A Clustering based Feature Selection Approach using Maximum Spanning Tree

Md. Hasan Tarek, Suravi Akhter, Sumon Ahmed, Md Shariful Islam*, Mohammad Shoyaib and Zerina Begum

Institute of Information Technology Dhaka, Bangladesh

**E-mail: shariful@iit.du.ac.bd*

Received on 08 August 2022, Accepted for publication on 05 January 2023

ABSTRACT

Mutual information (MI) based feature selection methods are getting popular as its ability to capture the nonlinear and linear relationship among random variables and thus it performs better in different fields of machine learning. Traditional MI based feature selection algorithms use different techniques to find out the joint performance of features and select the relevant features among them. However, to do this, in many cases, they might incorporate redundant features. To solve these issues, we propose a feature selection method, namely Clustering based Feature Selection (CbFS), to cluster the features in such a way so that redundant and complementary features are grouped in the same cluster. Then, a subset of representative features is selected from each cluster. Experimental results of CbFS and four state-of-the-art methods are reported to measure the excellency of CbFS over twenty benchmark UCI datasets and three renowned network intrusion datasets. It shows that CbFS performs better than the comparative methods in terms of accuracy and performs better in identifying attack or normal instances in security datasets.

Keywords: Clustering, Maximum Spanning Tree, Feature Selection, Mutual Information

1. Introduction

In this era of fourth industrial revolution (4IR), there has been an enormous growth in the uses of digital services in our daily life. This not only produces a huge amount of daily internet traffic but also increases the likelihood of numerous cyberattacks. Previous studies show that there are more than three billion cyber-attacks in a single day in the USA and Australia [1]. From these large amounts of Internet traffic data, we have to identify the important and discriminative features that can identify/classify the appropriate attack/incident in a reduced cost. There are numerous sorts of feature selection/extraction techniques available to identify those features. Feature selection is a process of selecting relevant, important features F_S and removing irrelevant features from a set of feature F. Relevant features selection and eliminating irrelevant features will reduce time and accelerate the classification performance [2], [3], [34], [35].

Various types of feature selection method have been proposed over time. Song et al. [4] divided feature selection methods into four broad categories, namely Filter, Wrapper, Embedded and Hybrid method. Wrapper method tries to find a subset of features, train those features using an algorithm and based on the predictive accuracy add or eliminate features from the subset. Generality of selecting a feature subset of wrapper method is low and it's cost of computation is very high. This method can be classifier dependent [5], [6]. Embedded method takes feature selection as part of their training process, it is not a generalization method as it is specific to a classifier and thus it is classifier dependent [6]. In Filter method, features are selected based on the scores with the target but it does not give the surety of the result. It is scalable to larger dimensional dataset and independent of classifiers [6], [7]. A Combination of Filter and Wrapper method is the Hybrid method [8].

Methods with classifier independence rank the features with respect to their relevance measures to the class label. These measures can be computed using different metrics e.g. distance, consistency, dependency, correlation and mutual information (MI) [6], [9]. Among these measures, MI is more popular than others because of its ability to capture the non-linear and linear relation between features in the dataset and it can be used with categorical as well as numerical values [6], [9], [10], [34], [35]. Several MI based methods have been proposed over time. Mutual information maximization (MIM) [11] is one of the earliest methods that tries to maximize the relevancy of features ignoring the feature redundancy information. Mutual Information based Feature Selection (MIFS) [12] incorporates this information in their proposed work. Joint Mutual Information (JMI) [10] considers both the relevancy, redundancy along with additional information about the class label (complementary) to their feature selection criteria.

However, due to the finite number of samples bias may exist in the dataset. Joint Bias corrected Mutual Information (JBMI) is proposed by Sharmin et al. [9] where they have calculated the amount of bias for relevancy, redundancy and complementary information. However, higher order feature interaction term is not considered which has been addressed in RelaxMRMR [13]. Recently, Roy et al. [14] proposed Discretization and feature Selection based on bias corrected Mutual information (DSbM) where they have showed the bias term for this higher order feature interaction information.

Besides, in feature selection method, search strategy plays an important part in the feature selection process. Different search strategy for feature selection exists in literature such as exhaustive search, Forward selection (FS), Backward elimination (BE), Genetic algorithm [15] and Convex based Relaxation Approximation (COBRA) [16]. An exhaustive

search strategy would be able to find optimal features but this needs to compute all possible pairs of features which is a NP-hard problem [16], [17]. In FS and BE, a feature is selected or removed one at a time if it satisfies the selection criteria. However, the problem is that the removed features cannot be re-selected and vice versa. Thus, it may select the redundant features [18]. Though genetic algorithm can select optimal features, it's computation cost is high and not practical in many high dimensional dataset [18]. Convex based Relaxation Approximation (COBRA) [16] is another parallel searching method that selects feature using MI with the help of semi-definite programming.

