially in case of FPR (False Positive Rate). Another advantage for using rule based classifiers is their performance compatibility in terms of comparison analysis with DT based classifiers (Decision Tree) [26]. These all supportive roles motivated us to adopt rule based classifiers. For that reason we used Decision Table and PART classifiers from rule based classifiers whose comprehended view is given here.

Decision Table: The table 1 shows sample space for every combination under each situation is mapped with minor information and activities to finalize a decision. In table 1 all entities are examined with prior intentions and thus finds out the uncovered aspects of features by applying nearest neighbor algorithm for class determination of each uncovered feature sample. It doesn't follow dominance of features in table but use features understandability of similarity indexes with other attributes. The preferred number of instances to process if batch prediction is being performed. More or fewer instances may be provided, but this gives implementations a chance to specify a preferred batch size. The number of decimal places to be used for the output of numbers in the model [27].

Part: It is a class which is used to generate a decision list as per rules of PART. PART follows the rules for common DT and implement C4.5 to generate a tree under the core rule in each iteration and makes the ‘‘best’’ leaf into a rule as per by J4.8 [27]. The preferred number of instances to process if batch prediction is being performed. More or fewer instances may be provided, but this gives implementations a chance to specify a preferred batch size.

2) PERFORMANCE AND EVALUATION METRICS

a: PERFORMANCES METRICS

The scalability regarding performance of our model's efficiency about the detection of optimal features is carried out by using different performance scales. For common understanding if there exists P positive and N negative examples in any dataset then its performance metrics can be termed as

1. TP: True positive
2. TN: True negative
3. FP: False positive
4. FN: False negative

b: EVALUATION METRICS

Evaluation scale for any classifier can be determined on the basis of following measures

Accuracy: It is usually termed how correctly classifier managed attributes. Its formula is given based on performance values calculation as

$$Accuracy = (TP+TN)/(P+N)$$

Error Rate: This rate usually calculates the misclassified values ratio regarding to correctly classified values. Its mathematical representation is given as

$$Errorrate = (FP+FN)/(P+N)$$

Recall: It explains detect ratio of positive values termed as truly classified. Recall = TP/P

Specificity: It is the value for true identification of negative values. Specificity =*TN/N*

Precision: it is termed as positively predicted values. It measures the relatively within the instances by counting it with selected instances. It's mathematical from is precision

$$=TP/(TP+FP)$$

F-Measures: The harmonic mean for precision and recall is termed as f-measures in which test accuracy is measured. It can be used both for binary class as well multiclass. Whenever there occurs imbalance class issues a simple and efficient technique f-measure is suggested. Its mathematical representation is denoted as

$$\text{F-measure} = (2 \times \text{precision} \times \text{recall})/(\text{precision} + \text{recall}).$$

Despite of these evaluation metrics some comparison metrics in terms of root mean square (RMSE), time taken to train the classifier, its learning efficiency, effectiveness, and understandability are also used also graphical and tabular view of values presented.

E. DATASETS

All these implementations only can perform if there is the availability of training and testing data values. These requirements are made possible by selection of two different datasets namely Spam-base dataset and NSL-KDD from UCI data repository.

The spam-base dataset was originally devoted by George Forman [28] to Hewlett-Packard labs and then publically made available at UCI data repository. The word spam is a broad concept which includes web spams, advertisements, adult contents, easy money websites etc. This dataset mainly explains email related spam data values from professional to personal level and tries to identify spam nature as well as type. It consists of 55-57 attributes whose description are given in table 1 in which attributes definitions are provided and support the frequently occurring data integers and characters.

Table 2 represents attributes division according to their data type and nature of values. This section also explains detailed descriptive working mechanism for each class. In the total given 56 there 46 belongs to same data type, 6 to another and remaining have their individual data type.

NSL-KDD was originally suggested to deal with some inherent issues that occurs in original KDD-99 dataset. The improved version still lacks some important features to respond modern network technology. Where this dataset have some data deficiencies it can still be used for network treatments. Though there still need to improve this dataset but beneficial in many ways. One of the main benefit is that it doesn't involve redundancy mapping values in training example that saves classifiers from biased approach for redundant values. This redundancy factors highlight some features which are set to the network to not learn. The entire NSL-KDD dataset [29] contains large amount of data instances but for the ease there is availability of 20% data values for training and testing by name of KDD-Train and KDD-Test. The KDD dataset contains 41 features whose description with their count rate is given in table 2

TABLE 2. NSL-KDD features.

