DOI: 10.51984/SUCP.V4I1.3916

المؤتمر العلمي الأول للتطبيقات الهندسية (ICEA'2024)
The First Scientific Conference on Engineering Applications (ICEA'2024)
Conference homepage: www.icea.ly

An Approach to Evaluating Clustering Quality for Network Anomaly Detection

*Eljilani Hmouda[a], Wei Li[b], Ling Wang[b], Ajoy Kumar[b]

[a]College of Science and Mathematics, 320 Stanley Ave, Greenwood, SC 29649, USA
[b]College of Computing and Engineering, 3300 S University Dr, Fort Lauderdale, FL 33328, USA

Keywords:
clustering evaluation
v-measure
f-measure

ABSTRACT

Clustering is a fundamental task in unsupervised learning and is important for extracting interesting patterns and structures within data. Evaluating the quality of clustering algorithms is a complex task, often requiring a balance between homogeneity and completeness. In this paper, we apply V-measure as an evaluation metric that effectively determines the clustering quality and balances these two aspects by computing their harmonic mean to find the best features. We explore the theoretical foundations of V-measure, its calculation, and strategies for optimizing clustering performance to reduce data dimensionality, maintain low false alerts, and high detection rate for anomaly detection in binary intrusion classification. Our findings highlight significant reductions in dimensionality and data volume, coupled with low false positive and false negative rates, thereby enhancing detection accuracy.

منهجية لتقييم جودة التجميع لاكتشاف الثغرات في الشبكة

*أجوي كومار[1] و وينق لانق[2] و وي لي[2] و الجيلاني حمودة[2]

[1]كلية العلوم والرياضيات، 320 ستانلي افنيو، قرينوود، كالورنيا الجنوبية، 29646 الولايات المتحدة الأمريكية
[2]كلية الحوسبة والهندسة، 3300 جنوب الجامعة درايف، فورت لدردل، فلوريدا، 33328 الولايات المتحدة الأمريكية

الكلمات المفتاحية:
تقييم التجميع
مقياس v
مقياس f.

الملخص

إن التجميع هو مهمة أساسية في التعلم غير الخاضع للإشراف وهو مهم لاستخراج الأنماط والهياكل المثيرة للاهتمام داخل البيانات. إن تقييم جودة خوارزميات التجميع مهمة معقدة، وغالبًا ما تتطلب التوازن بين التجانس والاكتمال. في هذه الورقة، نطبق مقياس الدقة كمقياس تقييم يحدد جودة التجميع بشكل فعال ويوازن بين هذين الجانبين من خلال حساب متوسطهما التوافقي للعثور على أفضل الميزات. نستكشف الأسس النظرية لمقياس الدقة وحسابه واستراتيجيات تحسين أداء التجميع لتقليل أبعاد البيانات والحفاظ على تنبيهات خاطئة منخفضة ومعدل اكتشاف مرتفع لاكتشاف الشذوذ في تصنيف التطفل الثنائي. تسلط نتائجنا الضوء على انخفاض كبير في الأبعاد وحجم البيانات، إلى جانب انخفاض معدلات الإيجابيات الخاطئة والسلبية الخاطئة، وبالتالي تعزيز دقة الكشف.

1. Introduction

Clustering techniques of unsupervised learning enable the discovery of inherent structures and patterns within data without predefined labels [1]. By grouping similar data points into clusters, these techniques facilitate insightful analysis and data understanding. Various algorithms such as K-means, hierarchical clustering, and DBSCAN present diverse approaches to partition data based on similarity metrics. Each algorithm has unique strengths suited to different data distributions and clustering objectives, making selection important for effective analysis and interpretation of clustered results [2]. Cluster evaluation measures are important in evaluating the quality and effectiveness of clustering algorithms. They provide quantitative insights into how well clusters represent underlying data patterns, balancing metrics like homogeneity and completeness. Evaluating these measures involves comparing cluster assignments against ground truth or employing intrinsic methods that analyze internal cluster cohesion and separation [3].

