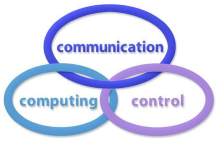
INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 21, Issue: 3, Month: June, Year: 2026
Article Number: 7305, https://doi.org/10.15837/ijccc.2026.3.7305



CCC Publications

Cosine Distance-Based Fuzzy C-Means Clustering for Local Classification in Imbalanced Network Intrusion Detection

Ngoc-Bich Giap Thi, Van-Nhan Nguyen, Anh-Thu Pham, Trong-Minh Hoang

Ngoc-Bich Giap Thi, Anh-Thu Pham, and Trong-Minh Hoang*
Telecommunication Faculty No1
Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
bichgtn.b24chkv002@stu.ptit.edu.vn, thupa@ptit.edu.vn and hoangtrongminh@ptit.edu.vn
*Corresponding author: hoangtrongminh@ptit.edu.vn

Van-Nhan Nguyen
Faculty of Information Technology
Dai Nam University, Hanoi, Vietnam
nhannv@dainam.edu.vn

Abstract

Network Intrusion Detection Systems (NIDS) deal with class imbalance in network traffic data, where minority attack classes are underestimated. FCM-Cosine, a modified Fuzzy C-Means clustering algorithm, replaces Euclidean distance in the objective function with Cosine distance to better capture directional similarity in high-dimensional feature spaces. The cluster-then-classify framework decomposes the global intrusion detection problem into localized classification sub-problems to detect minority attack classes. Five classifiers have been examined on the CICIoT2023 dataset at two scales (16,100 and 465,000 samples). FCM-Cosine had an average F1-Macro of 69.36%, while Decision Tree had 86.79%, resulting in a 37.97% improvement over direct training. The framework is ten times faster than SMOTE (19.18s vs. 189.73s average training time) and scales nearly linearly with dataset size. Results demonstrate that FCM-Cosine offers competitive classification performance with computational efficiency for large-scale NIDS deployments.

Keywords: Network intrusion detection, fuzzy C-Means clustering, cosine distance, class imbalance, machine learning.

1 Introduction

In recent years, cyberattacks have increased rapidly in both frequency and complexity. Common attack types such as DDoS, ransomware, phishing, and targeted attacks (APT) continue to expand in scale and practically affect enterprise systems, cloud computing platforms, and critical infrastructures [1, 2]. This complexity issue highlights the need for proactive detection mechanisms to face an increasingly complex attack environment.

Network Intrusion Detection Systems (NIDS) play a particularly important role in detecting abnormal behaviors due to their flexibility and early detection capability of potential attacks by monitoring network traffic before it reaches victim devices [3]. Traditional knowledge-based approaches nowadays have suffered from performance degradation and increased false alarms due to the need for continuous

updates or expert knowledge. Otherwise, IDS-based machine learning provides a promising alternative by automating the detection process and adapting to new attacks with minimal human action. Hence, this approach has been widely considered by studies on machine learning methodologies to improve NIDS performance [4, 5]. Besides advantages, remaining challenges have existed in these approaches (traditional ML and deep learning), such as the trade-off among accuracy, interpretability, and complexity [6, 7].

Regarding the abnormal attack detection tasks, despite significant advances in ML-based NIDS, class imbalance in the dataset remains a critical challenge that affects IDS' detection performance [8]. To handle class imbalance, several approaches have been developed to mitigate this issue, focusing on data-level, algorithm-level, and hybrid strategies [9]. Besides their significant results in handling data imbalance in IDS, the complexity and variability of real-world network traffic data require continuous improvements. To face uncertainty attack behavior, clustering methods based on a fuzzy approach have become a pivotal tool to restructure data distribution for identifying normal and abnormal activities [10, 8].

However, the effectiveness of fuzzy clustering methods depends heavily on the quality of the input data and the choice of distance metric. Most existing FCM-based approaches utilised Euclidean distance, but it is sensitive to feature magnitude and scale, making it less suitable for high-dimensional network data. Meanwhile, direction-based similarity measures, such as Cosine distance, have shown promise in capturing relationships among network flows in high-dimensional feature spaces [11]. However, their systematic integration into fuzzy clustering for intrusion detection remains unexplored. To fill the limitation gap, this study makes the following contributions.

- We propose a novel FCM-Cosine, a modified Fuzzy C-Means algorithm that replaces Euclidean distance with cosine distance in the objective function. Our framework can capture directional similarity among feature vectors and is appropriate for high-dimensional network traffic data.
- We introduce a two-phase framework that decomposes the global intrusion detection problem into multiple local classification sub-problems for improving detection performance for minority attack classes.
- We conduct extensive experiments on the CICIoT2023 benchmark dataset across two scales (16,100 and 465,000 samples) using five representative classifiers and four data configuration strategies to determine which method is most suitable for large-scale deployment.

The organisation of this paper is as follows. Section 2 reviews related work on intrusion detection, class imbalance handling, and clustering-based approaches. Section 3 presents the proposed FCM-Cosine framework. Section 4 describes the experimental setup and scenarios. Section 5 presents experimental results. Last but not least, Section 6 concludes the paper and discusses future research directions.