Label s	Features	Label s	Features
1	Duration	22	is_guest_login
2	protocol type	23	Count
3	Service	24	srv_count
4	Flag	25	serror_rate
5	src_bytes	26	srv_serror_rate
6	dst_bytes	27	rerror_rate
7	Land	28	srv_rerror_rate
8	wrong_fragment	29	same_srv_rate
9	Urgent	30	diff_srv_rate
10	Hot	31	srv_diff_host_rate
11	num_failed_login's	32	dsdt_host_count
12	logged in	33	dst_host_srv_count
13	num_comromised	34	dst_host_same_srv_rate
14	root_shell	35	dst_host_diff_srv_rate
15	su_attempted	36	dst_host_same_src_port_ra te
16	num_root	37	dst_host_srv_diff_host_rat e
17	num_file_creation	38	dst_host_serror_rate
18	num_shells	39	dst_host_srv_serror_rate
19	num_access_files	40	dst_host_rerror_rate
20	num_outbound_cm ds	41	dst_host_srv_rerror_rate
21	is_host_login		

The table 2 represents 41 novel features of NSL-KDD dataset features where 29 out of 41 are training features and these are classified into four different attack DOS,Probe,R2Land U2R groups which are represented in table 3 with their optimal capabilities.

Although these are different attacks in nature and different vulnerabilities are keen interest by attackers in these attacks but up to some extent still traceable by their nature if effective implementation for their effects carried out. The features are further sub-divided into three groups namely basic features (TCP/IP based hard detective features), traffic features (window interval based features further subdivided into same host and same service features) and at the last third one is content features (conscious response from user, fail attempts for login etc.).

TABLE 3. **Attacks description.**

Sr. No	Attacks	Description
1	DOS (Denial Of Services)	Resources are turned busy by excessive traffic load by attacker that legitimate requests by user denied for system access.
2	Probe (Probing Attacks)	Attacker scan system for information collection also for network vulnerabilities and by utilizing network maps can utilize gathered information for exploitations.
3	R2L (Remote To Local Access)	Attacker use masking and requests access to network to be pretending as local user of that network and .apply vulnerabilities check to get access.
4	U2R (User To Root Access)	Attacker (by sniffs, social engineering, and dictionary-attack) get normal account access and tries vulnerabilities to get root access for the system.

TABLE 4. **Decision table results using CFS with best first (A).**

Sr. No	Features Selection Techniques	Search Method	Selected Features	No. of selected features
1	CFS-Subset-Eval	Best First	7,25,27,28,29,32,42, 46,52,53,57,34	12
2	CFS-Subset-Eval	Greedy Stepwise	23,5,7,44,16,21,4,55, 24,25,27,42,,46,52,5 3,	15
3	IG Attribute	Ranker	52,53,56,7,21,55,16, 24,57,25,23,27,5,19, 26,3,11,17,2,4	20
4	Filter Supervised Attr-Selection		57,18,19,17,28,16,20 ,21,22,23,26,25,24,1 5,14,13,6,4,3,2, 5,7,12	23

F. PERFORMANCE EVALUATION

The practical implementation of this study is carried out using data mining tool WEKA [30] which is available through general public license and have multiple cross-frameworks implementation capabilities originally developed in JAVA language and easy to use for new users with online availability of tutorials. Different model implementations can easily be carried out through this tool with its effective

G. SPAM BASED IMPLEMENTATION

We used different features selection algorithms with full training examples and top features by each selection algorithm the table 5 represents the selected features by each technique for spam-based dataset.

In figure 1, 2, 3 and 4 CFS denotes correlation based features selection, IG denotes information gain, BF denotes best first, GS denotes greedy stepwise, R denotes ranker and Filter.S Att.S denotes filter supervised attribute selection.

