Selection of Sub-optimal Feature set for Classification using Graph based Clustering

Md. Hasan Tarek
Department of Computer Science and Engineering
Begum Rokeya University, Rangpur
Rangpur, Bangladesh
Email: hasan.tarek@brur.ac.bd

***Abstract*—Feature selection transcends mere dimensionality reduction, serving as a strategic instrument that enhances model interpretability, generalization, optimizes computational efficiency, improves model accuracy, mitigates overfitting, by isolating the most predictive features, thereby facilitating the development of robust and generalizable machine learning models. The researchers introduced several feature selection techniques to select significant and pertinent features. They used a variety of techniques, including filter, wrapper and embedding methods. However, they are unable to choose the optimal features, and a number of them (such as the wrapper technique) rely on the classification algorithm. These methods, which also use correlation, distance measures, are unable to represent the intricate and non-linear interactions of the features. Therefore, considering the ability to capture both linear and non-linear interactions, recent years have seen a rise in the use of techniques based on mutual information (MI). Graph-based techniques using MI are becoming more and more popular due to their enhanced classification accuracy, resilience, and generalizability in fields such as bioinformatics, text mining, image classification, and network systems. In this study, we presented Feature Selection using Graph based Clustering (FSGC), a graph-based clustering technique that groups feature(s) with similar characteristics by combining the MI and clustering technique. In FSGC, the cluster(s) are formed in such a way that provide redundant and complementary information. The experimental results on twenty benchmark datasets from different domains demonstrate that FSGC performs better than other compared state-of-the-art approaches in the majority of cases. Furthermore, it is used to examine the effects of separating attack class traffic from regular network traffic on network intrusion detection (IDS) datasets.**

Keywords—Feature Selection, Mutual Information, Clustering, Graph, Minimum Spanning Tree, Network Intrusion Detection System.

I. INTRODUCTION

The process of feature selection is essential to build successful classification models, especially when working with high-dimensional datasets where irrelevant and redundant features can have a detrimental influence on computing efficiency and accuracy. Feature selection is an essential procedure that enhances model accuracy, generalization, and computing efficiency by choosing a subset of pertinent characteristics and removing unnecessary or noisy ones[1]. Finding a subset of features that are most informative for the target variable is the main goal in order to enhance the model interpretability and lower the possibility of overfitting [2]. Langley [3] divided various feature selection techniques into two major categories (filter and wrapper) according to how much they rely on the inductive process that will ultimately employ the chosen subset. The inductive algorithm serves as the evaluation function for wrapper methods, whereas filter methods operate independently of it. In addition to this, there are hybrid approaches that integrate the wrapper and filter mechanisms. Nkiama *et al.* [4] proposed a method, where the relevant feature subset is chosen from the remaining features using Recursive Feature Elimination after the features are ranked according to their individual strengths using the ANOVA F-test univariate filter method. Filter, wrapper, and embedding approaches are examples of traditional feature selection techniques that frequently fail to capture intricate dependencies between features, particularly in the presence of non-linear or higher-order interactions [2], [5], [6].

Mutual Information (MI), a non-linear dependence metric that can capture complex statistical relationships between variables, has drawn a lot of interest. It can detect both linear and non-linear relationships, which makes it particularly useful for finding pertinent features for classification problems [7], [8], [9]. However, when calculating the value of MI for a finite number of samples, one of the main disadvantages is that there exists some mistake (bias) [10], [11]. By employing the bias-corrected MI mechanism to further refine relevancy estimations, the influence of sample size disparities can be reduced and classification performance can be enhanced [12], [13], [14]. The bias of MI towards multi-valued features can be reduced by normalizing it. Estévez *et al*.[15] presented a method, Normalized Mutual Information Feature Selection (NMIFS) where the redundancy of the features are measured by calculating the average of normalized MI (NMI).

Conventional MI-based feature selection techniques frequently select overlapping features by using greedy tactics that ignore inter-feature redundancy [16]. Clustering based methods can play a

vital role which aims to group similar data points. Agglomerative hierarchical clustering stands out above other methods due to its ease of use and capacity to reveal nested structures through the iterative merging of the most similar clusters. Because of this, it is useful in fields including network intrusion detection systems, image analysis, bioinformatics, and social networks [17], [18], [19]. However, it is sensitive to early-stage verdicts and computationally challenging. Besides, cluster number specification affects the efficiency of hierarchical clustering algorithms and may hinder the achievement of the optimal feature set. To overcome this, researchers have investigated graph-based clustering techniques, such as spectral clustering, minimum spanning tree (MST) clustering, divide data using cut-based or connectivity-based procedures after modeling it as a graph which are more scalable and better at handling complex, non-convex structures [20]. These methods represent data as a graph in which dependencies, such as MI, correlation or Symmetric Uncertainty (SU), are represented as edges and features as nodes. Clustering these graph representations facilitates the discovery of representative subsets and groups of shared characteristics [21]. Recent work integrates feature selection with hierarchical clustering by interpreting single linkage dendrograms as Minimum Spanning Tree (MST), enabling simultaneous optimization of feature subsets and cluster structures while preserving information [22].

Graph based feature selection has become an appealing alternative by utilizing graphs' capacity to express complex interactions between features [23], [24]. Song *et al.* [25] presented a graph based clustering method namely Fast clustering bAsed feature Selection algoriThm (FAST) using a cut-based technique that uses Symmetric uncertainty as edge value to form different clusters with the help of Minimum Spanning Tree (MST). Jaganath and Sasikumar [26] introduced a method for grouping transactional data with similarity scores using MST. Nevertheless, they employed correlation similarity metrics, which are unable to detect non-linear relationships among the features. Magendiran and Jayaranjani [27] proposed another graph-based selection technique that makes advantage of MST. To decide on the final feature subset, Liu *et al.* [28] devised a MST-based Feature Clustering (MFC) approach that uses a variation of information metric as the edge value to build the MST and then make the clusters from it. These approaches, however, only choose one representative feature from each cluster, ignoring the possibility that involving more informative feature(s) could improve algorithm performance. All things considered, the combination of graph representations, MST clustering, and MI-based dependency measures presents a viable method for sub-optimal feature selection. This serves as the foundation for the method presented in this research.

In this paper, a Graph based clustering method is proposed and the contributions of this work are as the followings:

- Firstly, we presented Feature Selection using Graph based Clustering (FSGC), a clustering method to group the redundant as well as additional information (complementary) given features in the same cluster.
- Secondly, we used JBMI on each cluster to choose the most relevant feature along with other features that provide additional information in order to select the final feature set.
- Thirdly, in order to analyze the performance of FSGC rigorous experiments on twenty benchmark datasets are presented.
- Finally, FSGC is utilized on two well-known publicly accessible IDS datasets to see how well it distinguishes between typical and unusual traffic in the network system.

The remaining part of the paper is organized as follows: Existing works are discussed in Section II, whereas our suggested approach is described in Section III. Section IV then presents the experimental results, and section V provides a summary of the study.