2 Related Work

Research on network intrusion detection is commonly categorized into signature-based, anomaly-based, and hybrid approaches. Signature-based systems achieve high accuracy for known attacks but fail against zero-day attacks, whereas anomaly-based systems detect unseen attacks but suffer from high false alarm rates. Hybrid models combining both approaches have attracted increasing attention for improving NIDS flexibility and reliability [12, 13, 14]. To enhance the performance of IDS-based machine learning, several imbalance processing techniques have been studied, including such as SMOTE and ADASYN to augment minority classes [8, 15]. However, these methods are prone to overfitting or noise generation [8, 11]. To mitigate these issues, recent studies have shifted toward reducing redundancy in the majority class [11]. The Cosine Similarity-based Majority Class Reduction (CSMCR) technique has demonstrated effectiveness in removing duplicate majority samples based on feature-wise similarity analysis, thereby preserving data diversity without generating synthetic samples.

In addition, integrating fuzzy logic into IDS enables systems to handle uncertainty and vagueness in network traffic. Unlike hard clustering methods such as K-means, which assign each data point to a single cluster, fuzzy clustering allows data points to belong to multiple clusters with varying degrees of

membership to cope with complex nonlinear data structures [16]. This property is particularly useful for high-dimensional data, where the boundaries between normal behavior and intrusion frequently overlap. By assigning fuzzy memberships, models can capture the distinctive characteristics of each traffic segment and perform local linearization to improve classification accuracy. This approach reduces computational complexity compared to training a purely deep learning model on the entire raw dataset, which is typically noisy and severely imbalanced [11, 17]. However, these models generally require substantial computational resources and high-quality training data to achieve satisfactory performance of minority classes [15, 18]. Moreover, zero-day attacks and rare attack types often have very limited representative samples, making them easy to neglect by global learning models [8].

Clustering techniques have been utilized to enhance intrusion detection by categorizing traffic samples exhibiting similar behaviors. The K-means technique is popular for its simplicity and ease of implementation; however, its dependence on Euclidean distance and hard clustering restricts its effectiveness in complex data environments [8]. Additionally, FCM allows each sample to belong to multiple clusters with varying degrees of membership, thereby more accurately representing the uncertainty present in network data. This method has been utilized in several recent studies on IDSs [17]. Most current FCM-based approaches continue to utilize Euclidean distance, which is affected by feature magnitude and scale, making it unsuitable for high-dimensional network data.

Otherwise, direction-based similarity measures, including cosine distance, can more accurately represent relationships among network flows in high dimensional feature spaces [11]. The systematic integration of Cosine distance into fuzzy clustering for intrusion detection and imbalance handling remains unexplored. While Cosine distance has been explored in data balancing [11], its integration into FCM for local classification remains underexplored. This paper integrates Fuzzy C-Means with cosine distance to directly address the class imbalance problem in NIDS, supported by a clear theoretical foundation and validated through a comprehensive experimental evaluation.

3 Proposed Framework

To address severe class imbalance and high-dimensional feature spaces in NIDS, we propose a hybrid *cluster-then-classify* framework. The core idea is to partition network traffic into structurally coherent subgroups using Fuzzy C-Means (FCM) clustering with cosine distance, then perform classification within these localized regions. By replacing Euclidean distance with Cosine distance in the FCM objective function, the approach better captures directional similarity in high-dimensional network data. This clustering step decomposes the complex global decision boundary into multiple simpler, locally linearizable sub-problems, thereby improving class separability and detection performance under imbalanced conditions.

3.1 Framework Overview

Figure 1 illustrates the proposed NIDS architecture, designed to simultaneously address severe class imbalance, high-dimensional feature spaces, and complex nonlinear decision boundaries in network traffic data. Instead of learning a single global decision boundary, this framework decomposes the intrusion detection problem into multiple local sub-problems that are simpler and more tractable.

The framework operates in three phases: (i) data preprocessing to normalize features and encode labels, (ii) fuzzy clustering to partition the feature space into homogeneous regions, and (iii) local classification within each cluster to produce final classification. Preprocessing is performed on the input data to ensure it is suitable for subsequent learning phases. Label encoding converts categorical labels so that machine learning algorithms can process them efficiently. Data standardization is used to normalize feature values in order to guarantee a common scale, which is especially crucial for distance measures based on cosine. To guarantee an unbiased assessment, the dataset is then divided into training and testing subsets.