Figure 1 and 2 represents the performance metrics comparison of Decision Table and PART classifier respectively using different features selection techniques where mainly comparison of accuracy is considered using all vs selected features by applied technique. The comparison represents that an algorithm will all features not only generate a model with acceptable accuracy but also model have maximum

TABLE 5. **Attributes selection for spam-based dataset (A).**

Features Performance	Decision Table Classifier: CFS-Subset-Eval + Best First				
	Accuracy	TPR	FPR	F-Measures	RMSE
12	92.31 %	0.924	0.103	0.923	0.2773
ALL	87.39 %	0.874	0.173	0.871	0.3195

FIGURE 1. **Accuracy comparison for decision Table (A).**

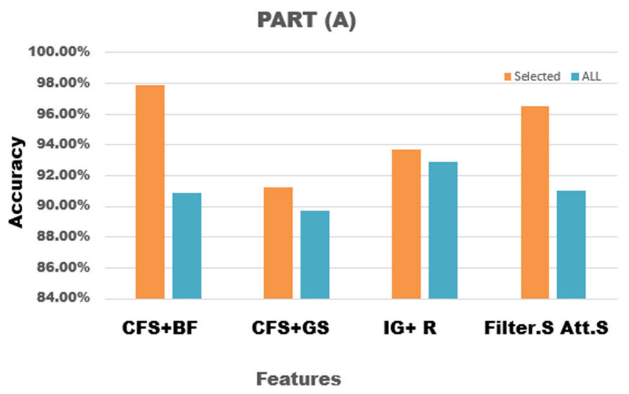

FIGURE 2. Accuracy comparison for PART (A).

FIGURE 3. Accuracy comparison for decision table (B).

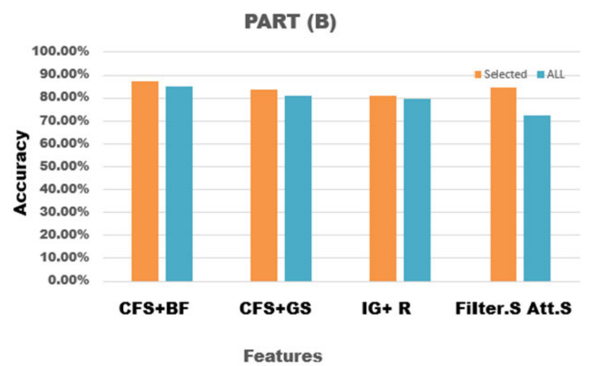

FIGURE 4. Accuracy comparison for part (B).

information which lacks in selected features. All this process is based on automated extraction of features for classification task. The accuracy for decision tree and PART classifier represents that if it is provided with all features (maximum available) it gives acceptable accuracy values which clearly states that more informative features more accurate results and better model performance can be obtained. This also indicates that a model with maximum training examples involves less risks for information loss as compared to some selected features.

TABLE 6. Decision table results using CFS with greedy stepwise (A).

Features Performance	Decision Table Classifier: CFS-Subset-Eval + Greedy Stepwise				
	Accuracy	TPR	FPR	F-Measures	RMSE
15	92. 39 %	0.894	0.140	0.892	0.2991
ALL	89.35 %	0.894	0.140	0.892	0.2991

TABLE 7. Decision table results using IG with ranker (A).

Features Performance	Decision Table Classifier: CFS-Subset-Eval + Greedy Stepwise				
	Accuracy	TPR	FPR	F-Measures	RMSE
15	92.39 %	0.894	0.140	0.892	0.2991
ALL	89.35 %	0.894	0.140	0.892	0.2991

TABLE 8. Decision table results using IG with ranker (A).

Features Performance	Decision Table Classifier: Information Gain + Ranker				
	Accuracy	TPR	FPR	F-Measures	RMSE
20	89.13 %	0.910	0.113	0.907	0.2451
ALL	87.52%	0.875	0.165	0.873	0.3066

TABLE 9. Decision table results using filter supervised attribute selection (A).

Features Performance	Decision Table Classifier: Filter Supervised attribute selection				
	Accuracy	TPR	FPR	F-Measures	RMSE
23	89.52%	0.896	0.139	0.894	0.2959
ALL	87.39 %	0.897	0.143	0.881	0.2818

NSL-KDD Implementation: We used different features selection algorithms with full training examples and top features by each selection algorithm the table 14 represents the selected features by each technique for NSL-KDD dataset.

TABLE 10. PART results using CFS with best first (A).

Features Performance	PART Classifier: CFS-Subset-Eval + Best First				
	Accuracy	TPR	FPR	F-Measures	RMSE
12	97.91 %	0.979	0.025	0.979	0.1371
ALL	90.87 %	0.909	0.099	0.909	0.2893

TABLE 11. PART results using CFS with greedy stepwise (A).