II. Literature Review

Feature selection has been thoroughly researched as a way to enhance classification algorithm performance by removing redundant and unnecessary ones. It improves model correctness, generalization, and computational efficiency by selecting a subset of relevant attributes and eliminating those that are noisy and redundant. Methods of feature selection can be broadly divided into filter, wrapper, and embedding. Filters, regardless of the selected predictor, choose subsets of variables as a pre-processing step, makes it computationally efficient and independent of specific classifiers. Wrappers rate subsets of variables based on their prediction power by using the learning machine of interest as a black box. Embedded methods are often tailored to certain learning machines and carry out variable selection during training [1], [3], [9]. Based on the score each feature has acquired during the selection process, Nkiama *et al.* [4] provided a feature selection mechanism that seeks to both find the features that would increase the detection rate and eliminate non-relevant features. A decision tree-based classifier is coupled to a recursive feature reduction method in order to accomplish that goal. The appropriate relevant features were then found in order to identify the network's abnormal traffic. Traditional feature selection methods sometimes fall short in capturing complex feature dependencies, especially when non-linear or higher-order interactions are present [2], [5], [6], [29]. The ability of MI-based filter approaches to capture intricate variable relationships makes them especially popular. MI is capable of capturing both linear and non-linear relationships between the variables [7], [13]. It measures statistical dependence, capturing linear and

nonlinear relationships [30], and remains invariant under invertible, differentiable transformations like translations and rotations [30], [31].

One of the foundational works in MI-based feature selection by Battiti [32] namely Mutual Information based Feature Selection (MIFS), who presented an approach for choosing features having a high MI with the target class. The Mutual Information Feature Selection (MIFS) criterion is presented as follows

$$J_{MIFS}(f_i) = \mathrm{I}(f_i;\mathrm{C}) - \beta \sum_{f_j \in S} I(f_i; f_j) \quad (1)$$

Here the set of features that are now selected is denoted by *S*. In order to guarantee feature relevance (MI between a feature and a class), it incorporates the $\mathrm{I}(f_i;\mathrm{C})$ term; however, it also applies an adjustment to enforce low correlations with features that have already been chosen in *S*. This approach makes the assumption that we are building our final feature subset iteratively, selecting features one after the other. The variable parameter β in the MIFS criterion, which controls the relative relevance of the MI between the candidate feature and the previously chosen feature(s) with regard to the MI with the target class, must be set experimentally.

Later Yang and Moody [33] used Joint Mutual Information (JMI) to concentrate on enhancing complementary (MI between two features given the class label) information between features. In the study of Brown *et al.* [34], they demonstrated that prevalent heuristics for information-based feature selection are approximately iterative conditional likelihood maximizers. The JMI criteria for feature is in (2)

$$J_{JMI}(f_i) = \mathrm{I}(f_i;\mathrm{C}) - \frac{1}{|S|}\sum_{f_j \in S}\left(I(f_i; f_j) - I(f_i; f_j|C)\right) \quad (2)$$

In this case, f_i is the candidate feature that will be chosen, *S* is the feature set that has already been chosen, and these three terms stand for relevancy, redundancy (MI between two features) and complementary respectively. However, there can be bias involved in calculating the MI value for a limited number of instances, which could impair the model's effectiveness. Addressing this, Sharmin *et al*. [13] proposed a work, Joint Bias corrected Mutual Information (JBMI) that attempts to resolve it. They also calculated the associated critical value. The JBMI formula becomes as the following equation

$$J_{JBMI}(f_i) = \mathrm{I}(f_i;\mathrm{C}) - \frac{(\mathcal{J}-1)(\mathcal{K}-1)}{2Nln2} + \frac{1}{|S|}\sum_{f_j \in S}\left(I(f_i; f_j|C) - \frac{(\mathcal{J}-1)(\mathcal{J}-1)\mathcal{K}}{2Nln2} - \mathrm{I}(f_i; f_j) + \frac{(\mathcal{J}-1)(\mathcal{J}-1)}{2Nln2}\right) \quad (3)$$

Here, $\mathcal{J}, \mathcal{J}$ represent the feature intervals of f_i and f_j. The numbers of classes and total samples are denoted by $\mathcal{K}$, *N* respectively. Most of the feature selection methods discussed uses low-dimensional MI, limiting high-order dependency capture. The authors of [35] addresses this gap by analyzing the use of high-order dependencies in MI-based feature selection. They presented a technique called RelaxMRMR (rMRMR) that establishes a series of assumptions that permit high-dimensional MI to be broken down into low-dimensional terms. By easing the assumptions, they developed a systematic method to incorporate higher-order feature interactions. For a more accurate estimation of joint MI, Roy *et al.* [29] address the bias issue for the high-order interaction term. They start by figuring just how biased this term is. Additionally, they demonstrated that the χ^2 distribution is followed when selecting features using a χ^2 based search. They also offered Discretization and feature Selection based on bias corrected Mutual information, which is expanded by including simultaneous forward selection and backward elimination (DSbM_BE).

As demonstrated in [13], [29] relevancy, redundancy, complementary terms may be utilized to choose features using the critical values because they also follow the χ^2 distribution. When a feature shares information with others, it may be misinterpreted as poor by traditional feature redundancy metrics, even if it offers useful additional categorization information. In order to solve this, Gao *et al.* [36] suggested a redundancy term that assesses each feature's relevance to the target class and presented a method namely Min-Redundancy and Max-Dependency (MRMD). Naghibi *et al*. [37] employed a feature subset selection method called convex based relaxation approximation (COBRA), which uses semi-definite programming to search across the subset space.

Clustering based methods are introduced to address the limitations of the conventional MI based feature selection techniques. These methods attempt to group those features with similar characteristics. Agglomerative clustering and its variants have been the subject of a significant amount of research. A thorough analysis of the clustering methods was given by Xu and Wunsch [17], who also highlighted the value of hierarchical models in identifying multi-level data patterns. A quicker implementation of agglomerative clustering was introduced by Müllner [38], which decreased the runtime in real-world applications. However, early-stage merging faults cannot be fixed due to the irreversible nature of the technique, and the outcomes differ greatly depending on the linking mechanism used. It restricts the flexibility of agglomerative clustering in noisy and high-dimensional circumstances. Additionally, the cluster number specification may make it more difficult to obtain the optimal feature set and has an impact on the effectiveness of hierarchical clustering techniques. In order to overcome these constraints, the researchers have looked into graph-based clustering methods like MSTs.