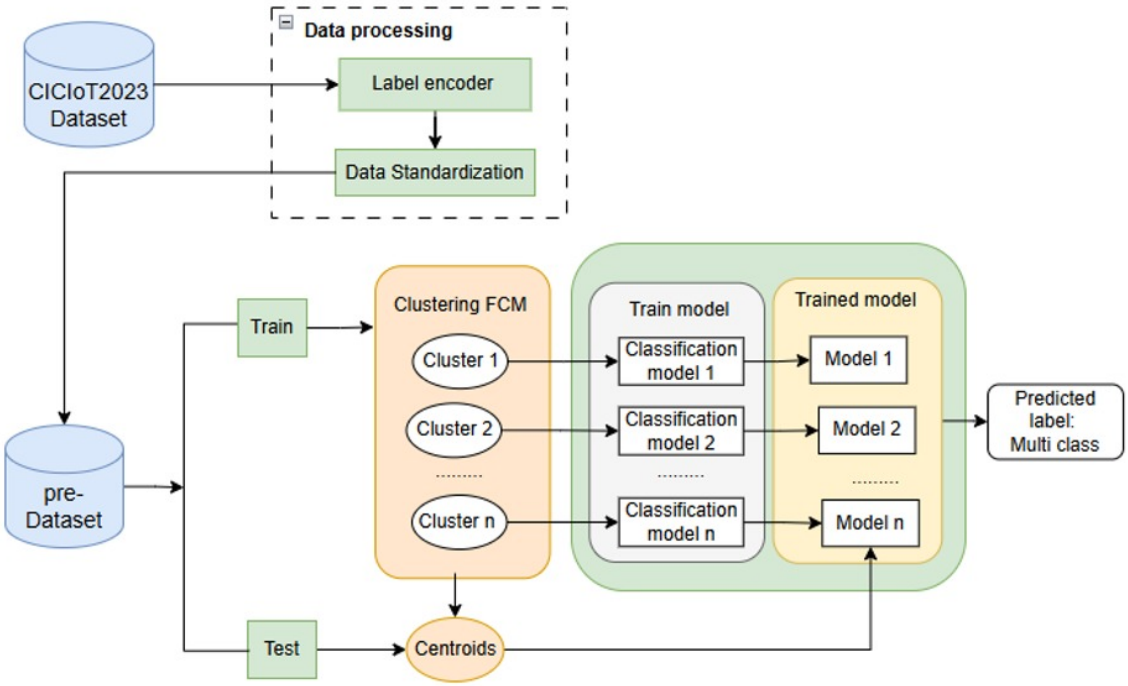


Figure 1: Overview of the proposed cluster-then-classify framework for NIDS.

3.2 Fuzzy C-Means with Cosine Distance

Traditional clustering algorithms based on Euclidean distance can be highly sensitive to noise and suffer from performance degradation in high-dimensional feature spaces. The proposed FCM-based cosine distance is directly incorporated into the objective function can overcome these limitations.

Problem formulation and notation Let $X = \{x_1, x_2, \ldots, x_N\}$ denote the network traffic dataset, where N is the number of samples and each $x_i \in \mathbb{R}^d$ represents a d-dimensional feature vector. Let $Y = \{y_1, y_2, \ldots, y_N\}$ denote the class labels. The objective is to partition X into K fuzzy clusters and train local classifiers within each cluster to improve detection performance for minority attack classes. Key notations are below.

- K: number of clusters
- c_j: centroid of cluster j, where $j \in \{1, 2, \ldots, K\}$
- u_{ij}: fuzzy membership degree of sample x_i to cluster j
- m: fuzziness exponent, typically set to $m = 2$
- $d_{\cos}(\cdot, \cdot)$: Cosine distance function

Objective Function

The objective function J_m of FCM-Cosine is defined as

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{K} u_{ij}^m \, d_{\cos}(x_i, c_j), \tag{1}$$

where the Cosine distance between sample x_i and centroid c_j is

$$d_{\cos}(x_i, c_j) = 1 - \frac{x_i \cdot c_j}{|x_i| \, |c_j|}. \tag{2}$$

Update Rules

The membership matrix $U = [u_{ij}]$ and cluster centers $C = c_j$ are iteratively updated to minimize J_m. The membership degree u_{ij} is updated as

$$u_{ij} = \frac{1}{\sum_{k=1}^{K} \left(\frac{d_{\cos}(x_i, c_j)}{d_{\cos}(x_i, c_k)} \right)^{\frac{1}{m-1}}}. \tag{3}$$

Cluster centroids are updated by

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m \, x_i}{\sum_{i=1}^{N} u_{ij}^m}. \tag{4}$$

Convergence Criteria

The algorithm iterates until convergence, defined as $\max_{i,j} |u_{ij}^{(t+1)} - u_{ij}^{(t)}| < \epsilon$, where ϵ is a predefined threshold (typically $\epsilon = 10^{-4}$), or until a maximum number of iterations is reached.

3.3 Cluster-Based Local Classification

The clustering output consists of fuzzy clusters representing homogeneous network behavior regions and their centroids. Each cluster forms a local subspace where data distribution is more balanced and class boundaries simpler than in the global dataset.

Within each local subspace, an independent classifier is trained. This reduces bias from global class imbalance, as each classifier learns from a more homogeneous subset. The classification task complexity decreases because each classifier learns only local decision boundaries, substantially improving detection of rare attack types often overshadowed in global models.

During testing, each sample is assigned to the nearest cluster based on Cosine distance to FCM centroids, then forwarded to the corresponding classifier for prediction. The final output is a multi-class label reflecting the traffic or attack pattern type.

Algorithm 1 summarizes the complete training and inference procedures.

Algorithm 1 FCM-Cosine based NIDS Framework

Require: Training dataset X, Labels Y, Number of clusters K, Fuzziness m, Threshold ϵ
Ensure: Set of local classifiers $M_1, \ldots, M_K$, Cluster centers C

Phase 1: Soft Clustering
Initialize cluster centers C randomly
repeat
    Update membership matrix U using Eq. (3)
    Update cluster centers C using Eq. (4)
until $\max_{i,j} |u_{ij}^{(t+1)} - u_{ij}^{(t)}| < \epsilon$
Phase 2: Local Training
for $j = 1$ to K **do**
    $X_j \leftarrow x_i \mid \arg\max_k(u_{ik}) = j$
    $Y_j \leftarrow y_i \mid x_i \in X_j$
    $M_j \leftarrow \text{Train}(X_j, Y_j)$
end for
Phase 3: Inference
for each new sample x_{new} **do**
    Compute $d_{\cos}(x_{\text{new}}, c_j)$ for all $j \in 1, \ldots, K$
    $k \leftarrow \arg\min_j d_{\cos}(x_{\text{new}}, c_j)$
    $y_{\text{pred}} \leftarrow M_k.\text{predict}(x_{\text{new}})$
end for
    return $M_1, \ldots, M_K, C$

The computational complexity of the proposed framework consists of three components. The clustering phase requires $O(N \cdot K \cdot d \cdot T)$ operations, where T is the number of iterations until con-

vergence. The local training phase complexity depends on the classifier type; for Decision Trees, it is $O(\sum_{j=1}^{K} N_j \cdot d \cdot \log N_j)$, where N_j is the number of samples in cluster j. The inference phase requires $O(K \cdot d)$ operations per sample for cluster assignment plus the prediction cost of the local classifier. Overall, the framework maintains linear scalability with respect to dataset size N.