Graph centric idea is employed in feature selection. Moradi and Rostami [3] presented a graph based unsupervised feature selection approach. Song et al. [4] described a method namely fast clustering-based feature selection algorithm (FAST) to select relevant features using minimum spanning tree where a representative feature is selected from each cluster. In this approach, only the representative feature from each cluster is selected without considering more than one feature from each cluster which may provide complementary information about the class that helps to achieve better classification performance. Apart from previously mentioned method, Nikama *et al.* proposed a feature elimination process based on ANOVA feature scoring and follows an exhaustive search for feature subset selection [24]. However, exhaustive search might be impossible when the dimensionality increases in a dataset.

Addressing the aforementioned issues in this work, we have proposed a new feature selection method namely, Clustering based Feature Selection (CbFS) that selects a prominent feature subset without exhaustive searching. The key contributions of this paper is summarized as follows: i) CbFS incorporates both redundancy and complementary information for creating a cluster of more relevant features. ii) JBMI is applied to select the best representative feature from each cluster to find the feature subset. iii) performance analysis of CbFS and other existing methods in twenty benchmark UCI datasets and three renowned network intrusion detection datasets are represented.

The rest of the paper is sorted out as the following, Section II discusses existing MI based methods, Section III describes our proposed feature selection method. Next, result analysis and discussion are presented in section IV and we summarize our work in section V.

2. Related work

Feature selection is a pre-processing step in machine learning that selects relevant subset of features while reducing the dimension of features as well as finding irrelevant and redundant feature subset [2], [19]. Different feature selection method has been proposed over time.

One of the earliest works in feature selection is Relief [20] algorithm. It uses distance measure to estimate the relevance of features that differentiate between the instances of the same and another class close to each other. If more feature is relevant to the target class it selects almost all of the features while a small subset of feature set might be helpful. Relief was introduced to solve only two class problem and could not identify redundant features.

Relief-F [21] is introduced that can work with incomplete, noisy dataset, also with multi class problem. But still cannot find the redundant features. To solve it, different feature selection methods are introduced such as, Correlation based Feature Selection (CFS) [22], Fast Correlation based Filter Solution (FCBF) [19], joint mutual information (JMI) [10] that can identify relevant as well as redundant features. CFS is a filtering approach which ranks features based on a correlation based heuristic evaluation function. The key of this method is that a good set of features contain those features that are highly correlated with the target class and yet not correlated with each other. FCBF can also select relevant and detect redundant features from the feature set without computing pairwise correlation analysis.

Mutual Information (MI) based feature selection methods are getting popular because of its capability to capture the non-linear and linear relationship among variables. A work presented in [5], namely, joint mutual information (JMI) that incorporates relevancy (MI between a feature and a class), redundancy (MI between two features) and complementary (MI between two features given the class label) in their feature selection criteria given in Eq. (1)

$$J_{JMI}(x_i) = I(x_i; C) - \frac{1}{|S|}\sum_{x_j \in S}\left(I(x_i; x_j) - I(x_i; x_j | C)\right) \quad (1)$$

Here, x_i is the candidate feature to be selected, S is the already selected feature set and these three terms represent relevancy, redundancy and complementary respectively. However, as the dataset contains finite number of samples there exists a bias which is addressed in the work of [9]. They have corrected the bias of these terms along with their corresponding critical value. Thus, the Joint Bias corrected Mutual Information (JBMI) formula becomes as the following Eq. (2)

$$J(x_i) = I(x_i; C) - \frac{(I-1)(K-1)}{2N\ln 2} - \frac{1}{|S|}\sum_{x_j \in S}\Big(I(x_i; x_j) - \frac{(I-1)(J-1)}{2N\ln 2} - I(x_i; x_j | C) + \frac{(I-1)(J-1)K}{2N\ln 2}\Big) \quad (2)$$

Here I, J are the intervals of features x_i *and* x_j and K *and* N are the number of classes and total number of samples respectively.

Higher order interaction of features along with relevancy, redundancy and complementary is considered in RelaxMRMR [13]. They showed that selected feature set is conditionally independent of the given feature x_i and any feature x_j in S under their relaxed assumption. Roy et al. [14] proposed discretization and feature selection based on bias corrected mutual information (DSbM) and extending it by simultaneous forward selection and backward elimination ($DSbM_{fb}$) that calculates the bias term of the higher order interaction of RelaxMRMR. This bias corrected feature selection can be expressed as of Eq. (3)

$$\begin{aligned} J(x_i) = I(x_i;C) - \frac{(I-1)(K-1)}{2N\ln 2} - \frac{1}{|S|}\sum_{x_j \in S}\Big(I(x_i;x_j) \\ - \frac{(I-1)(J-1)}{2N\ln 2} - I(x_i;x_j|C) \\ + \frac{(I-1)(J-1)K}{2N\ln 2}\Big) \\ - \frac{1}{|S||S-1|}\sum_{x_j \in S}\sum_{x_k \in S, j\neq k}\Big(I(x_i;x_k|x_j) \\ - \frac{(I-1)(L-1)J}{2N\ln 2}\Big) \end{aligned} \quad (3)$$