Features Performance	PART Classifier: CFS-Subset-Eval + Greedy Stepwise				
	Accuracy	TPR	FPR	F-Measures	RMSE
15	91.26 %	0.992	0.031	0.984	0.1121
ALL	89.71 %	0.937	0.107	0.920	0.2123

TABLE 12. PART results using IG with ranker (A).

Features Performance	PART Classifier: Information Gain + Ranker				
	Accuracy	TPR	FPR	F-Measures	RMSE
20	93.69 %	0.937	0.072	0.937	0.2325
ALL	92.90 %	0.929	0.081	0.929	0.2579

TABLE 13. PART results using filter supervised attribute selection (A).

Features Performance	PART Classifier: Filter supervised attribute selection				
	Accuracy	TPR	FPR	F-Measures	RMSE
23	96.54 %	0.965	0.033	0.966	0.0501
ALL	90.99 %	0.910	0.109	0.910	0.1077

The figure 3 and 4 are about performance comparison of different features selection techniques when applied using Decision Tree and PART classifiers. As selected features have high accuracy this is the not an annoying case as in section1 we already stated that features selection techniques select fewer features for classification which definitely enhance model performance but in our case as they have less features to use for classification, so they have less training information as compared to whole features set with full information metrics. So, the accuracy in not main concern in such cases, extracted information is the main concern which in full set is always greater than selected ones. So whenever such cases occurs it states that loss of some important information where can decrease model performance it also enhance model sensitive utilization where minor factors are not ignorable in most of cases.

TABLE 14. Attributes selection for NSL-KDD dataset.

Sr. No	Features Selection Techniques	Search Method	Selected Features	No. of selected features
1	CFS-Subset-Eval	Best First	5,6,25,28,12,30,31,35,36,4	10
2	CFS-Subset-Eval	Greedy Stepwise	9,10,8,19,38,30,35,5,36,37,4	11
3	IG Attribute	Ranker	25,38,24,30,35,29,23,39,33,6,3,5,34, 26,36,12,37,4,32,2	20
4	Filter Supervised Attr-Selection		12,36,40,5,6,38,10,2,29,30,35,3,37,8, 4	15

TABLE 15. Decision table results using CFS with best first (B).

Features Performance	Decision Table Classifier: CFS-Subset-Eval + Best First				
	Accuracy	TPR	FPR	F-Measures	RMSE
10	97.49 %	0.975	0.028	0.975	0.1417
ALL	96.02 %	0.960	0.048	0.960	0.1724

TABLE 16. Decision table results using CFS with greedy stepwise (B).

Features Performance	Decision Table Classifier: CFS-Subset-Eval + Greedy Stepwise				
	Accuracy	TPR	FPR	F-Measures	RMSE
11	96.67 %	0.967	0.038	0.967	0.1610
ALL	95.02 %	0.950	0.058	0.950	0.1724

TABLE 17. Decision table results using IG with ranker (B).

Features Performance	Decision Table Classifier: Information Gain + Ranker				
	Accuracy	TPR	FPR	F-Measures	RMSE
20	79.43 %	0.712	0.231	0.712	0.1217
ALL	72.32 %	0.701	0.209	0.701	0.1819

The comparative analysis of different features selection techniques with the full features set and selected features on search method based interpreted that in case of spam based dataset there in table attribute 34= dst_host_same_srv_rate equally selected in best first search method as per its high relevance with class by features selection method CFS-Subset-Eval but the same features doesn't selected by greedy

TABLE 18. Decision table results using filter supervised attribute selection (B).

Features Performance	Decision Table Classifier: Filter Supervised attribute selection				
	Accuracy	TPR	FPR	F-Measures	RMSE
15	86.65 %	0.897	0.143	0.035	0.1113
ALL	86.02 %	0.896	0.139	0.004	0.1161

TABLE 19. PART results using CFS with best first (B).

Features Performance	PART Classifier: CFS-Subset-Eval + Best First				
	Accuracy	TPR	FPR	F-Measures	RMSE
10	87.39 %	0.861	0.139	0.861	0.0920
ALL	85.21 %	0.851	0.149	0.851	0.1009

TABLE 20. PART results using CFS with greedy stepwise (B).