*Corresponding author:
E-mail addresses: ehmouda@lander.edu, (W. Li) lwei@nova.edu, (L. Wang) lingwang@nova.edu, (A. Kumar) akumar@nova.edu
Article History : Received 11 March 2024 - Received in revised form 19 September 2024 - Accepted 15 October 2024

Network Intrusion Detection Systems (NIDS) are significant for protecting digital infrastructures by identifying and mitigating unauthorized access and malicious activities. These systems analyze network traffic patterns in real-time, detecting anomalies and potential threats that deviate from established baselines. Through machine learning techniques like clustering, NIDS enhance anomaly detection by identifying unusual network behavior and distinguishing it from normal traffic [4]. In this study, we will leverage the CICIDS-2017 dataset for our experiments, a comprehensive dataset widely used in cybersecurity research. Effective deployment of NIDS requires robust clustering algorithms and evaluation metrics to ensure high detection accuracy and minimal false positives.

Evaluation metrics such as homogeneity and completeness are vital in reducing false positives and negatives. Homogeneity measures the purity of clusters concerning their data points, while completeness ensures that all related data points are grouped within the same cluster [5]. High scores in these metrics within NIDS mean that similar types of traffic whether normal or anomalous are accurately classified. This proper evaluation significantly reduces the likelihood of normal traffic being mistakenly flagged as an intrusion or actual intrusions going undetected, thereby improving the reliability of the system.

Internal validation techniques, such as analysing intra-cluster cohesion and inter-cluster separation, further enhance model interpretability. In the context of NIDS, these evaluations demonstrate how effectively the system can differentiate between various network behaviors. This clarity is essential for security analysts, allowing them to understand and trust the detection process, thereby ensuring that the NIDS is not only accurate but also transparent in its operations [5].

V-measure is an entropy-based external cluster evaluation metric [5]. It is designed to assess the quality of clustering by measuring two key aspects: homogeneity and completeness. Homogeneity ensures that all data points within a cluster belong to a single class, while completeness ensures that all data points of a given class are assigned to the same cluster. V-measure is calculated as the harmonic mean of these two scores, providing a balanced evaluation of clustering performance. Unlike other metrics, V-measure does not require a mapping between clusters and classes, and it evaluates independently of the dataset size, clustering algorithm, number of clusters, and number of classes. This makes it a comprehensive and robust tool for cluster evaluation. V-measure also surpasses other metrics like Q0 [6] and variation of information, which evaluate only homogeneity or completeness separately.

In this paper, we explore the application of V-measure as a metric for evaluating clustering techniques within the context of network intrusion detection systems. By balancing homogeneity and completeness, V-measure offers a nuanced approach to evaluating clustering quality, important for optimizing anomaly detection in binary intrusion classification. Our investigation underscores the importance of effective feature selection and algorithmic robustness in enhancing clustering performance, reducing data dimensionality, and improving detection accuracy. Through theoretical analysis and empirical validation, we demonstrate significant advancements in clustering efficacy, contributing to more resilient and accurate network security frameworks.

2. Validity Measure

A. V-measure

V-measure is an entropy-based metric used to evaluate clustering quality by evaluating both homogeneity and completeness. Homogeneity measures how well a cluster contains members from a single class, while completeness measures how well all members of a class are assigned to the same cluster. V-measure combines these two aspects into a single score, providing a balanced evaluation of clustering performance. It helps in comparing different clustering algorithms or evaluating the effectiveness of a single method by integrating the degree to which clusters are pure and the degree to which items from the same class are clustered together [7,8].

$$V_\beta = \frac{(1+\beta) * H * C}{(\beta * H) + C}$$

$$H: homogeneity, \quad C: completness,$$

$$\beta: the\ ratio\ of\ precision\ and\ recall$$

Fig. 1. Calculation V-measure

As defined in the equation above, the parameter β in the equation adjusts the balance between precision and recall in the score. Precision is the ratio of true positives to predicted positives, while recall is the ratio of true positives to actual positives. This balance helps model the trade-off between false positives and false negatives, leading to a more accurate assessment of the detection method. When β exceeds 1, completeness is given more weight in the calculation, whereas if β is less than 1, homogeneity is prioritized. V-measure allows for comparisons across different clustering solutions regardless of class size, cluster size, data point size, or clustering algorithm.