Here L is the number of intervals in x_k and $I(x_i;x_k|x_j)$ is the higher order interaction term. Apart from these Naghibi et al. [16] presented a search strategy for feature selection using convex based relaxation approximation (COBRA). It uses semi-definite programming to search the feature space and select the features. Recently, Gao et al. [23] proposed an MI based feature selection method namely min-redundancy and max-dependency (MRMD) which states that the larger value of redundancy term does not indicate how worse a candidate feature because at the same time that feature can give new classification information. Their new feature selection criteria is as follows in Eq. (4)

$$J_{\text{MRMD}}(x_i) = I(x_i;C) - \frac{1}{|S|}\sum_{x_i \in S}(I(x_i;x_j) - I(x_i;C|x_j)) \quad (4)$$

This criterion considers the relevancy between candidate feature x_i and class label C given the already selected feature x_j.

Nkiama et al. [24] presented a feature selection method to remove the irrelevant and select the relevant features for security dataset. To select these features, they have used ANOVA F-test to get the score to identify the strength of a feature related to the class label. Next a subset of features is selected using SelectPercentile method and performed a recursive feature elimination process to select the final feature set to identify different attack on the security dataset.

Song et al. [4] presented a FAST-clustering algorithm in feature selection. It first groups the features into clusters with the help of graph-theoretic based knowledge using minimum spanning tree. Then a representative feature from each cluster is selected that is more related to the target class to get the final set of selected features.

3. Proposed method

In this section, we have discussed our proposed CbFS method for feature selection. It clusters the redundant and complementary features in the same group and selects features from each cluster to get the final feature set F_S. Fig. 1 presents the overall workflow of CbFS.

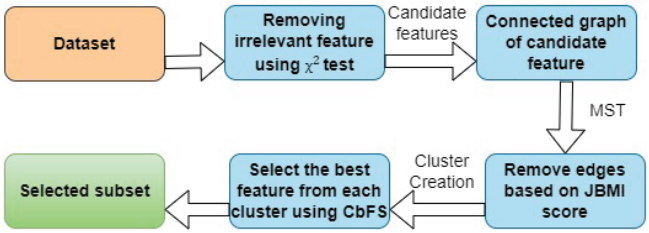

Fig. 1: Overview of feature selection method

MI can be normalized with various ways suggested in [25]. We have used Symmetric Uncertainty (SU) measure to normalize the MI value to bring them in a scale of (0,1). SU_{red} between two features x_i and x_j can be defined using Eq. (5)

$$SU_{red}(x_i;x_j) = \frac{2 \times I(x_i;x_j)}{H(x_i) + H(x_j)} \quad (5)$$

Here, I represent the MI and $H(x_i)$ represents the entropy of feature x_i. In the similar fashion, SU_{comp} between two features x_i and x_j given the class label C can be formulated as defined in Eq.(6)

$$SU_{comp}(x_i;x_j|C) = \frac{2 \times I(x_i;x_j|C)}{H(x_i) + H(x_j)} \quad (6)$$

3.1. Irrelevant Feature Removal

Irrelevant features may degrade the performance of classification. Thus, it is necessary to remove these features. From the feature set F, after removing these features we get candidate feature set F_C. These irrelevant features are removed using the bias corrected MI value between a feature and a class label (Relevancy) and its corresponding critical value is shown in [8]. Bias corrected relevancy value can be calculated using Eq.(7)

$$I'^{(x_i;C)} = I(x_i;C) - \frac{(I-1)(K-1)}{2N\ln 2} \quad (7)$$

Here I is the number of intervals of x_i, K is the number of classes and N is the total number of samples. The corresponding critical value of Eq. (7) can be calculated by using Eq. (8)

$$\chi^2_C = I(x_i;C) \times 2N\ln 2 \quad (8)$$

Irrelevant features are removed from F if the relevancy value of a feature is less than its corresponding χ^2 critical value. After removing these we get candidate feature set F_C from which clusters are created as described in the following section. The process of irrelevant feature removal is described in Algorithm. 1.

3.2. Cluster Creation

We want to create the clusters in such a way that the redundant and complementary features are grouped in the same cluster. To do this, we construct a fully connected graph with candidate feature set F_C, then construct Maximum Spanning Tree (MST) using Prim's algorithm.