Graph-based clustering techniques use nodes to represent features and edges to convey pairwise associations, which are usually quantified by correlation or mutual information. Finding groups of similar characteristics and choosing non-redundant, informative subsets are made easier by the creation of such graphs [20], [21]. These techniques have become more popular because of their capacity to capture intricate feature interactions and make it easier to identify representative subsets. It has been demonstrated that the underlying structure of feature spaces can be efficiently captured by graph-theoretical

methods, especially those that make use of MSTs [23], [24].The use of MSTs for cluster detection was initially presented in Zahn's [39] groundbreaking work, which also highlighted how they adhere to perceptual organization principles and showed how they can be applied in both low and high-dimensional areas. It can discover clusters without making assumptions about the distribution of data since it naturally captures hierarchical structures. Because MST-based clustering does not rely on presumptions regarding the geometric distribution of data, it is reliable across a wide range of application domains.

The Fast Clustering bAsed feature Selection algorithm (FAST), which has been proposed by Song *et al.* [25], employs MST to create the clusters. Nodes are the features, and the SU value represents the edge values. FAST operates in two phases. Graph-theoretic clustering techniques are used in the first phase to group features into clusters. To create the final subset of features, the most representative feature from each cluster that has a strong correlation with the target classes is chosen in the second stage. In order to ensure relative independence within clusters, the authors of [26] group similar features using correlation-based similarity measures. It refines the chosen feature subset using correlation measures and utilizes an MST-based approach for effective clustering. Another graph-centric clustering technique was presented by Magendiran and Jayaranjani[27] to aggregate the features into various, comparatively independent clusters. Liu *et al.* [28] proposed a supervised learning technique called MST-based Feature Clustering (MFC). The information-theoretic metric of variation of information is used to evaluate the relevance and redundancy of features. The approach chooses representative features that optimize relevance to the target label while minimizing pairwise redundancy during clustering. These methods select a representative feature from each cluster to obtain the final feature set.

The preceding discussion demonstrates that current studies based on clustering aim to group solely redundant features, neglecting the impact of including complementary features within the same cluster. In addition, the majority of these methods choose only one representative feature from each cluster. In this work, FSGC is introduced that combines MI and graph centric approach to address these concerns.

III. The Proposed Method for Feature Selection

This section contains a detailed discussion of our proposed study, Feature Selection using Graph based Clustering (FSGC). While attempting to incorporate features that offer additional information, FSGC aggregates redundant features in the same cluster. To get the final set of features F_s from the original feature set (F), it then chooses one or more features from the cluster(s) that was constructed. The phases taken in our proposed work is depicted in Figure. 1.

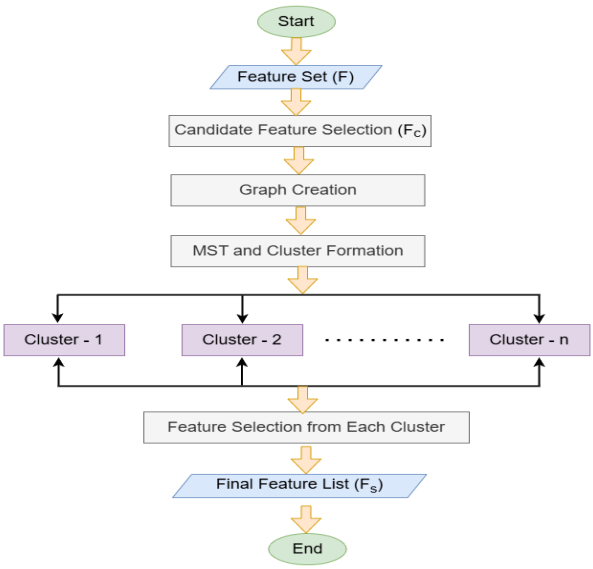

Figure 1: Proposed Method's Workflow.

Mutual Information (MI) quantifies how much information one random variable contains about another. Its value can be between 0 to ∞. MI can be normalized using different ways and some of them are discussed in [25], [40]. In FSGC, the relevance of a feature to the target class is measured by normalizing the MI value using Symmetric Uncertainty (SU). The SU_r (redundancy) value between two features f_i and f_j is calculated as follows

$$SU_r(f_i; f_j) = \frac{2 \times I(f_i; f_j)}{H(f_i) + H(f_j)} \quad (4)$$

where, $H(f_i)$ and $H(f_j)$ are the entropies of f_i and f_j respectively. Likewise, SU_c (complementary) between two features f_i and f_j given the class label C is computed as

$$SU_c(f_i; f_j | C) = \frac{2 \times I(f_i; f_j | C)}{H(f_i) + H(f_j)} \quad (5)$$

A. Graph and Cluster Formation

In a dataset, every feature is not equally important due to the noise and irrelevancy of the features. Therefore, removal of these irrelevant features is important. To remove these, bias corrected MI value between a feature and a class (Relevance) is computed using (6)

$$\mathrm{I}'(f_i; \mathrm{C}) = \mathrm{I}(f_i; \mathrm{C}) - \frac{(\mathcal{J}-1)(\mathcal{K}-1)}{2N \ln 2} \quad (6)$$

where C represents class labels, $\mathcal{J}$ denotes the number of intervals of f_i, $\mathcal{K}$ represents the number of classes and the total number of samples is N. The corresponding critical value of (6) is

$$\chi^2_C = \mathrm{I}(f_i; \mathrm{C}) \times 2\mathrm{N} \ln 2 \quad (7)$$

The features in the original feature set F are eliminated if their relevancy value falls below the corresponding χ^2 critical value. After this step we get our candidate features list F_c that will be used in the next phase of FSGC. **Line 1 - Line 8** describes this process depicted in **Algorithm 1**.

Algorithm 1 FSGC Algorithm

Require: Feature set, F
Ensure: Selected Feature Set, F_s

1: $F_s \leftarrow \emptyset; F_c \leftarrow \emptyset$
2: **for** $f_i \in F$ **do**
3: $J_R(f_i) \leftarrow f_i$ with respect to C using Eq. (6)
4: $\chi^2_C(R) \leftarrow$ Calculate using Eq. (7)
5: **if** $J_R(f_i) > \chi^2_C(R)$ **then**
6: $F_c \leftarrow F_c \cup f_i$
7: **end if**
8: **end for**
9: $G \leftarrow NULL$
10: **for** $(f_i, f_j) \in F_c$ **do**
11: $SU_r \leftarrow$Calculate $SU_{r_{ij}}$ using Eq. (4)
12: Insert f_i and f_j with $SU_{r_{ij}}$ as the edge value in G
13: **end for**
14: $minST \leftarrow Kruskal'sAlgorithm(G)$
15: $Forest \leftarrow minST$
16: **for** edge $e_{ij} \in Forest$ **do**
17: $SU_{c_{ij}} \leftarrow$Calculate $SU_{c_{ij}}$ using Eq. (5)
18: **if** $SU_{r_{ij}} < SU_{R_i}$ & $SU_{r_{ij}} < SU_{R_j}$ & $SU_{c_{ij}} < SU_{R_i}$ & $SU_{c_{ij}} < SU_{R_j}$ **then**
19: $Forest \leftarrow Forest - e_{ij}$
20: **end if**
21: **end for**
22: **for** Tree $T_i \in Forest$ **do**
23: Sort features in T_i in decreasing order based on corresponding $Relevance(R)$ value
24: $F'_s \leftarrow f_1; T_i \leftarrow T_i \setminus f_1$
25: **for** $f_j \in T_i$ **do**
26: $J_{JBMI}(f_j) \leftarrow$ Calculate using Eq. (3) and corresponding χ^2 critical value
27: **if** $J_{JBMI}(f_j) > \chi^2$ **then**
28: $F'_s \leftarrow F'_S \cup f_j$
29: **end if**
30: **end for**
31: $F_s \leftarrow F_S \cup F'_s$
32: **end for**
33: **return** F_s