B. Homogeneity

To meet our homogeneity criteria, a clustering algorithm must assign only data points from the same class to a single cluster. To calculate cluster homogeneity, let C represent the set of classes in the CICIDS2017 dataset, K the set of clusters, m the total number of elements, and a_{ck} the number of elements from class C assigned to cluster k . as illustrated in Figure 2. Note that homogeneity H(C|K) reaches its maximum and equals H(C) when the clustering provides no additional information and the class distribution within each cluster mirrors the overall class distribution.

$$H = \begin{cases} 1 & if\ H(C,K) = 0 \\ 1 - \dfrac{H(C|K)}{H(C)} & else \end{cases}$$

$$where$$

$$H(C|K) = -\sum_{k=1}^{|K|}\sum_{c=1}^{|C|}\frac{a_{ck}}{m}\log\frac{a_{ck}}{\sum_{c=1}^{|c|}a_{ck}}$$

$$H(C) = -\sum_{c=1}^{|C|}\frac{\sum_{k=1}^{|K|}a_{ck}}{m}\log\frac{\sum_{k=1}^{|K|}a_{ck}}{m}$$

Fig. 2. Calculation of Homogeneity in V-measure

C. Completeness

A clustering result meets the completeness criterion if all data points from a given class are grouped into a single cluster.

To calculate cluster completeness, let C represent the set of classes in the CICIDS2017 dataset, K the set of clusters, m the total number of elements, and a_{ck} the number of elements from class C assigned to cluster K, as illustrated in Figure 3.

D. Case Study 1: V-measure Calculation

To illustrate the concepts of homogeneity, completeness, and V-measure, let's consider a simple example with one feature set and two class labels (0 and 1):

Feature set: [1, 0, 1, 0] Class labels: [0, 1, 0, 1]

Using the calculations for homogeneity, completeness, and V-measure, we obtain the following scores:

Homogeneity = 1.0, Completeness = 1.0, V-measure = 1.0

$$C = \begin{cases} 1 & if\ H(K,C) = 0 \\ 1 - \dfrac{H(K|C)}{H(K)} & else \end{cases}$$

$$where$$

$$H(K|C) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|}\frac{a_{ck}}{m}\log\frac{a_{ck}}{\sum_{k=1}^{|K|}a_{ck}}$$

$$H(C) = -\sum_{k=1}^{|K|}\frac{\sum_{c=1}^{|C|}a_{ck}}{m}\log\frac{\sum_{c=1}^{|C|}a_{ck}}{m}$$

Fig. 3. Calculation of Completeness in V-measure

These scores indicate a perfect clustering result. The feature set satisfies homogeneity because each cluster contains data points from a single class, and it achieves perfect completeness because all members of each class are grouped together. The V-measure, as the harmonic mean of homogeneity and completeness, is 1.0, reflecting a completely accurate clustering.

E. Case Study 2: V-measure Calculation

Let's consider another example with a different feature set and class labels:

Feature set: [0, 1, 2, 3] Class labels: [0, 0, 1, 1]

Using the calculations for homogeneity, completeness, and V-measure, we obtain the following scores:

Homogeneity = 0.99, Completeness = 0.49, V-measure = 0.66

These scores indicate high homogeneity, but low completeness, suggesting that while each cluster predominantly contains data points from a single class, the same class is split across multiple clusters.

F. Case Study 3: V-measure Calculation

Consider the following example with a feature set and class labels:

Feature set: [0, 1, 0, 1] Class labels: [0, 0, 1, 1]

Calculating the homogeneity, completeness, and V-measure, we get the following scores:

Homogeneity = 0.0, Completeness = 0.0, V-measure = 0.0

These scores indicate that the data points are neither homogeneous nor complete, as the distribution of data points across different class labels is entirely mixed.

G. Clustering Evaluation

For our clustering technique, we utilize homogeneity and completeness, derived from v-measure [5]. The calculation of homogeneity and completeness is based on conditional entropy analysis. These metrics assess the clustering quality of traffic network data by forming data points based on related frequencies of network IP addresses, network ports, and other features. To select the suitable clustering, we implement the K-means clustering algorithm due to its simplicity, ease of interpreting results, and computational efficiency. K-means clusters each attribute separately, and then we use the v-measure to evaluate the clustering quality.

Cluster homogeneity requires that the purer networks traffic data in a clustering is, the more reliable the cluster is [7]. All data points in a data set, D belong to label classes K_1, K_2 for binary classification. Assume C_1 , a clustering where $C \in C_1$ includes data points from two classes, and C_2 a clustering that corresponds to C_1 except that C_2 is divided into two clusters that contains data points in K_1, K_2 respectively. Therefore, in terms of clustering quality measure Q , C_2 has a homogeneity score higher than C_1 as follows: $Q\left(C_2, C_g\right) > Q(C_1, C_g)$ (where Cg is a ground truth of clustering C, Q is measure score) and as shown in Figure 5 high homogeneity and low completeness.