Algorithm 1: Removal of irrelevant features

Input: Feature set, F
Output: Candidate feature set, F_C

$F_C \leftarrow \emptyset$
for $x_i \in F$ do
Calculate $J_{rel}(x_i)$ for x_i with respect to C using Eq. (7)
Calculate $\chi^2_C(rel)$ using Eq. (8)
if $J_{rel}(x_i) > \chi^2_C(rel)$ then
$F_C \leftarrow F_C \cup x_i$
end
end

MST is used here to get that one path with highest redundancy value that connects all features. If any edge in the MST is removed, we get new trees (clusters). This edge will be removed if both the redundancy and complementary information of two features is less than their relevancy value. Thus, the more redundant and complementary features are grouped in the same cluster. This cluster creation process is described using the following example.

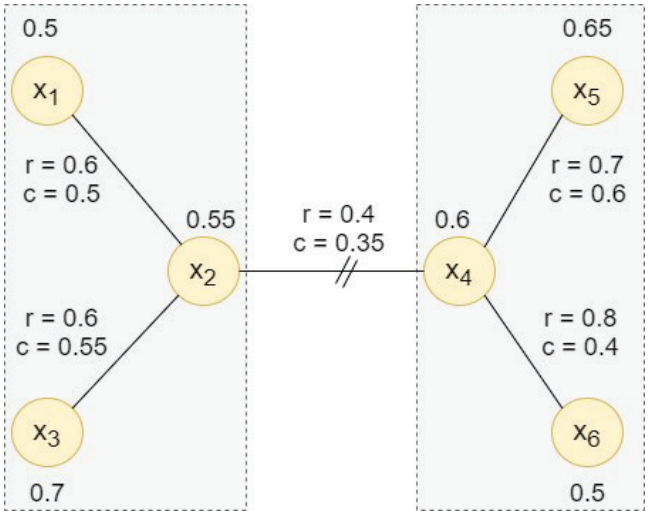

Fig. 2: Cluster creation example

Example: Let's consider from the original feature set *F*, after removing irrelevant features we get six candidate features $F_C = \{x_1,x_2,x_3,x_4,x_5,x_6\}$. Using these features, we construct a fully connected graph and from this graph MST is created using Prim's algorithm presented in Fig. 2. In the figure, the value on each node represent the relevancy $SU_{rel}(x_i,C)$ value and the value on each edge *r,c* represents the redundancy $SU_{red}(x_i,x_j)$ and complementary $SU_{comp}(x_i,x_j|C)$, respectively. We traverse on each edge in Fig. 2 and check whether both of it's SU_{red} and SU_{comp} value are less than the SU_{rel} value. If this condition is met, that connecting edge is eliminated and new trees (cluster) are formed. From the figure, we notice that both the redundancy and complementary value between x_2 and x_4 is 0.4 and 0.35 which are less than their corresponding relevancy value 0.55 and 0.6, respectively. Thus, removing this edge results in two clusters $\{x_1,x_2,x_3\}$ and $\{x_4,x_5,x_6\}$. This whole process is presented in Algorithm 2.

Algorithm 2: Cluster creation

Input: Candidate Feature set, F_C
Output: Clusters , Forest

G←NULL; $F_S \leftarrow F_S \cup x_i$
for Each pair $(x_i,x_j) \in F_S$ **do**
$SURed_{ij}$ ←Calculate $SU(x_i;x_j)$ using Eq. (5)
Add x_i and x_j with $SURed_{ij}$ as the edge value in G
end
maxSpanTree ← PrimsAlgorithm(G)
Forest ← maxSpanTree
for Each edge $e_{ij} \in$ Forest **do**
$SUcomp_{ij}$ ←Calculate $SU(x_i;x_j|C)$ using Eq. (6)
if $SURed_{ij} < SURel_i$ & $SURed_{ij} < SURel_j$ & $SUcomp_{ij} < SURel_i$ &
$SUcomp_{ij} < SURel_j$ then
Forest ← Forest − e_{ij}
end
end

3.3. Final Feature Selection

In this section, final features F_S are selected from the clusters obtained from the previous section. To get this, JBMI is applied on each cluster. In this process, highest relevancy valued feature is selected from each cluster. Then find out if the next highest relevancy valued feature can give us complementary information about the class label. If a feature in that cluster meet the condition, it is also included in the selected feature set described in Algorithm 3. Finally, combining these features from each cluster give us the final feature subset F_S.

Algorithm 3: Feature subset selection

Input: Clusters , Forest
Output: Selected Subset, F_S

$F_S \leftarrow \emptyset$
for Each tree $T_i \in$ Forest **do**
Sort features in T_i in decreasing order based on corresponding relevance value
$F_S' \leftarrow x1$;Ti ← Ti \ x1
for $x_j \in T_i$ **do**
Calculate JBMI score ($J_{JBMI}(x_j)$) using Eq. (2) and corresponding χ^2 critical value

```
7        if J_JBMI(x_j) > χ² then
8            F_S' ← F_S' ∪ xj
9        End
10   end
11   F_S ← F_S ∪ F_S'
12   end
13   return F_S
```

4. Experimental result

This section presents the dataset description and result comparisons of CbFS with other four state-of-the-art methods.