To construct the fully connected graph (G), the features of F_c are used. The redundancy SU_r value between two features is used as the edge value of G. Afterwards. Kruskal's algorithm is applied on G to construct the MST. The reason MST-based clustering algorithms are utilized is that they capture global structure, are widely used in real world, and do not presuppose that data points are placed around centers or distanced by a typical geometric curve. Moreover, Kruskal's approach is appropriate for large feature sets and operates in $O(E \log E)$, where E denotes the number of edges (in this case, feature pairs).

Afterwards, to group the redundant features as well as complementary ones into the same cluster the condition in ***Line 18 of Algorithm*** 1 is applied. It indicates that if both of the connecting features' redundancy and complementary information are lower than their relevancy values to the target class, then they are most likely does not share same characteristics. Therefore, the connecting edge should be removed and they should be grouped into distinct clusters. The remaining clusters are generated in a similar manner.

Example: Figure. 2 illustrates the example of cluster creation. In this scenario, f_1, f_2, f_3, f_4 and f_5 represent the candidate feature list, while e_{ij} is the edge value linking feature f_i to feature f_j. Once the MST is formed with these features, there is only one connecting edge between the two features. To form the cluster(s), the edge(s) to be removed depend on **Line 18 of Algorithm 1**. Assuming in this case that e_{24} meets the criterion and its removal results in two clusters, {f_1, f_2, f_3} and {f_4, f_5}.

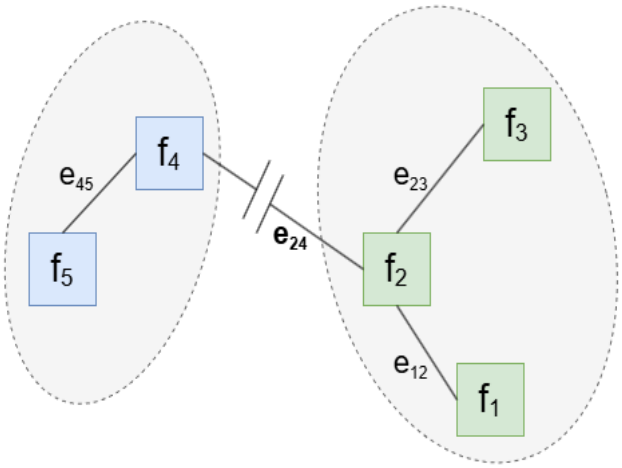

Figure 2: Cluster Creation Example.

B. *Final Feature Set Selection*

Subsequently, for the selection of the final set of features (F_s), *JBMI* is applied in each cluster. It selects the feature with the most relevant score and then check for other features if they can provide complementary information regarding the class. If the condition in **Algorithm 1 Lines 27-29** satisfies, then that feature is also included in the final feature set F_s. The overall feature selection process of FSGC method is shown in **Algorithm 1**. The experimental results are generated using this selected feature set, which is examined in the part that follows.

IV. Experimental Results and Analysis

In this section, we compute and conduct several tests and present the results in comparison with Five other state-of-the-art methods to show the efficacy of our proposed method, FSGC. We have also discussed the dataset description and implementation details in here.

A. *Dataset Description and Experimental Setup*

Twenty benchmark datasets from various domains are chosen from the UCI machine learning repository [41] and Knowledge Extraction based on Evolutionary Learning (KEEL) [42] which are popularly used in

different works in order to assess the effectiveness of our proposed approach. Moreover two network intrusion detection datasets (IDSs), AWID [43] and NSL-KDD [44] are chosen to check the performance of anomalous traffic classification using FSGC. Table 1 describes the dataset information.

Table 1: Dataset Description

Dataset	Feature	Class	Instances
Iris	4	3	150
Appendicitis	7	2	106
Ecoli	7	2	336
Pima	8	2	768
Glass	9	6	214
Saheart	9	2	462
Heart	13	2	270
Cleveland	13	5	297
Marketing	13	9	6876
Vehicle	18	2	846
Hepatitis	19	2	80
Waveform	21	3	5000
Thyroid	21	3	7200
Parkinsons	22	2	195
Steel	27	7	1941
Dermatology	34	6	366
Spectfheart	44	2	267
Sonar	60	2	208
Coil2000	85	2	9822
Madelon	500	2	2600
Security Dataset			
AWID	78	2	575315
NSL-KDD	42	5	148517

To find out the experimental result, five state-of-the-arts methods, FAST, JMI with COBRA (JC), DSbM_BE, rMRMR and MRMD are compared with our proposed work. The amount of features that FSGC selects is utilized to generate the outcomes of the two ranking methods, rMRMR and MRMD. Linear Support Vector Machine (SVM) algorithm is used to obtain the result with 10-fold cross validation (10-CV) technique and five equal width discretization. Also, to get further experimental result on IDSs dataset another method by Nkiama et al. [4] is compared and in this setting Decision tree (DT) classifier is used to detect the anomalous traffic.

B. Result and Discussion

In this section, FSGC with other methods results on the datasets in Table 1 are presented to show the superiority of our method's performance than others. Table 2 presents the accuracy results of FSGC with other comparative methods. The results indicate that overall accuracies of the proposed method are better than those are compared with. For example, in Hepatitis dataset the accuracy of FSGC is *85%*. Though FAST achieve similar accuracy however the number of features selected by FSGC(*4*) is less than FAST(*5*). Although the accuracy of the suggested method (*74.10%*) in the Sonar dataset is somewhat lower than that of DSbM_BE (*76.40%*), the number of features chosen by FSGC is substantially lower. This depicts that FSGC selects the relevant features and remove the redundant ones. The chosen feature number, however, is greater than the comparative approaches in some datasets (such as *Steel*). This is because it uses JBMI to identify the most essential features for each cluster, together with those that offer additional information to improve performance. Additionally, we have displayed the number of FSGC wins, ties, or losses comparing with others. It demonstrates that our proposed approach typically outperforms other state-of-the-art methods.