Cluster completeness requires that the networks traffic data of a singular cluster belong to the same class [7]. Consider clustering C_1, which contains clusters C_1 and C_2 where the members of clustering belong to the same class as indicated by K_1, K_2 . Assume that clustering C_2 is identical to C_1 except that C_1 and C_2 are integrated into one cluster in C_2 . Therefore, in terms of clustering quality measure Q, C_2 has a completeness score higher than C_1 as follows: $Q\left(C_2, C_g\right) > Q(C_1, C_g)$ where (Cg is a ground truth of clustering C, Q is measure score).

Therefore, clustering is perfectly homogeneous when all data points within each cluster belong to the same class label (benign or attack). However, it is not complete if not all data points of a given class label are within the same cluster. Figure 4 illustrates an extreme example of cluster boundaries that yield varying levels of completeness and homogeneity [8].

3. Methodology

Before To demonstrate the effectiveness of using V-measure for clustering evaluation, we conducted three experiments.

In Experiment A, we pre-processed and transformed raw data into an understandable format, then applied three classifiers to all training and testing datasets, followed by V-measure. If the V-measure score was greater than 1, completeness was weighted more strongly for classification purposes. If the V-measure score was less than 1, homogeneity was weighted more strongly in the classification process. The same number of features selected by V-measure was used for F-measure evaluation.

In Experiment B, we applied F-measure to all training and testing datasets and evaluated the outcomes using Decision Tree, Random Forest, and AdaBoost algorithms. We compared V-measure and F-measure based on the confusion matrices produced.

In Experiment C, we implemented the three machine learning algorithms using the identical features selected by both V-measure and F-measure, ensuring a consistent basis for comparison.

Three supervised learning classifiers are implemented to evaluate the clustering metrics tasks: Decision Tree (DT), Random Forest, and Adaptive Boosting (AdaBoost). DT is efficient for large datasets and analyzes key factors indicating abnormal activities [9]. Random Forest, an ensemble of decision trees, achieves accurate predictions by combining multiple trees [10]. AdaBoost excels in pattern recognition and binary classification, particularly in intrusion detection, due to its speed, low computational complexity, and ability to update training patterns during classification [11].

A. Experiment Criteria

Three experiments were conducted to evaluate V-measure by comparing it with F-measure. The performance of the selected features was assessed using three machine learning algorithms. The experiments were carried out using Anaconda Python Distribution 3.7, Jupyter notebook, and Scikit-learn, and deployed on Amazon SageMaker Studio [7]. Amazon SageMaker, a cloud machine-learning platform, supports elastic learning and incremental training, with Amazon EC2 providing configurable, memory-optimized instances (ml.r5.2xlarge) featuring 8 virtual CPUs and 64 GB memory to ensure fast performance for pre-processing, feature selection, and classifier application, thereby reducing computational costs in terms of memory and execution time.

B. Data Preprocessing

To compute V-measure and F-measure, each feature in the CICIDS-2017 dataset was processed based on its class label (benign or attack), resulting in two clusters [12]. Table I lists all the features. The dataset, containing 2,830,743 records, was pre-processed using Python scripts with Pandas and Numpy. Class labels for benign and attack were replaced with 0 and 1, respectively. After removing records with missing or infinity values, 2,827,876 records remained. The dataset was then randomly split into 70% for training and 30% for testing. This partitioning approach, used in other empirical studies, has shown high accuracy rates [13].