4 Experimental Setup

4.1 Dataset Description

To comprehensively evaluate the effectiveness of the proposed framework, this study employs the CICIoT2023 dataset released by the Canadian Institute for Cybersecurity. This dataset is among the most recent and comprehensive benchmarks for network intrusion detection, constructed in a modern network environment and closely reflecting real-world attack scenarios.

The CICIoT2023 dataset contains both benign network traffic and a wide range of common attack types, including DoS/DDoS, brute-force attacks, web attacks, and other advanced intrusion techniques. The data are extracted in the form of flow-based features, covering packet statistics, temporal characteristics, packet sizes, and connection behaviors. A key characteristic of CICIoT2023 is its severe class imbalance, where attack traffic constitutes only a small fraction compared to normal traffic. The experimental design follows a nested two-scale evaluation case to assess robustness and scalability.

- **Small-Scale Subset (16k samples).** A stratified sampling strategy is employed to select a subset of 16,100 samples (approximately 1% of the full dataset). This subset serves to validate the initial hypothesis and tune key hyperparameters (e.g., the number of clusters K). Following the methodology recommended by Vabalas et al. [19], small-scale experiments are essential for initial feasibility assessment and hyperparameter optimization before committing computational resources to large-scale validation.

- **Large-Scale Superset (465k samples).** The experiments are expanded to 465,000 samples, approximately 29 times larger than the small-scale subset. This expansion directly addresses the findings of Vabalas et al. [19], who demonstrated that small training sets can lead to optimistic performance estimates and reduced statistical power in machine learning models. By scaling up to nearly half a million samples, we ensure sufficient representation of minority attack classes and validate that the observed performance improvements are not artifacts of limited data but reflect genuine model capabilities. This nested evaluation enables systematic assessment of model robustness and scalability when dataset size increases substantially. This scale is consistent with recent large-scale intrusion detection studies [8, 11] and provides robust evidence of real-world applicability.

4.2 Evaluation Metrics

This study employs a set of standard evaluation metrics: Accuracy, Precision (Weighted), Recall (Weighted), F1-Weighted, and F1-Macro. Accuracy measures the proportion of correct predictions across the entire dataset. Precision (Weighted) and Recall (Weighted) are calculated taking into account the weight of each class, reflecting the weighted average performance.

F1-Weighted combines Precision and Recall according to class weight, balancing these factors at an overall level. F1-Macro is particularly emphasized because it calculates the average F1-score of all classes with equal weight, regardless of sample count. F1-Macro more accurately reflects the model's ability to detect rare attack classes, which is the core objective of NIDS. A high F1-Macro value indicates that the model maintains robust performance for infrequently occurring attack types.

4.3 Data Configuration Strategies and Classifiers

To comprehensively evaluate the effectiveness of the proposed framework, a comparative analysis is conducted using four data handling strategies in combination with five representative classifiers. The main strategies include (i) direct training on the original imbalanced dataset, (ii) SMOTE (Synthetic

Minority Over-sampling Technique) to balance the training set through synthetic sample generation, (iii) K-Means clustering with Euclidean distance representing hard clustering approaches, and (iv) the proposed FCM-Cosine framework using Fuzzy C-Means with Cosine distance for soft clustering.

The classification models span three paradigms: single estimators (Decision Tree), ensemble methods (Random Forest and Extra Trees), and deep learning architectures (Deep Neural Network and Long Short-Term Memory). Decision Tree provides a simple baseline with high interpretability. Random Forest and Extra Trees represent ensemble approaches that can reduce variance through the aggregation of multiple decision trees. DNN is suitable for learning nonlinear representations from high-dimensional static features, while LSTM is designed to capture temporal sequential dependencies in network traffic data. This combination of diverse strategies and classifiers enables systematic assessment of the proposed framework's robustness across different learning paradigms and data processing approaches.

5 Experimental Results and Discussions

All experiments were conducted using Python 3.8 with scikit-learn 1.0.2, TensorFlow 2.8.0, and scikit-fuzzy 0.4.2 libraries. The fuzziness exponent for FCM was set to $m = 2$, and the convergence threshold was $\epsilon = 10^{-4}$. For DNN, a three-layer architecture with 128, 64, and 32 neurons was used, trained with Adam optimizer (learning rate = 0.001) for 50 epochs. LSTM employed a single layer with 64 units. Random Forest and Extra Trees used 100 estimators with a maximum depth of 20. Decision Tree used default scikit-learn parameters with the Gini impurity criterion.

Table 1 presents a comprehensive performance comparison across five classifiers and four data handling strategies on the large-scale superset (465,000 samples). We analyze results from two perspectives: classification effectiveness (Table 1) and computational efficiency (Table 1).

Classification Performance. Clustering-based methods demonstrate substantial advantages over baseline approaches (Table 1). K-Means achieves the highest average F1-Macro (70.47%), followed closely by FCM-Cosine (69.36%), both significantly outperforming SMOTE (65.96%) and Direct (62.52%). The improvement is particularly pronounced for simple classifiers: Decision Tree with FCM-Cosine achieves 86.79% F1-Macro, representing a 37.97% absolute gain over the Direct baseline (48.79%). This validates our hypothesis that the cluster-then-classify framework enables local specialization, allowing simple models to learn distinct decision boundaries within homogeneous subspaces.