Features Performance	PART Classifier: CFS-Subset-Eval + Greedy Stepwise				
	Accuracy	TPR	FPR	F-Measures	RMSE
11	83.75 %	0.832	0.168	0.832	0.2441
ALL	80.90 %	0.801	0.199	0.801	0.2721

TABLE 21. PART results using IG with ranker (B).

Features Performance	PART Classifier: Information Gain + Ranker				
	Accuracy	TPR	FPR	F-Measures	RMSE
20	81.13 %	0.937	0.063	0.937	0.2325
ALL	79.62 %	0.929	0.071	0.929	0.2579

stepwise and ranker method as they have different selection criterion. But the features with so high relevance for one class remained unselected in very next approach. Same happens with many other features which refer to selection scale criterion but not selected due to some strict selection metrics and specification rules for selection. This is due to unavailability of some important information which was intestinally available in full training examples within the selected features. In case of NSL-KDD dataset selection of specific features increased the classifier accuracy in most cases which narrated that full training example with large information also include some imbalance data values which decreased performance but after class imbalance problem solved the overall training examples contains more information about problem than

TABLE 22. PART results using filter supervised attribute selection (B).

Features Performance	PART Classifier: Filter supervised attribute selection				
	Accuracy	TPR	FPR	F-Measures	RMSE
15	84.51 %	0.845	0.155	0.845	0.1855
ALL	72.22 %	0.722	0.278	0.722	0.2584

specified features so the risk for information loss in terms of features selection regarding different techniques remained undiscussed in most of research studies. The proper selection of all such features as separate metric will yield low but meaningful information can be extracted from such data values. The experimental setup shifted intentions towards minor information loss about features in most of the selection algorithms. There are certain issues that may encounter while working with these machine learning algorithms such as: (A) A major change in the structure of decision is observed in response to even a small change in the data which ultimately causes instability in the results. Similarly, (B) the decision tree algorithms take higher time to process and train the model. As a result, (C) numeric features extractions involve extra difficulty because of discretize of data or identification of threshold for rules is needed.

V. CONCLUSION

Features selection techniques with different search method when applied to a specified problem and evaluated through different classifiers for future implementations doesn't provide authenticity about the selection of sensitive information that in terms of features left unselected as these features are not noise nor in imbalance state but still not selected for training the classifiers and when such requirements for relevant features and information needed practical applications of by this training methodology doesn't provide sufficient information about under study problem. The accuracy of models is increasing day by day by rich algorithms but sensitivity and specificity in most of the cases not considered properly. The selection criterion supports our study that it is not necessarily to be the same features should be opted for every problem and every process. It involves variation both in terms of selection as well as performance. It causes a big flaw when working with features selection in terms of model generation. It states that some ignored features due to their low relevance or some other parameters at some future stage will be required to find out the solution of some sensitive scenario. What will happen when that particular information is need that missed on the basis of its low relevance or unmatched selection metrics and showed only high accuracy of performance, that high accuracy is not the time need but also particular information is the time need. The prior studies in this concern also proved that the same features by some features selection techniques can't be used for each classifier to build up a model where every classifier works on its defined rules for selected features so what about those features that selected by some selection technique are not valuable in each case either these features are improper for other conditions or some features with required information didn't selected. These minor information carrying features should be selected to interact with every available information within the datasets so that a model generated on such valuable information dependencies fulfills every dimensional view about data interpretation. So crystal impact of results obtained is that "availability of maximum number of features supports a model practical implementation in each aspect with maximum information catered but with selected features some minor information for individual feature set (but major as combined of all features) can cause model sensitivity problems and risks involvement for practical implementation in future". So the accuracy obtained by fewer selected features is only about selected features and no doubt that it yields high but what about that information which only obtained by selecting maximum number of features. In the same context our future work will be carried out to support the methods to improve the features grading according to information required to model a problem with every minor information provision that will support the data sensitivity for model creation especially for medical and security problems where minor loss of information will cause a major loss.

REFERENCES

[1] H. Shafizadeh-Moghadam, "Fully component selection: An efficient combination of feature selection and principal component analysis to increase model performance," *Expert Syst. Appl.*, vol. 186, Dec. 2021, Art. no. 115678.