Table 1: Dataset Description

Dataset	Feature	Instances	Class
Iris	4	150	3
Appendicitis	7	106	2
Ecoli	7	336	8
Pima	8	768	2
Saheart	9	462	2
Shuttle	9	57999	7
Heart	13	270	2
Wine	13	178	3
Cleveland	13	297	5
Australian	14	690	2
Vehicle	18	846	2
Ring	20	7400	2
Thyroid	21	7200	3
Parkinsons	22	195	2
Steel	27	1941	7
Ionosphere	34	351	2
Spectfheart	44	267	2
Optdigits	64	5620	10
Coil2000	85	9822	2
Madelon	500	2600	2
Security Dataset			
NSL-KDD	42	148517	5
AWID	78	575315	2
CIC-IDS2017	78	2827876	15

4.1. Dataset Description

To evaluate the performance of our method, 20 datasets are used from UCI machine learning repository [26], Arizona State University [27] and Knowledge Extraction based on Evolutionary Learning (KEEL) [28] repository. Also, to examine the performance of the security dataset, three datasets namely NSL-KDD [29], AWID [30] and CIC-IDS2017 [31] are selected. These datasets characteristics are shown in Table-1.

4.2. Result and Discussion

We have compared CbFS result with three other state-of-the-art feature selection methods namely FAST, DSbM and JMI with COBRA (JC) and one ranking method namely MRMD on twenty benchmark datasets. To get the result with ranking method we have used the number of selected features by CbFS method for each dataset. To produce these results we have performed five equal width discretization on those dataset. Also K-fold (K=10) cross validation is used with linear Support Vector Machine (SVM) and decision tree (DT) to calculate the accuracy and F-score of each dataset.

The result of CbFS and other methods are given in Table. 2. It is noticeable by inspecting the table that CbFS performs better than other comparative methods. In the table, the number of selected features corresponding to that method are shown in the parenthesis. For example, in *Cleveland* dataset CbFS achieves 60.64% accuracy with four features whereas JC selects all features and still gets 53.44% which is less than ours. Moreover, FAST and DSbM each selects four features but their results are still worse than ours. The results indicate that CbFS groups the redundant and complementary features in the same cluster and select the relevant and important ones from those clusters that helps to achieve better performance. We have also showed the number of win, tie or loss of CbFS with other comparative methods. It shows that in most of the cases our proposed method wins over other methods. Moreover, t-test (at 95% confidence interval) is performed to show the significant number of wins or loses presented in Table. 2. The result demonstrates that CbFS significantly wins compare to other methods.

To clearly understand the superiority of our methods Friedman rank test [32] is also performed. To compare which method's performance is significant, it uses Nemenyi test [33] after rejecting the null hypothesis. This result also shows that CbFS gets the first rank among other methods presented in the second last row of this table. The last row of the table also shows that Friedman rank test of CbFS significantly outperforms other state-of-the-art methods marked with ✓ (at 5% level of significance) except MRMD method. We have also calculated the F-score result presented in Table 3. F-score helps to give better insight of the result when the dataset is imbalanced. From the F-score result, we observe that it also gives similar type of result as accuracy. The overall result indicates the superiority of CbFS method in terms of both accuracy and F-score compared to other methods.

Network-traffic dataset result: Apart from the twenty datasets we also compare CbFS result with three other renowned network-traffic datasets. For these datasets, in additional to the existing methods, another one proposed by Nkiama et al. [24] is also taken into consideration which focuses in identifying security related features.

Table 4 presents the DT result comparison of CbFS and other state-of-the-art methods while applying in intrusion detection

datasets. The results in this table show that CbFS performs better than others in terms of accuracy. The experimental results also depict that CbFS correctly identifies relevant features that can accurately predict the attack classes, though the number of selected features is comparatively higher than others. Moreover, it can be seen from the result that with this number of features, the accuracy result of CbFS is better than other comparative methods. Further, we report the class-wise accuracy and confusion matrix of network traffic dataset in Table V and Table 4 respectively. From the confusion matrix result, we can observe different class identification ability of our proposed CbFS method.

Table 2: Accuracy (SVM and DT) comparison among different methods. (*) and (◦) represent significant win or loss corresponding to that method, Bold face results represent overall win.