Table I. Features Numbers and Names

F-ID	Feature Name	F-ID	Feature Name
1	Destination Port	41	Packet Length Mean
2	Flow Duration	42	Packet Length Std
3	Total Fwd Packets	43	Packet Length Variance
4	Total Backward Packets	44	FIN Flag Count
5	Total Length of Fwd Packs	45	SYN Flag Count
6	Total Length of Bwd Packs	46	RST Flag Count
7	Fwd Packet Length Max	47	PSH Flag Count
8	Fwd Packet Length Min	48	ACK Flag Count
9	Fwd Packet Length Mean	49	URG Flag Count
10	Fwd Packet Length Std	50	CWE Flag Count
11	Bwd Packet Length Max	51	ECE Flag Count
12	Bwd Packet Length Min	52	Down/Up Ratio
13	Bwd Packet Length Mean	53	Average Packet Size
14	Bwd Packet Length Std	54	Fwd Segment Size Avg
15	Flow Bytes/s	55	Bwd Segment Size Avg
16	Flow Packets/s	56	Fwd Header Length
17	Flow IAT Mean	57	Fwd Avg Bytes/Bulk
18	Flow IAT Std	58	Fwd Avg Packets/Bulk
19	Flow IAT Max	59	Fwd Avg Bulk Rate
20	Flow IAT Min	60	Bwd Avg Bytes/Bulk
21	Fwd IAT Total	61	Bwd Avg Packets/Bulk
22	Fwd IAT Mean	62	Bwd Avg Bulk Rate
23	Fwd IAT Std	63	Sub flow Fwd Packets
24	Fwd IAT Max	64	Sub flow Fwd Bytes
25	Fwd IAT Min	65	Sub flow Bwd Packets
26	Bwd IAT Total	66	Sub flow Bwd Bytes
27	Bwd IAT Mean	67	Fwd Init Win bytes
28	Bwd IAT Std	68	Bwd Init Win bytes
29	Bwd IAT Max	69	Fwd Act Data Pkts
30	Bwd IAT Min	70	Fwd Seg Size Min
31	Fwd PSH Flags	71	Active Mean
32	Bwd PSH Flags	72	Active Std
33	Fwd URG Flags	73	Active Max
34	Bwd URG Flags	74	Active Min
35	Fwd Header Length	75	Idle Mean
36	Bwd Header Length	76	Idle Std
37	Fwd Packets/s	77	Idle Max
38	Bwd Packets/s	78	Idle Min
39	Min Packet Length	79	Label
40	Max Packet Length		

C. Evaluation Criteria

The results were presented using standard binary classification metrics for the two classes, Benign and Attack: true positive (TP), false positive (FP), true negative (TN), and false negative (FN) [14]. From these metrics, we calculated the classification accuracy, detection rate, false positive rate, and false negative rate.

- Accuracy: the ratio of correctly classified samples to the total number of samples.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN}$$

- Detection Rate (DT): the ratio between total numbers of detected attacks to the total number of attacks in the dataset.

$$\text{DT} = \frac{TP}{TP+TN}$$

- False Positive Rate (FPR): the number of non-intrusions inaccurately detected; false positive rate is defined as:

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

- False Negative Rate (FNR): the number of intrusions inaccurately detected; false negative rate is defined as:

$$\text{FNR} = \frac{FN}{FN+TN}$$

D. Results

After preprocessing the dataset, we applied both V-measure and F-measure to select the best 44 features for analysis. We then ran three classifiers Decision Tree, Random Forest, and AdaBoost against the training dataset to evaluate these features using standard classification metrics. This approach allowed us to compare the performance of the classifiers based on the selected features, providing insights into the effectiveness of V-measure and F-measure in identifying the most

Table I. Features Numbers and Names

relevant features for accurate classification.

In evaluating clustering performance, V-measure has demonstrated notable progress over traditional metrics like F-measure. V-measure, which balances homogeneity and completeness, provides a more nuanced understanding of clustering quality by ensuring that clusters are both internally cohesive and externally distinct. In the experiments conducted, V-measure consistently produced higher accuracy rates across different algorithms: Decision Tree (99.8%), Random Forest (99.9%), and AdaBoost (99.9%), compared to F-measure's accuracy rates of 98.7%, 98.8%, and 96.7% respectively. This indicates that V-measure more effectively captures the underlying structure of the data, leading to better classification performance.

Table II. Features Numbers and Names

SM	V-measure			F-measure		
	DT	RF	AdaBoost	DT	RF	AdaBoost
AC	99.8%	99.9%	99.9%	98.7%	98.8	96.7%
DR	99.6%	99.7%	97.8%	98.6%	98.6	86.4%
FPR	0.0008	0.0006	0.007	0.011	0.011	0.007
FNR	0.0008	0.0005	0.005	0.003	0.003	0.032