Moreover t-test with a *5%* level of significance is conducted to provide a comprehensive understanding of the superiority of FSGC. The result shows the number of significant win or loss of ours with other methods presented in the third row from the last in Table 2. Furthermore, another significance test used in different papers, Friedman rank test [45] is also conducted to elucidate the upper hand of our approach. After rejecting the null hypothesis, it applies the Nemenyi test [46] to compare which method's performance is significant. This result indicates that FSGC ranks highest among the other methods in this table's second-to-last row. The table's final row further demonstrates that, at *95%* confidence interval, the Friedman rank test of FSGC performs noticeably better than other state-of-the-art techniques indicated by the symbol √. Additionaly, f-score is also computer to get a better insight of FSGC shown in Table 3.

Table 2: Accuracy (SVM) comparison with state-of-the-art methods. Bold face results indicate an overall win, while (*) and (◦) indicate a significant win or loss related to that approach.

Dataset	My method	FAST	JC	DSbM_BE	MRMD	rMRMR
Iris	0.940(2)	0.880(1)*	0.913(2)	0.927(2)	**0.953(2)**	0.933(2)
Appendicitis	**0.869(1)**	**0.869(1)**	0.850(4)	0.800(2)	0.750(1)*	0.800(1)*
Ecoli	**0.970(3)**	0.967(2)	0.944(5)*	0.947(3)	0.941(3)*	0.944(3)*
Pima	0.744(5)	0.741(2)	0.736(8)	**0.745(5)**	0.649(5)*	0.722(5)
Glass	**0.528(3)**	0.444(2)*	0.517(7)	0.509(4)	0.443(3)*	0.509(3)
Saheart	0.721(5)	0.654(1)	0.717(9)	0.711(5)	**0.730(5)**	0.717(5)
Heart	**0.833(9)**	0.800(4)	0.811(10)	0.793(6)	0.826(9)	0.830(9)
Cleveland	**0.613(6)**	0.589(4)	0.534(13)*	0.538(4)*	0.553(6)*	0.522(6)*
Marketing	**0.32(10)**	0.263(1)*	0.313(10)	0.305(4)*	0.316(10)	0.319(10)
Vehicle	**0.749(6)**	**0.749(1)**	0.744(16)*	0.744(2)*	0.744(6)*	0.744(6)*
Hepatitis	**0.850(4)**	**0.85(5)**	0.822(18)	0.833(4)	0.811(4)	0.811(4)
Waveform	0.847(19)	0.701(6)*	0.808(13)*	0.653(3)*	**0.854(19)**	0.84(19)
Thyroid	0.932(8)	0.932(6)	0.929(18)*	0.925(4)*	**0.933(8)**	0.931(8)
Parkinsons	0.845(12)	**0.850(5)**	0.845(14)	0.815(11)	0.81(12)	0.84(12)
Steel	**0.709(22)**	0.563(7)*	0.696(20)	0.66(13)*	0.679(22)*	0.688(22)
Dermatology	0.964(32)	0.762(8)*	0.953(22)	0.96(30)	**0.97(32)**	0.955(32)
Spectfheart	0.79(10)	**0.802(7)**	0.743(31)*	0.796(15)	0.786(10)	0.786(10)
Sonar	0.741(6)	0.707(13)	0.727(60)	**0.764(15)**	0.645(6)*	0.741(6)
Coil2000	0.94(39)	0.94(17)	0.940(85)*	0.940(6)*	0.94(39)*	0.94(39)*
Madelon	0.599(11)	0.566(60)*	0.543(500)*	**0.613(11)**	0.552(11)*	0.578(11)
W/T/L	**-**	**13/5/2**	**18/2/0**	**15/1/4**	**14/1/5**	**18/2/0**
Sig. W/L	**-**	**7/0**	**8/0**	**7/0**	**10/0**	**5/0**
Avg. Rank	1.83	3.68	4.08	3.80	3.85	3.78
F-Rank Test	-	√	√	√	√	√

Table 3: F-score (SVM) comparison with state-of-the-art methods. Bold face results indicate an overall win, while (*) and (◦) indicate a significant win or loss related to that approach.

Dataset	My method	FAST	JC	DSbM_BE	MRMD	rMRMR
Iris	0.940(2)	0.875(1)*	0.919(2)	0.936(2)	**0.958(2)**	0.939(2)
Appendicitis	**0.854(1)**	**0.854(1)**	0.785(4)	0.692(2)*	0.429(1)*	0.688(1)*
Ecoli	**0.966(3)**	0.963(2)	0.712(5)*	0.752(3)*	0.485(3)*	0.692(3)*
Pima	**0.741(5)**	0.722(2)	0.706(8)*	0.718(5)	0.500(5)*	0.690(5)*
Glass	**0.484(3)**	0.391(2)*	0.396(7)*	0.347(4)*	0.255(3)*	0.386(3)*
Saheart	**0.714(5)**	0.566(1)*	0.671(9)	0.668(5)	0.684(5)	0.670(5)
Heart	**0.833(9)**	0.799(4)	0.81(10)	0.792(6)	0.824(9)	0.830(9)
Cleveland	**0.569(6)**	0.552(4)	0.303(13)*	0.265(4)*	0.281(6)*	0.291(6)*
Marketing	**0.233(10)**	0.156(1)*	0.207(10)*	0.184(4)*	0.195(10)*	0.214(10)*
Vehicle	**0.642(6)**	**0.642(1)**	0.427(16)*	0.427(2)*	0.427(6)*	0.427(6)*
Hepatitis	**0.836(4)**	**0.836(5)**	0.704(18)	0.738(4)	0.604(4)*	0.659(4)*
Waveform	0.846(19)	0.694(6)*	0.808(13)*	0.649(3)*	**0.854(19)**	0.84(19)
Thyroid	**0.903(8)**	0.903(6)	0.446(18)*	0.320(4)*	0.526(8)*	0.504(8)*
Parkinsons	0.835(12)	**0.851(5)**	0.772(14)	0.731(11)*	0.691(12)*	0.766(12)
Steel	**0.705(22)**	0.532(7)*	0.704(20)	0.681(13)	0.615(22)*	0.655(22)*
Dermatology	0.964(32)	0.712(8)*	0.948(22)	0.96(30)	**0.968(32)**	0.951(32)
Spectfheart	0.755(10)	**0.758(7)**	0.592(31)*	0.637(15)*	0.44(10)*	0.537(10)*
Sonar	0.737(6)	0.696(13)	0.727(60)	**0.77(15)**	0.646(6)*	0.747(6)
Coil2000	**0.911(39)**	**0.911(17)**	0.484(85)*	0.485(6)*	0.485(39)*	0.484(39)*
Madelon	0.598(11)	0.566(60)*	0.544(500)*	**0.613(11)**	0.552(11)*	0.579(11)
W/T/L	-	**13/5/2**	**20/0/0**	**18/0/2**	**17/0/3**	**19/0/1**
Sig. W/L	-	**8/0**	**11/0**	**11/0**	**15/0**	**12/0**
Avg. Rank	1.53	3.38	3.90	4.00	4.30	3.90
F-Rank Test	-	√	√	√	√	√

Network Intrusion Detection Datasets Result and Discussion:

This section discusses the result of IDSs data in a more detailed way. From Table 1, it can be seen that NSL-KDD dataset is a multi-class dataset. It contains five class labels which are Normal, DoS, R2L, Probe and U2R discussed in [44]. We have created two versions of FSGC, which we will refer to as NSL-KDD (B) and NSL-KDD (M), in order to gain a better understanding of FSGCs application to this dataset. NSL-KDD (B) will have two (Binary) classes as target labels (Normal, Attack) where all attack traffics will be treated as *Attack* classes, with the exception of *Normal* traffic flows while NSL-KDD (M) will have all the five classes.