	SVM					DT				
Dataset	**CbFS**	**FAST**	**JC**	**DSbM**	**MRMD**	**CbFS**	**FAST**	**JC**	**DSbM**	**MRMD**
Iris	94.00(2)	91.33(1)	91.33(2)	92.67(2)	**95.33**	**94.00**	91.33	92.67	91.33	92.00
Appendicitis	**87.64(1)**	**87.64(1)**	85.00(4)	80.00(2)	75.00*	**86.73**	**86.73**	80.83	82.50	83.33
Ecoli	**77.68(5)**	63.09(1)*	74.21(5)	66.32(4)*	72.63	**76.44**	64.27*	72.37	67.11*	74.74
Pima	**75.26(3)**	72.92(2)	74.29(8)	74.94(5)	73.38	**75.39**	74.74	71.30*	71.95	73.12
Saheart	69.69(2)	69.26(2)	71.70(9)	71.06(5)	**72.98**	68.84	67.32	65.11*	65.11	**71.49**
Shuttle	94.11(5)	84.23(2)*	**94.73(7)**	93.93(5)*	93.12*	94.22	84.26*	**94.74**	94.24	93.19*
Heart	**83.70(5)**	80.00(4)	81.11(10)	80.74(9)	81.85	81.11	**82.59**	72.22*	77.04	**82.59**
Wine	96.11(9)	95.49(4)	91.58(9)	**96.84(9)**	95.79	91.54	**93.30**	93.16	88.42	92.63
Cleveland	**60.64(4)**	60.63(4)	53.44(13)*	54.38(4)*	53.75*	**60.33**	56.59	48.75*	52.50*	50.00*
Australian	85.51(4)	85.51(4)	68.29(11)*	**87.71(9)**	87.57	86.23	84.06	65.00*	85.57	**87.86**
Vehicle	**88.29(6)**	77.77(1)*	84.35(13)*	76.82(3)*	81.65*	**90.54**	79.90*	89.76	79.18*	82.82*
Ring	69.64(20)	65.60(5)*	66.92(11)*	61.47(3)*	**70.04**	86.88	78.95*	86.37	70.50*	**90.08**◦
Thyroid	**93.21(5)**	**93.21(6)**	92.93(18)*	92.51(6)*	93.12	**93.21**	93.19	92.82*	92.61*	93.12
Parkinsons	**85.11(1)**	**85.11(6)**	84.5(14)	81.5(11)	76.50*	87.11	84.58	**90.00**	86.50	83.00
Steel	**70.94(19)**	56.41(7)*	69.65(20)	63.94(9)*	68.74	**70.69**	58.01*	68.74	69.60	69.55
Ionosphere	87.48(29)	87.77(8)	85.00(34)	65.00(4)*	86.11	90.30	90.03	**90.56**	76.39*	90.28
Spectfheart	79.42(1)	**79.43(8)**	74.29(31)*	78.93(27)	78.57	**79.42**	70.50*	74.64*	72.50*	78.57
Optdigits	97.26(48)	89.8(14)*	97.21(55)	96.33(52)*	**97.53**	**90.82**	86.44*	89.68	90.18*	89.79*
Coil2000	**94.03(12)**	**94.03(17)**	93.98(85)*	94.00(11)*	94.00*	90.22	90.68	91.16◦	**93.98**◦	93.95◦
Madelon	59.12(10)	57.69(60)	54.35(500)*	**61.31(16)**	57.96	**75.81**	68.81*	67.92*	55.77*	65.62*
W/T/L	-	13/5/2	18/0/2	16/0/4	15/0/5	-	16/1/3	15/0/5	18/0/2	14/0/6
Sig. W/L	-	6/0	8/0	10/0	6/0	-	8/0	8/1	9/1	5/2
Avg. Rank	1.78	3.45	3.48	3.38	2.93	1.83	3.43	3.28	3.75	2.73
F. Rank Test	-	✓	✓	✓	-	-	✓	✓	✓	-

5. Conclusion

In this work, we have proposed a feature selection method named as CbFS that firstly removes the irrelevant features to get the candidate features set. Then a fully connected graph is created from which MST is created using Prim's algorithm. Then from this MST, clusters are formed in such a way that the redundant and complementary features are grouped together. Finally, a subset of features is selected from each cluster using JBMI that help to achieve better classification performance. To evaluate the performance of our proposed method, rigorous experiments have been performed on twenty benchmark datasets and network traffic datasets; results are compared with four state-of-the-art methods namely FAST, JC, DSbM and MRMD. Apart from this method we have compared with another method proposed by Nkiama et al. to compare the network intrusion

detection dataset results. Experimental results on twenty benchmark UCI datasets show that in most of the cases CbFS significantly outperforms other comparative methods. Moreover, CbFS also performs well in identifying attack or normal data instances on security dataset.

Acknowledgement

This work is funded by the Centennial Research Grant (CRG) of University of Dhaka, Bangladesh.

Table 3: F-score (SVM and DT) comparison among different methods; Bold face results represent overall win.