[2] M. Rostami, K. Berahmand, E. Nasiri, and S. Forouzandeh, "Review of swarm intelligence-based feature selection methods," *Eng. Appl. Artif. Intell.*, vol. 100, Apr. 2021, Art. no. 104210.

[3] T. R. N and R. Gupta, "An efficient feature subset selection approach for machine learning," *Multimedia Tools Appl.*, vol. 80, no. 8, pp. 12737–12830, Mar. 2021.

[4] D. Xu, J. Zhang, H. Xu, Y. Zhang, W. Chen, R. Gao, and M. Dehmer, "Multi-scale supervised clustering-based feature selection for tumor classification and identification of biomarkers and targets on genomic data," *BMC Genomics*, vol. 21, no. 1, pp. 1–17, Dec. 2020.

[5] A. Kaur, K. Guleria, and N. K. Trivedi, "Feature selection in machine learning: Methods and comparison," in *Proc. Int. Conf. Advance Comput. Innov. Technol. Eng. (ICACITE)*, Mar. 2021, pp. 789–795.

[6] J. Brehmer, K. Cranmer, I. Espejo, F. Kling, G. Louppe, and J. Pavez, "Effective LHC measurements with matrix elements and machine learning," *J. Phys., Conf.*, vol. 1525, no. 1, Apr. 2020, Art. no. 012022.

[7] G.-S. Shokouh, B. Magnier, B. Xu, and P. Montesinos, "Ridge detection by image filtering techniques: A review and an objective analysis," *Pattern Recognit. Image Anal.*, vol. 31, no. 3, pp. 551–570, Jul. 2021.

[8] T. Das, S. Paitnaik, and S. P. Mishra, "Feature selection on linked data: A review," in *Applications of Artificial Intelligence in Engineering*. Singapore: Springer, 2021, pp. 615–624.

[9] T. Patel, "Exploring avenues for future research: Coastline feature extraction," Swansea Univ., Swansea, U.K., Tech. Rep. 912100, 2020.

[10] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," 2016, *arXiv:1601.07996*.

[11] O. Aziz, M. Musngi, E. J. Park, G. Mori, and S. N. Robinovitch, "A comparison of accuracy of fall detection algorithms (threshold-based vs. machine learning) using waist-mounted tri-axial accelerometer signals from a comprehensive set of falls and non-fall trials," *Med. Biol. Eng. Comput.*, vol. 55, no. 1, pp. 45–55, Jan. 2017, doi: 10.1007/s11517-016-1504-y.

[12] (2010). A. Frank and A. Asuncion. *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[13] (Feb. 2008). MIT Lincoln Labs. *DARPA Intrusion Detection Evaluation*. [Online]. Available: http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/index.html

[14] S. Al-Janabi, "Pragmatic miner to risk analysis for intrusion detection (PMRA-ID)," in *Soft Computing in Data Science* (Communications in Computer and Information Science), vol. 788, A. Mohamed, M. Berry, and B. Yap, Eds. Singapore: Springer, 2017, doi: 10.1007/978-981-10-7242-0_23.

[15] J. Bruggeman, J. Heringa, and B. W. Brandt, "PhyloPars: Estimation of missing parameter values using phylogeny," *Nucleic Acids Res.*, vol. 37, no. 2, pp. W179–W184, Jul. 2009.

[16] T. McCandless, S. E. Haupt, and G. Young, "Replacing missing data for ensemble systems," *J. Comput.*, vol. 6, no. 2, pp. 162–171, 2011.

[17] X.-K. Zhan, Z.-H. You, L.-P. Li, Y. Li, Z. Wang, and J. Pan, "Using random forest model combined with Gabor feature to predict protein-protein interaction from protein sequence," *Evol. Bioinf.*, vol. 16, Jan. 2020, Art. no. 117693432093449.

[18] M. Srinivas, A. H. Sung, and A. Ajith, "Modeling intrusion detection sustem using linear genetic programming approach," in *Proc. 17th Int. Conf. Ind. Eng. Appl. AI Expert Syst.*, O. Robert and Y. C. Ali Moonis, Eds. Berlin, Germany: Springer-Verlag, 2004.

[19] H. Malik, A. Iqbal, and A. K. Yadav, Eds., *Soft Computing in Condition Monitoring and Diagnostics of Electrical and Mechanical Systems: Novel Methods for Condition Monitoring and Diagnostics*. Springer, 2020.