Results from FSGC and all other approaches on IDS datasets are summarized in Table 4 and 5. From these two tables, we can observe that FSGC performs better than all other comparative approaches when it comes to recognizing each single class in a multi-class

dataset or solving a binary class problem, as demonstrated by the accuracy and F-score results. Binary and multiclass results are examined in the following sections to provide a clearer picture of the IDS datasets outcomes.

Table 4: Overall Accuracy result using Decision Tree on Network Intrusion Detection Dataset

Dataset	My Method	FAST	JC	DSbM_BE	rMRMR	Nkiama et al.
AWID	**0.9966(21)**	0.9638(12)	0.9925(41)	0.7575(3)	0.9890	0.9341(8)
NSL-KDD (B)	**0.9757(32)**	0.8686(5)	0.9651(32)	0.8297(6)	0.9679	0.8418(4)
NSL-KDD (M)	**0.9765(34)**	0.8531(8)	0.9716(31)	0.7981(10)	0.9743	0.7749(4)

Table 5: Overall F-score result using Decision Tree on Network Intrusion Detection Dataset

Dataset	My Method	FAST	JC	DSbM_BE	rMRMR	Nkiama et al.
AWID	**0.9966**	0.9622	0.9737	0.7898	0.9891	0.9116
NSL-KDD (B)	**0.9757**	0.867	0.9653	0.8468	0.968	0.8416
NSL-KDD (M)	**0.9762**	0.8429	0.8764	0.5569	0.8836	0.7718

1) Binary-class Dataset

Figure 3 and 4 presents the accuracy and F-score results of AWID binary class (Normal and Attack) dataset's result respectively. From it, we can notice that while FSGC and other approaches are quite equal in regular traffic analysis, FSGC performs significantly better than all other methods with the exception of the rMRMR method in attack class data classification. This is due to the fact that rMRMR can also record feature interaction information similar to ours. Moreover, when FSGC is applied to the NSL-KDD (B) dataset, the accuracy and F-score results are fairly comparable shown in Figure. 5 and 6 accordingly. This was made possible by choosing the most pertinent and complementary feature set, which aided in capturing the feature interaction information from the cluster formation by FSGC.

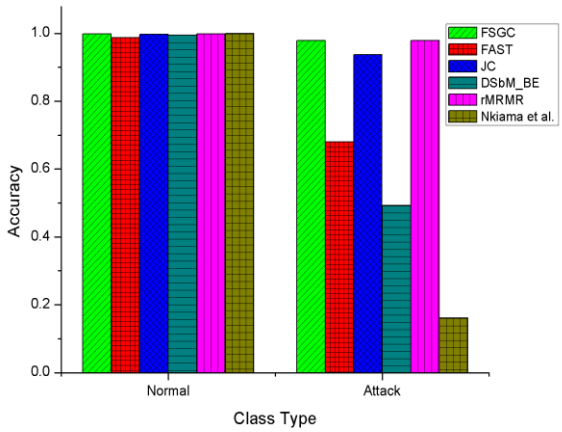

Figure 3: Class wise accuracy FSGC and other state-of-the-art methods on AWID dataset.

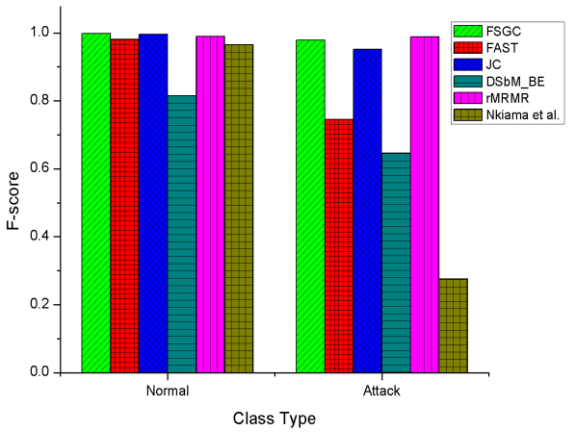

Figure 4: Class wise F-score of FSGC and other state-of-the-art methods on AWID dataset.

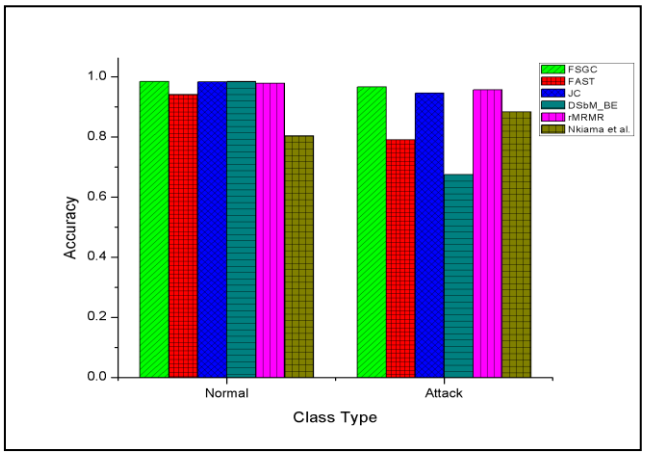

Figure 5: Class wise accuracy of FSGC and other state-of-the-art methods on NSL-KDD (B).

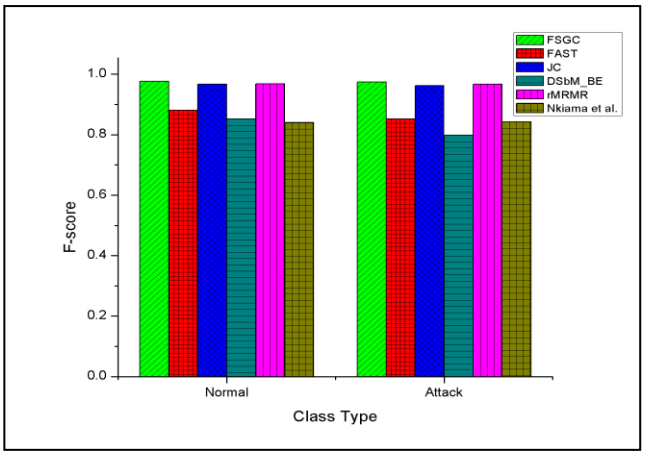

Figure 6: Class wise F-score of FSGC and other state-of-the-art methods on NSL-KDD (B).