	SVM					DT				
Dataset	**CbFS**	**FAST**	**JC**	**DSbm**	**MRMD**	**CbFS**	**FAST**	**JC**	**DSbM**	**MRMD**
Iris	93.97	91.27	91.88	93.6	**95.8**	**93.98**	91.27	93.24	92.34	92.7
Appendicitis	**86.09**	**86.09**	78.47	69.23	42.86	**85.31**	**85.31**	71.49	74.21	76.08
Ecoli	**76.24**	56.8	53.43	32.83	43.83	**75.32**	58.57	48.91	40.15	46.66
Pima	**74.16**	72.36	71.38	72.18	69.6	**74.54**	72.25	68.16	69.3	69.54
Saheart	**68.87**	63.19	67.13	66.83	68.23	66.45	**65.94**	61.39	60.68	65.98
Shuttle	**93.48**	77.06	60.35	48.54	44.19	**93.62**	77.17	62.54	55.44	65.85
Heart	**83.56**	79.89	80.96	80.66	81.7	80.91	82.41	72.29	76.88	**82.64**
Wine	96.1	95.47	92.46	**97.02**	96.18	91.37	93.25	**93.65**	89.21	93.39
Cleveland	56.12	**56.79**	30.34	28.73	29.36	**55.99**	52.99	27.67	26.31	27.17
Australian	85.52	85.52	67.69	**88.49**	88.32	86.18	83.94	64.54	85.58	**87.79**
Vehicle	**88.22**	71.81	77.11	46.27	71.86	**90.56**	79.28	85.89	70.04	75.31
ring	69.47	63.15	67.44	61.5	**70.18**	86.87	78.85	86.44	70.57	**90.1**
Thyroid	**90.31**	**90.31**	44.58	32.67	50.24	**90.31**	90.28	47.76	41.69	50.24
Parkinsons	**85.16**	**85.16**	77.18	73.08	58.39	**86.92**	83.52	85.9	81.9	76.91
Steel	**70.66**	53.26	70.41	61.97	61.67	70.56	57.63	69.43	**72.14**	71.15
Ionosphere	87.07	**87.34**	83.98	44.47	85.14	**90.29**	89.95	90.06	73.7	89.72
Spectfheart	70.31	**77.21**	59.2	64.21	44	70.31	**70.81**	61.8	58.89	44
Optdigits	97.25	89.79	97.25	96.37	**97.56**	**90.82**	86.44	89.79	90.22	89.92
Coil2000	**91.14**	**91.14**	48.45	48.45	48.45	89.98	**90.34**	55.62	58.13	57.03
Madelon	59.1	57.65	54.36	**61.32**	57.98	**75.77**	68.8	67.87	55.81	65.57
Win/tie/loss	-	**12/5/3**	**19/1/0**	**17/0/3**	**15/0/5**	**-**	**15/1/4**	**19/0/1**	**19/0/1**	**15/0/5**

Table 4: Decision tree accuracy comparison among different methods for network-traffic datasets

Dataset	**CbFS**	**FAST**	**JC**	**DSbM**	**Nkiama**	**MRMD**
NSL-KDD	**99.17(30)**	86.56(4)	98.93(34)	82.97(6)	86.64(5)	99.16
AWID	**99.66(19)**	96.38(12)	96.89(41)	75.00(4)	93.35(8)	98.87
CIC-IDS2017	**91.16(55)**	88.76(8)	78.30(57)	72.51(6)	81.98(8)	82.85

Table 5: Class-wise accuracy (DT) comparison among different methods in network traffic dataset; Bold face results represent overall win.

Dataset	Type	CbFS	FAST	JC	DSbM	Nkiama	MRMD
NSL-KDD	Normal	98.45	92.26	98.3	98.1	71.18	**99.12**
	Dos	98.26	88.8	97.64	99.61	95.25	**99.63**
	R2L	75.41	2.27	68.51	78.72	14.46	**91.89**
	Probe	97.44	57.68	97.4	97.34	63.08	**98.66**
	U2R	**64.68**	0	61.15	61.54	52.78	63.46
AWID	Normal	**99.81**	98.78	99.72	99.45	99.95	99.62
	Attack	**97.92**	68.04	93.73	49.27	15.32	95.6

Table 6: Confusion matrix of CbFS for NSL-KDD dataset

Type	Normal	Dos	R2L	Probe	U2R
Normal	75859	341	464	359	31
Dos	908	52456	3	18	0
R2L	909	1	2827	7	5
Probe	278	24	7	13716	52
U2R	67	0	18	4	163