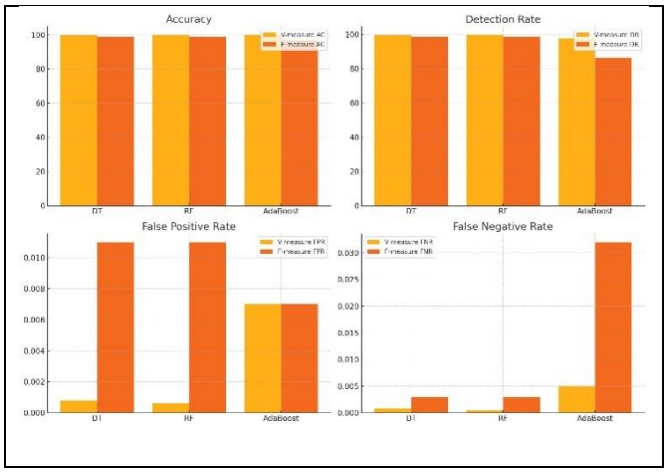

Fig. 5. Cluster Outlines of Completeness and Homogeneity Values

V-measure outperformed F-measure in terms of detection rate and false positive rate, crucial metrics for evaluating the effectiveness of clustering in identifying the correct class labels. For instance, the detection rates for V-measure were 99.6% (DT), 99.7% (RF), and 97.8% (AdaBoost), significantly higher than those for F-measure, which were 98.6% (DT), 98.6% (RF), and 86.4% (AdaBoost). Additionally, V-measure achieved lower false positive rates and false negative rates across the board. The false positive rates for V-measure were 0.0008 (DT), 0.0006 (RF), and 0.007 (AdaBoost), compared to F-measure's 0.011 (DT and RF) and 0.007 (AdaBoost). The false negative rates also favored V-measure, highlighting its superior capability in reducing both types of errors. These results underscore V-measure's effectiveness and reliability in clustering evaluation, making it a preferred choice over F-measure in scenarios requiring precise and accurate classification.

An analysis of Figure 5 illustrates a surprisingly high accuracy and detection rate, coupled with a low false positive rate of 0.0006 and a false negative rate of 0.0005. This demonstrates that V-measure provides better clustering quality compared to F-measure, resulting in more precise and reliable classification performance.

E. Algorithm: V-measured-Based-IDS

Input: Feature set $\boldsymbol{F} = \{\boldsymbol{f_1}, \boldsymbol{f_2}, \dots, \boldsymbol{f_i} \dots, \boldsymbol{f_n}\}$
where $\mathbf{1} \leq \boldsymbol{i} \leq \boldsymbol{n}$, $\boldsymbol{C}$, $\boldsymbol{K}$, dataset $\boldsymbol{D}$ where $|\boldsymbol{D}| = \boldsymbol{m}$

Output: V-measure of all features in
$\boldsymbol{F}$: $\boldsymbol{V(F)} = \{\boldsymbol{V(f_1)}, \boldsymbol{V(f_2)}, \dots, \boldsymbol{V(f_i)} \dots, \boldsymbol{V(f_n)}\}$
where $\mathbf{1} \leq \boldsymbol{i} \leq \boldsymbol{n}$

1: For all ($\mathbf{1} \leq \boldsymbol{i} \leq \boldsymbol{n}$)
2: Perform clustering of items in $\boldsymbol{D}$ into $\boldsymbol{K}$ clusters based on feature $\boldsymbol{f_i}$
//Calculate homogeneity
3: Calculate $\boldsymbol{H(C)} = -\sum_{c=1}^{|C|} \frac{\sum_{k=1}^{|K|} a_{ck}}{m} \log \frac{\sum_{k=1}^{|K|} a_{ck}}{m}$
4: Calculate $\boldsymbol{H(C|K)} = -\sum_{k=1}^{|K|} \sum_{c=1}^{|C|} \frac{a_{ck}}{m} \log \frac{a_{ck}}{\sum_{c=1}^{|C|} a_{ck}}$
5: If $\boldsymbol{H(C,K)} = \mathbf{0}$ then $\boldsymbol{H} = \mathbf{1}$
6: else $\boldsymbol{H} = \mathbf{1} - \frac{\boldsymbol{H(C|K)}}{\boldsymbol{H(C)}}$
//Calculate completeness
7: Calculate $\boldsymbol{H(K)} = -\sum_{k=1}^{|K|} \frac{\sum_{c=1}^{|C|} a_{ck}}{m} \log \frac{\sum_{c=1}^{|C|} a_{ck}}{m}$
8: Calculate $\boldsymbol{H(K|C)} = -\sum_{c=1}^{|C|} \sum_{k=1}^{|K|} \frac{a_{ck}}{m} \log \frac{a_{ck}}{\sum_{k=1}^{|K|} a_{ck}}$
9: If $\boldsymbol{H(K,C)} = \mathbf{0}$ then $\boldsymbol{K} = \mathbf{1}$
10: else $\boldsymbol{K} = \mathbf{1} - \frac{\boldsymbol{H(K|C)}}{\boldsymbol{H(K)}}$
//Calculate $\boldsymbol{\beta}$
11: Calculate True Positive ($\boldsymbol{TP}$), True Negative ($\boldsymbol{TN}$), False Positive ($\boldsymbol{FP}$), False Negative ($\boldsymbol{FN}$)
12: Calculate $\mathbf{precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}}$, $\mathbf{recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}}$
13: Calculate $\boldsymbol{\beta} = \frac{\boldsymbol{precision}}{\boldsymbol{recall}}$
//Calculate v-measure
14: Calculate $\boldsymbol{V(f_i)} = \frac{(\mathbf{1}+\boldsymbol{\beta}) * \boldsymbol{H} * \boldsymbol{C}}{(\boldsymbol{\beta} * \boldsymbol{H}) + \boldsymbol{C}}$
15: return $\boldsymbol{V(F)} = \{\boldsymbol{V(f_1)}, \boldsymbol{V(f_2)}, \dots, \boldsymbol{V(f_i)} \dots, \boldsymbol{V(f_n)}\}$