2) Multi-class Dataset

To observe how well FSGC handles multi-class IDS datasets, it is additionally applied to the NSL-KDD (M) dataset. We can observe from the accuracy comparison of the class results shown in Figure 7 that the FSGC performs more effectively or comparably to other comparative ones when it comes to classifying distinct classes. Additionally, in order to obtain a better understanding of the results displayed in Figure 8, we have also computed the F-scores of these approaches. Apart from these results, to observe the various class identification capabilities of our proposed FSGC approach, we present the confusion matrix result displayed in Table 6.

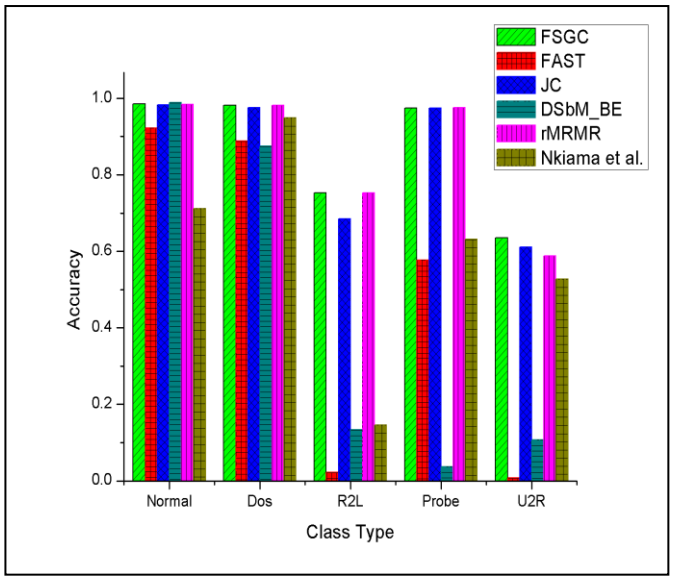

Figure 7: Class wise accuracy of FSGC and other comparative methods on NSL-KDD (M).

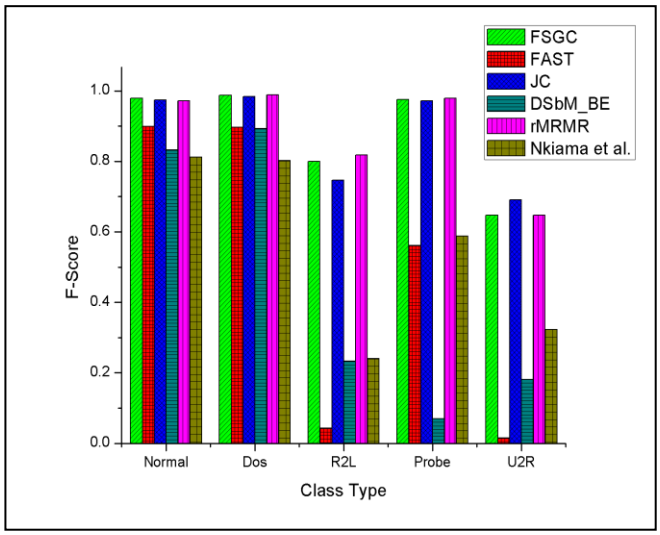

Figure 8: Class wise F-score of FSGC and other comparative methods on NSL-KDD (M).

Table 6: Confusion Matrix of NSL-KDD (M)

	Normal	Dos	R2L	Probe	U2R
Normal	75909	328	453	338	26
Dos	955	52414	5	11	0
R2L	911	3	2822	5	8
Probe	276	25	5	13722	49
U2R	69	0	19	4	160

V. CONCLUSION

The work Feature Selection using Graph based Clustering (FSGC) is presented in this paper. To obtain the candidate feature set, it first eliminates the features that are not relevant. This set is used to create a fully connected graph with symmetric uncertainty values serving as the graph's edges and features acting as its nodes. The Minimum Spanning Tree is then built using Kruskal's algorithm. Clusters are subsequently formed from it in a way that groups similar features together with complementary information. JBMI is then performed to each cluster to capture the most crucial and complementary information provided by the features in order to produce the final feature set. The performance of the FSGC is examined using twenty publicly available benchmark datasets from various fields. The findings demonstrate that FSGC performs better than other state-of-the-art techniques in the majority of instances, both in terms of accuracy and F-scores. Furthermore, it is applied on network intrusion detection system datasets, demonstrating encouraging results in distinguishing abnormal traffic from regular traffic in both binary and multi-class datasets.

REFERENCES

[1] I. Guyon and A. Elisseeff, “An Introduction to Variable and Feature Selection,” *Journal of Machine Learning Research*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[2] M. Dash and H. Liu, “Feature selection for classification,” *Intelligent Data Analysis*, vol. 1, no. 1, pp. 131–156, Jan. 1997, doi: 10.1016/S1088-467X(97)00008-5.

[3] P. Langley, “Selection of Relevant Features in Machine Learning.:,” Defense Technical Information Center, Fort Belvoir, VA, Nov. 1994. doi: 10.21236/ADA292575.

[4] H. Nkiama, S. Zainudeen, and M. Saidu, “A Subset Feature Elimination Mechanism for Intrusion Detection System,” *ijacsa*, vol. 7, no. 4, 2016, doi: 10.14569/IJACSA.2016.070419.

[5] H. Almuallim and T. G. Dietterich, “Learning With Many Irrelevant Features.,” in *AAAI*, 1991, pp. 547–552. Accessed: Dec. 23, 2025. [Online]. Available: https://web.engr.oregonstate.edu/~tgd/publications/aaai91-focus.ps.gz

[6] “Irrelevant Features and the Subset Selection Problem,” in *Machine Learning Proceedings 1994*, Morgan Kaufmann, 1994, pp. 121–129. doi: 10.1016/B978-1-55860-335-6.50023-4.

[7] “Estimating mutual information | Phys. Rev. E.” Accessed: Dec. 23, 2025. [Online]. Available: https://journals.aps.org/pre/abstract/10.1103/PhysRevE.69.066138

[8] M. A. Ambusaidi, X. He, P. Nanda, and Z. Tan, “Building an Intrusion Detection System Using a Filter-Based Feature Selection Algorithm,” *IEEE Transactions on Computers*, vol. 65, no. 10, pp. 2986–2998, Oct. 2016, doi: 10.1109/TC.2016.2519914.

[9] Md. H. Tarek, Md. 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)*, Dec. 2021, pp. 521–526. doi: 10.1109/ICMLA52953.2021.00087.

[10] Stefano Panzeri and Alessandro Treves, “Analytical estimates of limited sampling biases in different information measures,” *Network*, vol. 7, no. 1, p. 87, Feb. 1996, doi: 10.1088/0954-898X/7/1/006.

[11] “Correcting for the Sampling Bias Problem in Spike Train Information Measures | Journal of Neurophysiology | American Physiological Society.” Accessed: Dec. 23, 2025. [Online]. Available: https://journals.physiology.org/doi/full/10.1152/jn.00559.2007

[12] B. C. Ross, “Mutual Information between Discrete and Continuous Data Sets,” *PLOS ONE*, vol. 9, no. 2, p. e87357, Feb. 2014, doi: 10.1371/journal.pone.0087357.