References

1. Cyber Security Report. https://docs.broadcom.com/doc/ istr-22-2017-en. Accessed on: 2022-07-05.
2. N. Magendiran and J. Jayaranjani, "An efficient fast clustering-based feature subset selection algorithm for high-dimensional data," *International journal of innovative research in science*, vol. 3, no. 1, pp. 405–408, 2014.
3. P. Moradi and M. Rostami, "A graph theoretic approach for unsupervised feature selection," *Engineering Applications of Artificial Intelligence*, vol. 44, pp. 33–45, 2015.
4. Q. Song, J. Ni, and G. Wang, "A fast clustering-based feature subset selection algorithm for high-dimensional data," *IEEE transactions on knowledge and data engineering*, vol. 25, no. 1, pp. 1–14, 2011.
5. G. Brown, A. Pocock, M.-J. Zhao, and M. Lujan, "Conditional likelihood´ maximization: a unifying framework for information theoretic feature selection," *The journal of machine learning research*, vol. 13, no. 1, pp. 27–66, 2012.
6. M. Bennasar, Y. Hicks, and R. Setchi, "Feature selection using joint mutual information maximization," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8520–8532, 2015.
7. M. Dash and H. Liu, "Feature selection for classification," *Intelligent data analysis*, vol. 1, no. 3, pp. 131–156, 1997.
8. S. Das, "Filters, wrappers and a boosting-based hybrid for feature selection," in *Icml*, vol. 1, pp. 74–81, 2001.
9. S. Sharmin, M. Shoyaib, A. A. Ali, M. A. H. Khan, and O. Chae, "Simultaneous feature selection and discretization based on mutual information," *Pattern Recognition*, vol. 91, pp. 162–174, 2019.
10. H. Yang and J. Moody, "Feature selection based on joint mutual information," in *Proceedings of international ICSC symposium on advances in intelligent data analysis*. Citeseer, pp. 22–25, 1999.
11. D. D. Lewis, "Feature selection and feature extract ion for text categorization," in *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992.
12. R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Transactions on neural networks*, vol. 5, no. 4, pp. 537–550, 1994.
13. N. X. Vinh, S. Zhou, J. Chan, and J. Bailey, "Can high-order dependencies improve mutual information based feature selection?" *Pattern Recognition*, vol. 53, pp. 46–58, 2016.
14. P. Roy, S. Sharmin, A. A. Ali, and M. Shoyaib, "Discretization and feature selection based on bias corrected mutual information considering high-order dependencies," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, pp. 830–842, 2020.
15. J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm," in *Feature extraction, construction and selection*. Springer, pp. 117–136, 1998.
16. T. Naghibi, S. Hoffmann, and B. Pfister, "A semidefinite programming based search strategy for feature selection with mutual information measure," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 8, pp. 1529–1541, 2014.
17. M. R. Gary and D. S. Johnson, "Computers and intractability: A guide to the theory of np-completeness," 1979.
18. K. Z. Mao, "Orthogonal forward selection and backward elimination algorithms for feature subset selection," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 1, pp. 629–634, 2004.
19. L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proceedings of the 20th international conference on machine learning (ICML-03)*, pp. 856–863, 2003.
20. K. Kira, L. A. Rendell *et al.*, "The feature selection problem: Traditional methods and a new algorithm," in *Aaai*, vol. 2, pp. 129–134, 1992.

21. I. Kononenko, "Estimating attributes: analysis and extensions of relief," in *European conference on machine learning*. Springer, pp. 171– 182, 1994.
22. M. A. Hall, "Correlation-based feature selection for machine learning," 1999.
23. W. Gao, L. Hu, and P. Zhang, "Feature redundancy term variation for mutual information-based feature selection," *Applied Intelligence*, vol. 50, no. 4, pp. 1272–1288, 2020.
24. H. Nkiama, S. Z. M. Said, and M. Saidu, "A subset feature elimination mechanism for intrusion detection system," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 4, pp. 148– 157, 2016.
25. T. O. Kvalseth, "Entropy and correlation: Some comments," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 17, no. 3, pp. 517– 519, 1987.
26. D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml
27. J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Computing Surveys (CSUR)*, vol. 50, no. 6, p. 94, 2018.
28. J. Alcala-Fdez,´ A. Fernandez,´ J. Luengo, J. Derrac, S. Garc´ıa, L. Sanchez, and F. Herrera, "Keel data-mining software tool: data set´ repository, integration of algorithms and experimental analysis framework." *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, 2011.
29. Nsl-kdd. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html
30. Awid-aegean wi-fi intrusion dataset. [Online]. Available: https://icsdweb.aegean.gr/awid/
31. Ids 2017. [Online]. Available: https://www.unb.ca/cic/datasets/ids2017.html
32. J. Demsar, "Statistical comparisons of classifiers over multiple data sets,"˜ The Journal of Machine Learning Research, vol. 7, pp. 1–30, 2006.
33. P. Nemenyi, "Distribution-free mulitple comparisons phd thesis princeton university princeton," 1963.
34. M. H. Tarek, M. M. H. U. Mazumder, S. Sharmin, M. S. Islam, M. Shoyaib, and M. M. Alam, "RHC: Cluster based feature reduction for network intrusion detections," in 2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC). IEEE, pp. 378–384, 2022.
35. M. H. Tarek, M. E. Kadir, S. Sharmin, A. A. Sajib, A. A. Ali, and M. Shoyaib, "Feature subset selection based on redundancy maximized clusters," in 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE, pp. 521–526, 2021.