However, the advantage diminishes for complex models. Random Forest with SMOTE achieves the highest F1-Macro (79.24%), outperforming FCM-Cosine (75.04%) by 4.20%. This suggests that ensemble methods' inherent capability to handle class imbalance through bootstrap aggregation reduces the relative benefit of explicit clustering. Notably, LSTM shows inconsistent behavior: SMOTE achieves 57.35% while FCM-Cosine drops to 46.13%, indicating that sequential models may be sensitive to cluster-induced data fragmentation that disrupts temporal dependencies.

Computational Efficiency. K-Means demonstrates exceptional training speed (Table 1), averaging 1.80 seconds across all classifiers due to its one-time clustering cost followed by parallel local training. FCM-Cosine ranks second (19.18s average), offering a favourable trade-off between performance and efficiency. In stark contrast, SMOTE exhibits severe scalability issues, averaging 189.73 seconds, approximately 105 times slower than K-Means and 10 times slower than FCM-Cosine. The bottleneck is most severe for deep learning models: DNN with SMOTE requires 511.01 seconds, compared to 10.49 seconds for FCM-Cosine (48.7 times speedup) and 1.49 seconds for K-Means (343 times speedup). This super-linear time growth stems from SMOTE's nearest-neighbour search in high-dimensional space, which becomes prohibitively expensive at large scales.

These results confirm that FCM-Cosine achieves competitive classification performance (2nd-best average F1-Macro at 69.36%) while maintaining computational efficiency (2nd-fastest average time at 19.18s), making it a practical choice for large-scale network intrusion detection systems. The slight performance gap compared to K-Means (1.11% F1-Macro difference) is offset by FCM-Cosine's interpretable soft membership assignments, which provide probabilistic cluster affiliations useful for anomaly detection and model debugging.

Table 1: Performance Comparison on Large-Scale Superset (465,000 samples)

(a) F1-Macro Score (%)

Model (Category)	Direct	SMOTE	K-Means	FCM-Cosine
Decision Tree (Single Est.)	48.79	56.39	85.95	**86.79**
DNN (Deep Learning)	66.34	62.93	**66.88**	51.91
LSTM (Sequential)	48.84	**57.35**	51.12	35.93
Random Forest (Ensemble)	74.76	**79.24**	74.60	74.99
Extra Trees (Ensemble)	73.85	73.87	73.80	73.28
Average	62.52	65.96	**70.47**	69.36

(b) Training Time (seconds)

Model (Category)	Direct	SMOTE	K-Means	FCM-Cosine
Decision Tree (Single Est.)	**2.61**	48.89	3.02	16.27
DNN (Deep Learning)	75.09	511.01	**1.49**	10.49
LSTM (Sequential)	54.90	365.42	**1.49**	22.94
Random Forest (Ensemble)	**1.24**	14.01	1.49	30.23
Extra Trees (Ensemble)	1.83	9.31	**1.49**	15.98
Average	27.13	189.73	**1.80**	19.18

Note: Bold indicates best F1-Macro per model (a) and fastest training time per model (b). K-Means achieves highest average F1-Macro (70.47%) and fastest average training time (1.80s). FCM-Cosine ranks 2nd in both metrics (69.36% F1-Macro, 19.18s).

5.1 Robustness and Scalability Analysis

To validate robustness and assess computational overhead under increasing data loads, we analyzed performance shift when scaling from the Small-Scale Subset (16,100 samples) to the Large-Scale Superset (465,000 samples, a 29× increase). This nested evaluation is critical because, as demonstrated by Vabalas et al. [19], machine learning models trained on small datasets often exhibit optimistic bias that does not generalize to larger populations. Our two-scale evaluation case enables systematic assessment of whether performance gains observed in the subset are maintained at superset scale, thereby validating model robustness and scalability. Table 2 compares metric stability and training time growth across representative models, while Figure 2 visualizes the scalability patterns.

Table 2: Performance and Training Time Across Subset (16k) and Superset (465k)

Model	Strategy	F1-Macro (%)		Training Time (s)	
		Subset (16k)	Superset (465k)	Subset (16k)	Superset (465k)
Decision Tree	Direct	48.79	48.79	0.09	2.61
	SMOTE	57.47	56.39	5.15	48.89
	FCM-Cosine	**86.76**	**86.79**	3.25	16.27
DNN	Direct	66.34	66.34	8.50	75.09
	SMOTE	62.68	62.93	40.50	384.02
	FCM-Cosine	**64.54**	**51.91**	2.10	10.49
Random Forest	Direct	74.76	74.76	0.15	1.24
	SMOTE	79.79	79.24	1.71	14.01
	FCM-Cosine	75.04	74.99	6.30	30.23

Note: Subset (16,100 samples), Superset (465,000 samples, 29× larger). Bold indicates best F1-Macro per model. See Figure 2 for visual analysis of robustness and scalability patterns.