4. Conclusion

The application of V-measure as an evaluation metric has proven to be highly effective in assessing clustering quality, particularly in the context of network intrusion detection systems. By balancing the critical aspects of homogeneity and completeness, V-measure provides a nuanced and robust assessment framework that enhances the accuracy and reliability of clustering algorithms. Our experiments demonstrate that V-measure not only outperforms traditional metrics like F-measure but also significantly improves detection rates while minimizing false positive and negative rates. These findings suggest that V-measure is a superior choice for feature selection and clustering evaluation in binary intrusion classification, contributing to more secure and efficient network security systems.

5. References

[1] D. Greene, P. Cunningham, and R. Mayer, "Unsupervised learning and clustering," in Machine Learning Techniques for Multimedia: Case Studies on Organization and Retrieval, pp. 51-90, 2008.

[2] S. Chander and P. Vijaya, "Unsupervised learning methods for data clustering," in Artificial Intelligence in Data Mining, pp. 41-64, 2021. Academic Press.

[3] R. Xu and D. Wunsch, "Survey of clustering algorithms," IEEE Transactions on Neural Networks, vol. 16, no. 3, pp. 645-678, May 2005.

[4] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," Trans. Emerg. Telecommun. Technol., vol. 32, no. 1, p. e4150, Jan. 2021.

[5] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," Proc. 2007 Joint Conf. Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 410-420, Jun. 2007.

[6] B. E. Dom, "An information-theoretic external cluster-validity measure," arXiv preprint arXiv:1301.0565, 2012.

[7] E. Hmouda, "A Validity-based Approach for Feature Selection in Systems," Ph.D. dissertation, Nova Southeastern Univ., 2022.

[8] E. Hmouda and W. Li, "Validity Based Approach for Feature Selection in Intrusion Detection Systems," SoutheastCon, pp. 1-8, Mar. 2020. IEEE.

[9] Y. Song and L. U. Ying, "Decision tree methods: applications for classification and prediction," Shanghai Arch. Psychiatry, vol. 27, no. 2, p. 130, 2015.

[10] A. Parmar, R. Katariya, and V. Patel, "A review on random forest: An ensemble classifier," in Int. Conf. Intelligent Data Communication Technologies and Internet of Things (ICICI), pp. 758-763, 2019. Springer International Publishing.

[11] T. An and M.-H. Kim, "A new diverse AdaBoost classifier," in 2010 Int. Conf. Artificial Intelligence and Computational Intelligence, vol. 1, pp. 359-363, 2010. IEEE.

[12] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," ICISSp, vol. 1, pp. 108-116, 2018.

[13] A. Gholamy, V. Kreinovich, and O. Kosheleva, "Why 70/30 or 80/20 relation between training and testing sets: A pedagogical explanation," Int. J. Intell. Technol. Appl. Stat., vol. 11, no. 2, pp. 105-111, 2018.

[14] S. Marsland, Machine Learning: An Algorithmic Perspective. Chapman and Hall/CRC, 2011.