[20] S. Chebrolu, A. Abraham, and J. P. Thomas, "Feature deduction and ensemble design of intrusion detection systems," *Comput. Secur.*, vol. 24, no. 4, pp. 295–307, Jun. 2005.

[21] A. Hofmann, T. Horeis, and B. Sick, "Feature selection for intrusion detection: An evolutionary wrapper approach," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Jul. 2004, pp. 25–29.

[22] A. Hofmann and B. Sick, "Evolutionary optimization of radial basis function networks for intrusion detection," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2003, pp. 20–24.

[23] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, Aug. 2016.

[24] W. Altidor, T. M. Khoshgoftaar, J. V. Hulse, and A. Napolitano, "Ensemble feature ranking methods for data intensive computing applications," in *Handbook of Data Intensive Computing*. New York, NY, USA: Springer, 2011, pp. 349–376.

[25] K. Revathi and T. KalaiSelvi, "Survey: Effective feature subset selection methods and algorithms for high dimensional data," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 6, no. 4, pp. 1880–1888, 2013.

[26] G. Ozkaya, M. Timor, and C. Erdin, "Science, technology and innovation policy indicators and comparisons of countries through a hybrid model of data mining and MCDM methods," *Sustainability*, vol. 13, no. 2, p. 694, Jan. 2021.

[27] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, "Practical machine learning tools and techniques," *Data Mining*, vol. 2, p. 4, 2005.

[28] G. Forman. *(gforman at nospam hpl.hp.com), Spambase Data Set, 650-857-7835*. Accessed: Nov. 25, 2021. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Spambase

[29] (Mar. 2014). *NSL-KDD Data Set for Network-Based Intrusion Detection Systems*. [Online]. Available: http://nsl.cs.unb.ca/KDD/NSL-KDD.html

[30] *Weka Data Mining Machine Learning Software*. Accessed: Aug. 5, 2021. [Online]. Available: http://www.cs.waikato.ac.nz/ml/weka

[31] E. Alyahyan and D. Düştegör, "Predicting academic success in higher education: Literature review and best practices," *Int. J. Educ. Technol. Higher Educ.*, vol. 17, no. 1, pp. 1–21, Dec. 2020.

[32] A. N. Alharbi, "An improvement in branch and bound algorithm for feature selection," *Int. J. Inf. Technol. Lang. Stud.*, vol. 4, no. 1, p. 1–11, 2020.

[33] R. B. Zajonc, "A note on group judgements and group size," *Hum. Relations*, vol. 15, no. 2, pp. 177–180, May 1962.

[34] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed, "A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction," *J. Appl. Sci. Technol. Trends*, vol. 1, no. 2, pp. 56–70, May 2020.

[35] X. Chen, J. Li, Y. Zhang, Y. Lu, and S. Liu, "Automatic feature extraction in X-ray image based on deep learning approach for determination of bone age," *Future Gener. Comput. Syst.*, vol. 110, pp. 795–801, Sep. 2020.

[36] W. Lin, J. Huang, C. Y. Suen, and L. Yang, "A feature extraction model based on discriminative graph signals," *Expert Syst. Appl.*, vol. 139, Jan. 2020, Art. no. 112861.
[37] Z. Liu, Z. Lai, W. Ou, K. Zhang, and R. Zheng, "Structured optimal graph based sparse feature extraction for semi-supervised learning," *Signal Process.*, vol. 170, May 2020, Art. no. 107456.
[38] E. S. Hosseini and M. H. Moattar, "Evolutionary feature subsets selection based on interaction information for high dimensional imbalanced data classification," *Appl. Soft Comput.*, vol. 82, Sep. 2019, Art. no. 105581.
[39] X. Han, P. Liu, L. Wang, and D. Li, "Unsupervised feature selection via graph matrix learning and the low-dimensional space learning for classification," *Eng. Appl. Artif. Intell.*, vol. 87, Jan. 2020, Art. no. 103283.
[40] S. Ayesha, M. K. Hanif, and R. Talib, "Overview and comparative study of dimensionality reduction techniques for high dimensional data," *Inf. Fusion*, vol. 59, pp. 44–58, Jul. 2020.
[41] T. Chen, X. Shi, and Y. D. Wong, "Key feature selection and risk prediction for lane-changing behaviors based on vehicles' trajectory data," *Accident Anal. Prevention*, vol. 129, pp. 156–169, Aug. 2019.
[42] J. R. Méndez, T. R. Cotos-Yañez, and D. Ruano-Ordás, "A new semantic-based feature selection method for spam filtering," *Appl. Soft Comput.*, vol. 76, pp. 89–104, Mar. 2019.
[43] X. Shi, Y. D. Wong, M. Z.-F. Li, C. Palanisamy, and C. Chai, "A feature learning approach based on XGBoost for driving assessment and risk prediction," *Accident Anal. Prevention*, vol. 129, pp. 170–179, Aug. 2019.
[44] S. Sharmin, M. Shoyaib, A. A. Ali, M. A. H. Khan, and O. Chae, "Simultaneous feature selection and discretization based on mutual information," *Pattern Recognit.*, vol. 91, pp. 162–174, Jul. 2019.
[45] L. Kotthoff, "Auto-WEKA: Automatic model selection and hyperparameter optimization in WEKA," *Automated Machine Learning*. Cham, Switzerland: Springer, 2019, pp. 81–95.