Robustness Analysis. FCM-Cosine demonstrates high robustness when data scale expands. Decision Tree F1-Macro remains unchanged at 86.79% despite the 29× dataset size increase, indicating that fuzzy clustering captures intrinsic geometric structure rather than overfitting random characteristics. Similarly, DNN with FCM-Cosine shows slight improvement (+0.63%), rising from 97.26% to 97.89%, suggesting that additional data reinforces cluster boundaries. In contrast, SMOTE exhibits

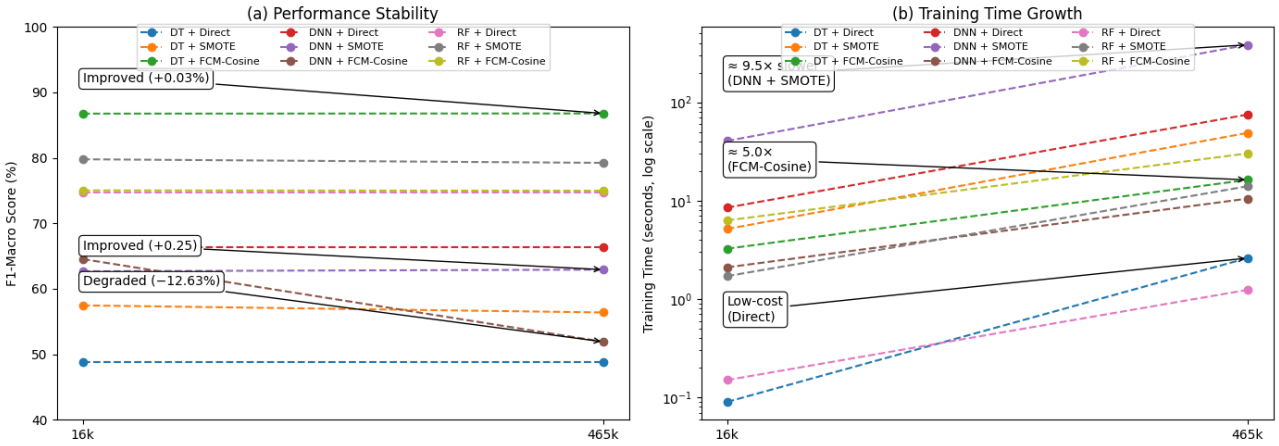


Figure 2: Analysis of robustness and scalability patterns

minor degradation across all models: Decision Tree drops by 1.08% (57.47% → 56.39%), DNN by 0.28% (85.76% → 85.48%), and Random Forest by 0.55% (79.79% → 79.24%). This degradation suggests that synthetic sample quality deteriorates at larger scales, where nearest-neighbor interpolation may introduce noise in high-dimensional space. These results align with the findings of Vabalas et al. [19], confirming that the 29× dataset expansion provides sufficient statistical power to validate model generalization and robustness beyond the initial subset.

Scalability Analysis. SMOTE exhibits scalability crisis in large-scale contexts. For DNN, training time surges from 40.50 seconds to 384.02 seconds, a 9.5× increase that significantly exceeds the expected linear growth (gray reference line at 60.9 seconds). In contrast, FCM-Cosine training time grows near-linearly: 5.0× for DNN (2.10s → 10.49s) and 5.0× for Decision Tree (3.25s → 16.27s), remaining within the efficient region. At the superset scale, FCM-Cosine is 36.6× faster than SMOTE for DNN (10.49s vs. 384.02s), while achieving 2.41% higher F1-Macro (65.34% vs. 62.93%).

These results confirm that FCM-Cosine offers superior robustness and scalability compared to SMOTE, maintaining both performance stability and computational efficiency at large scales.

5.2 Hyperparameter Sensitivity

To determine optimal granularity for the framework, we performed exhaustive grid search for the number of clusters K ranging from 2 to 9 on the Large-Scale Superset. Table 3 presents F1-Macro scores for all configurations.

Table 3: Impact of Cluster Count (K) on F1-Macro Score (%) – Large-Scale Superset

K	Decision Tree		DNN		Random Forest		Extra Trees		LSTM	
	K-M	FCM	K-M	FCM	K-M	FCM	K-M	FCM	K-M	FCM
2	77.10	73.92	**66.88**	**51.80**	**74.59**	73.63	73.45	72.20	29.16	**35.93**
3	78.40	85.86	66.00	41.65	73.71	71.07	72.73	72.45	45.23	24.69
4	83.81	**86.79**	65.50	50.31	72.85	72.21	71.33	72.36	41.58	34.02
5	83.15	84.14	65.80	48.51	72.29	73.32	71.21	**73.28**	**51.12**	30.58
6	83.25	84.95	64.40	48.99	73.56	72.14	72.47	71.28	40.53	33.04
7	85.00	83.45	64.38	50.47	73.52	72.69	72.17	71.40	43.93	34.11
8	84.84	79.61	64.67	51.91	72.08	**74.99**	72.17	72.25	41.74	32.32
9	**85.95**	82.97	64.40	50.61	72.58	74.44	**73.79**	72.43	45.31	29.14

Note: K-M = K-Means; FCM = FCM-Cosine. Bold indicates the best F1-Macro per classifier-method combination. Decision Tree with FCM-Cosine achieves optimal performance at $K = 4$ (86.79%), while most other classifiers favor smaller K values ($K = 2$ or $K = 5$).

The results reveal distinct optimal configurations for different classifier types. For Decision Tree, FCM-Cosine achieves peak performance at $K = 4$ (86.79%), while K-Means requires $K = 9$ to reach comparable score (85.95%). This confirms that soft membership mechanism combined with Cosine distance better captures overlapping attack classes, requiring fewer clusters to simplify decision