[13] 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, July 2019, doi: 10.1016/j.patcog.2019.02.016.

[14] Md. H. Tarek, Md. M. H. U. Mazumder, S. Sharmin, Md. 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)*, Jan. 2022, pp. 378–384. doi: 10.1109/CCNC49033.2022.9700680.

[15] P. A. Estevez, M. Tesmer, C. A. Perez, and J. M. Zurada, “Normalized Mutual Information Feature Selection,” *IEEE Transactions on Neural Networks*, vol. 20, no. 2, pp. 189–201, Feb. 2009, doi: 10.1109/TNN.2008.2005601.

[16] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005, doi: 10.1109/TPAMI.2005.159.

[17] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, May 2005, doi: 10.1109/TNN.2005.845141.

[18] J. Song, Z. Zhu, and C. Price, "Feature Grouping for Intrusion Detection System Based on Hierarchical Clustering," in *Availability, Reliability, and Security in Information Systems*, S. Teufel, T. A. Min, I. You, and E. Weippl, Eds., Cham: Springer International Publishing, 2014, pp. 270–280. doi: 10.1007/978-3-319-10975-6_21.

[19] A. Oubaouzine, T. Ouaderhman, and H. Chamlal, "A graph partitioning-based hybrid feature selection method in microarray datasets," *Knowl Inf Syst*, vol. 67, no. 1, pp. 633–660, Jan. 2025, doi: 10.1007/s10115-024-02292-3.

[20] A. Ng, M. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems*, MIT Press, 2001. Accessed: Dec. 23, 2025. [Online]. Available: https://proceedings.neurips.cc/paper/2001/hash/801272ee79cfde7fa5960571fee36b9b-Abstract.html

[21] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, Jan. 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[22] M. Labbé, M. Landete, and M. Leal, "Dendrograms, minimum spanning trees and feature selection," *European Journal of Operational Research*, vol. 308, no. 2, pp. 555–567, July 2023, doi: 10.1016/j.ejor.2022.11.031.

[23] "Data Clustering: Theory, Algorithms, and Applications | SIAM Publications Library," ASA-SIAM Series on Statistics and Applied Mathematics. Accessed: Dec. 23, 2025. [Online]. Available: https://epubs.siam.org/doi/book/10.1137/1.9780898718348

[24] A. Kumar and J. Yadav, "Minimum spanning tree clustering approach for effective feature partitioning in multi-view ensemble learning," *Knowl Inf Syst*, vol. 66, no. 11, pp. 6785–6813, Nov. 2024, doi: 10.1007/s10115-024-02182-8.

[25] 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, Jan. 2013, doi: 10.1109/TKDE.2011.181.

[26] K. Jaganath, M. P. Sasikumar, and I. Me, "Graph Clustering and Feature Selection for High Dimensional Data," *International Journal of Innovative Research in Computer and Communication Engineering*, no. 1, 2007.

[27] N. Magendiran, J. Jayaranjani, and P. Scholar, "An Efficient Fast Clustering-Based Feature Subset Selection Algorithm for High- Dimensional Data," vol. 3, no. 1.

[28] Q. Liu, J. Zhang, J. Xiao, H. Zhu, and Q. Zhao, "A Supervised Feature Selection Algorithm through Minimum Spanning Tree Clustering," in *2014 IEEE 26th International Conference on Tools with Artificial Intelligence*, Nov. 2014, pp. 264–271. doi: 10.1109/ICTAI.2014.47.

[29] 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 *Advances in Knowledge Discovery and Data Mining*, H. W. Lauw, R. C.-W. Wong, A. Ntoulas, E.-P. Lim, S.-K. Ng, and S. J. Pan, Eds., Cham: Springer International Publishing, 2020, pp. 830–842. doi: 10.1007/978-3-030-47426-3_64.

[30] T. M. Cover and J. A. Thomas, *Elements of information theory*. Hoboken, NJ: Wiley-Interscience, 2001.

[31] S. Kullback, *Information Theory and Statistics*. Courier Corporation, 1997.

[32] 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, July 1994, doi: 10.1109/72.298224.

[33] H. Yang and J. Moody, "Data Visualization and Feature Selection: New Algorithms for Nongaussian Data," in *Advances in Neural Information Processing Systems*, MIT Press, 1999. Accessed: Dec. 23, 2025. [Online]. Available: https://proceedings.neurips.cc/paper/1999/hash/8c01a75941549a705cf7275e41b21f0d-Abstract.html

[34] "Conditional likelihood maximisation: a unifying framework for information theoretic feature selection: The Journal of Machine Learning Research: Vol 13, No null." Accessed: Dec. 23, 2025. [Online]. Available: https://dl.acm.org/doi/abs/10.5555/2188385.2188387

[35] 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, May 2016, doi: 10.1016/j.patcog.2015.11.007.

[36] W. Gao, L. Hu, and P. Zhang, "Feature redundancy term variation for mutual information-based feature selection," *Appl Intell*, vol. 50, no. 4, pp. 1272–1288, Apr. 2020, doi: 10.1007/s10489-019-01597-z.

[37] 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, Aug. 2015, doi: 10.1109/TPAMI.2014.2372791.

[38] D. Müllner, “Modern hierarchical, agglomerative clustering algorithms,” arXiv.org. Accessed: Dec. 23, 2025. [Online]. Available: https://arxiv.org/abs/1109.2378v1

[39] C. T. Zahn, “Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters,” *IEEE Transactions on Computers*, vol. C–20, no. 1, pp. 68–86, Jan. 1971, doi: 10.1109/T-C.1971.223083.

[40] T. O. Kvalseth, “Entropy and Correlation: Some Comments,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 17, no. 3, pp. 517–519, May 1987, doi: 10.1109/TSMC.1987.4309069.

[41] “Home - UCI Machine Learning Repository.” Accessed: Dec. 23, 2025. [Online]. Available: http://archive.ics.uci.edu/

[42] J. Alcalá-Fdez *et al.*, “KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework”.

[43] “AWID - Aegean Wi-Fi Intrusion Dataset.” Accessed: Dec. 23, 2025. [Online]. Available: https://icsdweb.aegean.gr/awid/awid2

[44] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, July 2009, pp. 1–6. doi: 10.1109/CISDA.2009.5356528.

[45] J. Dems˘ar and J. Demsar, “Statistical Comparisons of Classifiers over Multiple Data Sets”.

[46] “DISTRIBUTION-FREE MULTIPLE COMPARISONS. - ProQuest.” Accessed: Dec. 23, 2025. [Online]. Available: https://www.proquest.com/openview/c1f3e8829e8351e9c2a1c5e51778c6cf/1?pq-origsite=gscholar&cbl=18750&diss=y