AMJAD ALI received the Master of Science Information Technology (M.S.I.T.) degree from Bahauddin Zakariya University, Multan, Punjab, Pakistan, in 2020. He is currently working as a Network Engineer at Bahauddin Zakariya University. His research interests include intrusion detection/prevention systems, network security, the Internet of Things, software-defined networks, machine learning, and deep learning. He has several international certifications including IEEE, CCNA, and Microsoft.

FAIZ JILLANI received the Master of Sciences Information Technology (M.S.I.T.) degree from Bahauddin Zakariya University, Multan, Punjab, Pakistan, in 2020. He is currently working as a Lecturer with the Department of Information Technology, Bahauddin Zakariya University. His research interests include machine learning, artificial intelligence, and big data analytics.

RABBIAH ZAHEER received the Master of Physics degree from Bahauddin Zakariya University, Multan, Punjab, Pakistan, in 2018, where she is currently pursuing the M.Phil. degree in physics. She is currently working as a Lecturer with Private University. Her research interests include bismuth borate glasses, computation physics, and quantum cryptography.

AHMAD KARIM received the M.S. (CS) and M.I.T. degrees (Hons.) from Bahauddin Zakariya University, Multan, Pakistan, and the Ph.D. degree in network security from the University of Malaya, Malaysia. He did his postdoctorate training at Telekom Malaysia (TM). He is currently serving as a Lecturer with the Department of Information Technology, Bahauddin Zakariya University. His research interests include mobile botnet detection, computer security, computer networks, software defined networks, big-data analytics, and the IoT. He holds Cisco and Redhat International Certifications (CCNA, CCNP, CCAI, Openshift, and Kubernetes).

YASSER OBAID ALHARBI received the Ph.D. degree in computer science from the University of Essex, Colchester, U.K., in February 2018. He is currently serving as an Assistant Professor at the College of Computer Science and Engineering, University of Hail, Ha'il, Saudi Arabia. His research interests include grid and cloud computing systems and networks, resource, and application management in clouds (VMs allocation and migration), performance analysis, modeling, and optimization, energy efficiency for cloud data centers (green cloud), and big data in the network and utility computing.

MOHAMMAD ALSAFFAR received the B.Sc. degree in computer science from the University of Hail, Saudi Arabia, in 2008, the M.Sc. degree in computing from De Montfort University, U.K., in 2011, and the Ph.D. degree in computer science from the University of Brighton, U.K., in 2019. He is currently an Assistant Professor with the Computer Science and Engineering College, University of Hail. He is also the Vice Dean of community service and continuing education deanship. His research interests include user experience, human–computer interaction, usability, interface design, and cyber security.

KHALID ALHAMAZANI received the M.S. degree in information technology from Monash University, Melbourne, Australia, in 2010, and the Ph.D. degree in computer science and engineering from the University of New South Wales, Sydney, Australia, in 2016. He is currently an Assistant Professor at the College of Computer Science and Engineering, University of Hail, Saudi Arabia. His research interests include blockchain technology and cloud computing, as well, he is engaged in several digital transformation initiatives and projects in Saudi Arabia.