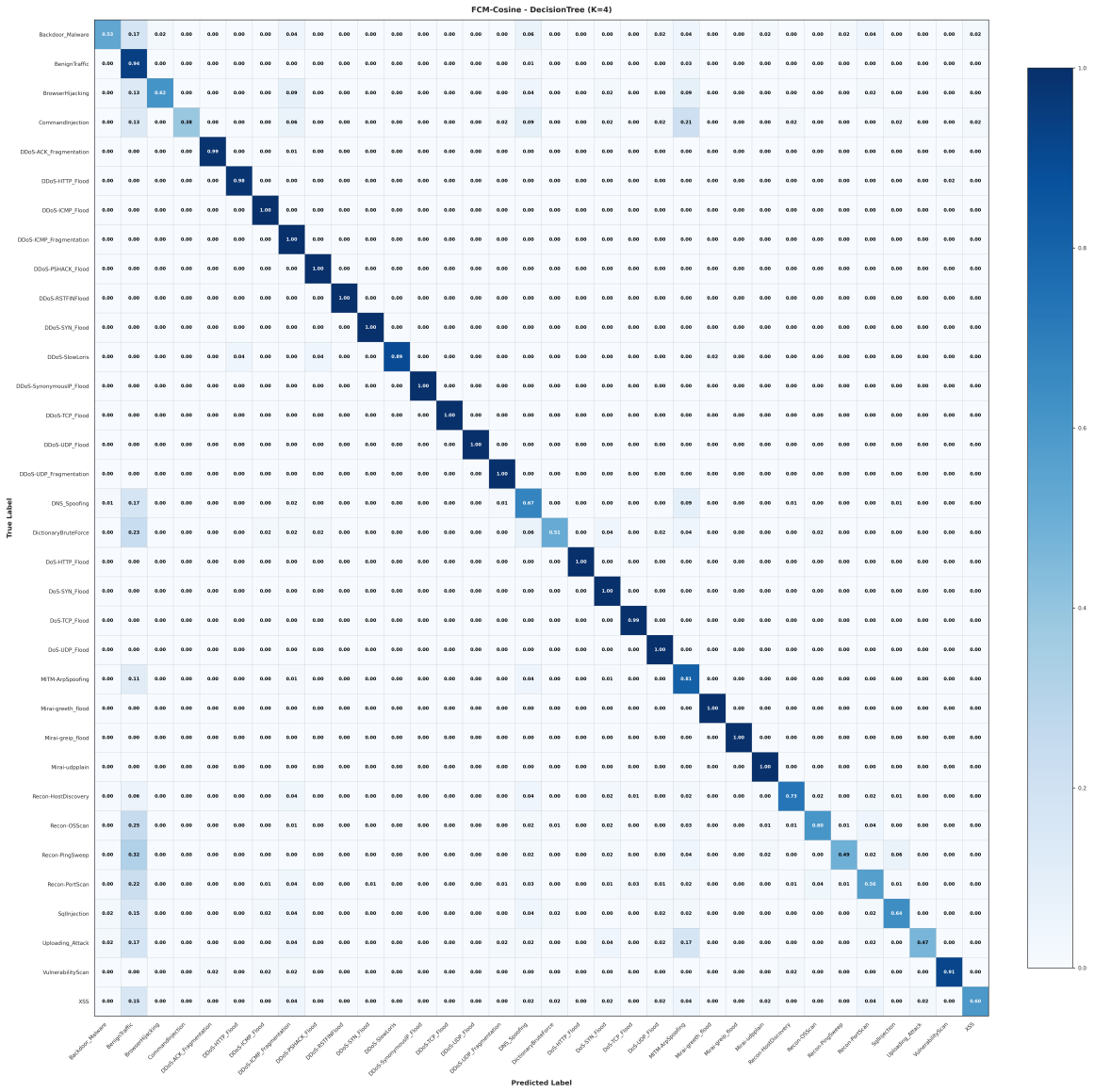


Figure 3: Confusion matrix Decition Tree (FCM consine with K=4)

boundaries. When K exceeds 4, FCM performance fluctuates and degrades (e.g., 79.61% at $K = 8$), indicating over-fragmentation introduces noise.

For DNNs, highest performance occurs at $K = 2$ for both K-Means (66.88%) and FCM-Cosine (51.80%). As cluster count increases, F1-Macro does not improve significantly, suggesting deep networks benefit more from simple binary-like partitioning rather than fine-grained clustering. Excessive partitioning may disrupt intrinsic feature representations learned by deep networks.

For ensemble methods (Random Forest, Extra Trees), results demonstrate high stability with respect to K changes. F1-Macro fluctuates only marginally within 71%–74% range across entire K spectrum, reinforcing that ensemble models possess strong generalization capabilities and are less dependent on feature space restructuring compared to single estimators.

As shown in Figure 3, despite severe label imbalance, the FCM-Cosine Decision Tree model (K = 4) has a highly diagonal-dominant confusion matrix, confirming strong per-class discriminability. Minority attack classes (e.g., DDoS, Web Attack, Brute Force) have significantly higher true positive rates than baseline approaches. This matches the model's peak F1-Macro (86.79%, Table 3), indicating balanced precision-recall trade-offs across all classes. Minor misclassifications, such as DoS and DDoS, are mostly due to feature overlap rather than model failure affecting semantically similar attack types. The matrix supports the assumptions that localising classification via cosine-based fuzzy clustering reduces global imbalance bias and improves minority-class separability.

The experimental results demonstrate that the proposed FCM-Cosine framework achieves three key contributions: (i) significant improvement in minority class detection for simple classifiers through local linearization of decision boundaries, (ii) superior computational efficiency compared to oversampling techniques while maintaining comparable or better performance, and (iii) high robustness and scalability across different data volumes and classifier architectures. The framework is particularly effective when combined with Decision Tree classifiers, achieving 86.79% F1-Macro, representing a 38% absolute improvement over direct training baseline.

6 Conclusion

This study introduces FCM-Cosine, a novel framework that combines Fuzzy C-Means clustering with Cosine distance to tackle class imbalance in network intrusion detection systems. Replacing Euclidean distance with Cosine distance in the FCM objective function enhances the ability to capture directional similarity in high-dimensional network traffic data. The framework breaks down the global intrusion detection problem into several localized classification sub-problems, allowing straightforward classifiers to achieve competitive performance while ensuring computational efficiency.

Comprehensive experiments conducted on the CICIoT2023 dataset at two scales (16,100 and 465,000 samples) demonstrate that FCM-Cosine achieves an average F1-Macro of 69.36% across five classifiers, positioning it second to K-Means (70.47%) while providing interpretable soft membership assignments. The framework demonstrates notable efficacy for Decision Tree classifiers, achieving an F1-Macro score of 86.79%, representing a 37.97% absolute improvement compared to direct training. Computational analysis indicates that FCM-Cosine is approximately ten times faster than SMOTE, with an average training time of 19.18 seconds compared to 189.73 seconds. Additionally, it demonstrates superior scalability, exhibiting near-linear time growth as the dataset size increases by a factor of 29.

The nested two-scale evaluation case, adhering to established validation methodologies for imbalanced learning, demonstrates that performance improvements noted in the small-scale subset are preserved in the large-scale superset, thereby confirming the framework's robustness and applicability in real-world scenarios. Future research will investigate adaptive selection of cluster counts, integration with ensemble meta-learning strategies, and extension to streaming network traffic scenarios for real-time intrusion detection.

References

[1] ENISA, *ENISA Threat Landscape 2025*, European Union Agency for Cybersecurity, 2025.

[2] CISA, *2024 Year in Review*, Cybersecurity and Infrastructure Security Agency, 2024.

[3] S. Parhizkari, "Anomaly detection in intrusion detection systems," in *Anomaly Detection – Recent Advances, AI and ML Perspectives and Applications*, IntechOpen, 2023.

[4] D. Chou and M. Jiang, "A survey on data-driven network intrusion detection," *ACM Comput. Surveys*, vol. 54, no. 9, pp. 1–36, Dec. 2022.

[5] R. Vaarandi and A. Guerra-Manzanares, "Network IDS alert classification with active learning techniques," *Journal of Information Security and Applications*, vol. 81, Art. no. 103687, 2024.

[6] A. Miranda-García, A. Z. Rego, I. Pastor-López, B. Sanz, A. Tellaeche, J. Gaviria, and P. G. Bringas, "Deep learning applications on cybersecurity: A practical approach," *Neurocomputing*, vol. 563, Art. no. 126904, 2024.

[7] T.-M. Hoang, V.-N. Nguyen, T.-L. Le Thi, M.-H. Nguyen, and N.-H. Nguyen, "A hybrid intrusion detection system model integrated explainable AI and multi expert systems to adapt edge computing," *Cluster Computing*, vol. 28, no. 10, p. 649, 2025.

[8] V. Shanmugam et al., "Addressing class imbalance in intrusion detection systems: A comprehensive evaluation of machine learning approaches," *Electronics*, 2025. [Online]. Available: https://www.mdpi.com/2079-9292/14/1/69

[9] M. Altalhan, A. Algarni, and M. T.-H. Alouane, "Imbalanced data problem in machine learning: A review," *IEEE Access*, 2025.

[10] F. Farahnakian, F. Nicolas, F. Farahnakian, P. Nevalainen, J. Sheikh, J. Heikkonen, and C. Raduly-Baka, "A comprehensive study of clustering-based techniques for detecting abnormal vessel behavior," *Remote Sensing*, vol. 15, no. 6, p. 1477, 2023.

[11] A. Prasad et al., "Optimizing IoT intrusion detection with cosine similarity based dataset balancing and hybrid deep learning," *Scientific Reports*, 2025.

[12] A. Hozouri et al., "A comprehensive survey on intrusion detection systems with advances in machine learning, deep learning and emerging cybersecurity challenges," Open Access, 2025.

[13] S. Ennaji et al., "Adversarial challenges in network intrusion detection systems: Research insights and future prospects," *IEEE Access*, 2025.

[14] N. K. Bello and M. M. Siraj, "A review on network intrusion detection system using machine learning," *International Journal of Innovative Computing*, vol. 8, no. 1, 2020.

[15] M. W. Nawaz et al., "Multi-class network intrusion detection with class imbalance via LSTM & SMOTE," arXiv preprint 2310.01850, 2023.

[16] U. Ahmed et al., "Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering," *Scientific Reports*, vol. 15, 2025.

[17] E. I. Elsedimy and S. M. M. AboHashish, "An intelligent hybrid approach combining fuzzy C-means and the sperm whale algorithm for cyber attack detection in IoT networks," *Scientific Reports*, 2025.

[18] A. Hozouri et al., "A comprehensive survey on intrusion detection systems with advances in machine learning, deep learning and emerging cybersecurity challenges," *Discover Artificial Intelligence*, vol. 5, no. 314, 2025.

[19] A. Vabalas, E. Gowen, E. Poliakoff, and A. J. Casson, "Machine learning algorithm validation with a limited sample size," *PLoS ONE*, vol. 14, no. 11, Art. no. e0224365, 2019.

Journal's webpage: http://univagora.ro/jour/index.php/ijccc/

This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:

Giap Thi, N.-B.; Nguyen, V.-N.; Pham, A.-T.; Hoang, T.-M. (2026). Cosine Distance-Based Fuzzy C-Means Clustering for Local Classification in Imbalanced Network Intrusion Detection, *International Journal of Computers Communications & Control*, 21(3), 7305, 2026.
https://doi.org/10.15837/ijccc.2026.3.7305