ICCA: An Improved Intrusion Detection Algorithm for Healthcare Data Classification and URLs phishing

Abdalraouf Alarbi[1], Zafer Albayrak[2*], Muhammet Çakmak[3], Hakan Can Altunay[4]

[1] Department of Computer Science, School of Engineering,University of Bridgeport, 126 Park Avenue, Bridgeport, CT 06604, United States
Email: aalarbe@my.bridgeport.edu

[2] Department of Computer Engineering, Sakarya University of Applied Sciences, 54100 Sakarya, Turkey
Email: zaferalbayrak@subu.edu.tr

[3] Department of Computer Engineering, Sinop University, 5700 Sinop, Turkey
Email: mcakmak@sinop.edu.tr

[4] Department of Computer Technologies, Carsamba Chamber of Commerce Vocational School, Ondokuz Mayis University, 55200 Samsun, Turkey
Email: hakancan.altunay@omu.edu.tr

[*] Corresponding author

Abstract: Classification is a fundamental task in machine learning that involves assigning data instances to one or more predefined categories or classes. Among the various classification algorithms available is the Core Classification Algorithm (CCA). However, CCA has limitations, particularly when dealing with high-dimensional data, which can negatively affect its classification performance. To address these limitations, this study proposes a new algorithm called the Improved Core Classification Algorithm (ICCA), which enhances the performance of CCA by incorporating novel features and techniques. In this article, the principles and design of ICCA were described and its performance was compared to that of CCA and other state-of-the-art classification methods on four datasets from the healthcare and phishing URLs domains. Experimental results on four datasets demonstrate that ICCA consistently outperforms the original CCA, achieves the highest accuracy on the high-dimensional phishing and cardiovascular datasets, and remains competitive on imbalanced medical data. Overall, this work contributes to the advancement of classification algorithms and provides a valuable tool for various real-world applications.

Keywords: Classification; Phising attacks; Hybridization; cybersecurity

1 Introduction

In the discipline of machine learning, classification is a fundamental process that entails grouping data points into different categories based on specific features or labels. This task requires grouping data points into distinct categories based on certain features or labels. The research that has been done on classification has been published in a wide variety of fields, such as statistics and expert systems. Typically, the process of classification will require a two-step method, in which a model will first classify a set of data classes and then assess the predictive consequences of its processing. The Support Vector Machine (SVM), the Naive Bayes algorithm, Decision Trees, and Neural Networks are a few examples of the categorization algorithms that are used most frequently [1, 2].

Machine learning is revolutionizing the healthcare industry by offering advanced solutions to enhance disease diagnosis, treatment planning, and preventive care. One of the most promising applications of machine learning in healthcare is the ability to predict patient outcomes and identify individuals at risk of developing severe conditions like heart disease, cancer, or diabetes. These predictive models can assist clinicians in making data-driven decisions that improve patient care. For instance, machine learning algorithms have been employed to analyze EHRs, genetic information, and imaging data to forecast disease progression or recommend personalized treatment plans. Furthermore, machine learning is transforming medical imaging by improving the quality and efficiency of diagnostic tools such as CT and MRI scans. Algorithms designed to detect anomalies can automate the identification of abnormalities in images, reducing the reliance on human interpretation while increasing accuracy. As a result, radiologists can provide faster and more accurate diagnoses, enabling earlier interventions that could save lives[3, 4]. In summary, the integration of machine learning into healthcare has the potential to significantly improve patient outcomes, reduce healthcare costs, and streamline clinical workflows.

Cybersecurity is more important than ever as technology becomes central to daily life. With the rise of cyberthreats like phishing and data breaches, protecting sensitive information is crucial [5-7]. Staying alert and using updated security measures can help individuals and organizations guard against attacks, making digital spaces safer for everyone [8, 9].

Phishing attacks have emerged as a major threat in today's digital landscape, with perpetrators deploying increasingly sophisticated tactics to deceive users into sharing sensitive information. Traditional approaches to detecting phishing URLs often rely on comparing incoming URLs against databases of known phishing sites. However, this method struggles to keep pace with the growing complexity and volume of phishing attempts, necessitating more advanced techniques. Machine learning offers a powerful solution by analyzing URL features and identifying patterns indicative of phishing attempts. By training algorithms to detect both

known and previously unseen phishing URLs, machine learning models can enhance the detection of malicious URLs with greater accuracy and adaptability. This capability allows for real-time identification of phishing threats, providing a stronger defense compared to conventional methods [10, 11].

In this study, a new algorithm called the Improved Core Classification algorithm (ICCA) is presented and it's derived from the previous algorithm CCA [12]. This work depends on the use of an active set (A_S) which gives a better representation of the class and classifies the point according to the vote in similarity by measuring the distance using Euclidean distance. Despite the fact that K-means algorithms' output varies widely from implementation to implementation, this property was leveraged throughout the training model phase to achieve better accuracy overall. The suggested algorithm (ICCA) will be tested and validated in the context of phishing URLs and healthcare domains.

The rest of this paper is organized as follows: Section 2 presents a comprehensive review of the existing literature related to classification algorithms, machine learning applications, and the challenges they address. Section 3 introduces the ICCA, detailing its methodology, theoretical foundation, and potential applications. Section 4 provides a thorough analysis of the experimental setup, data collection, and results, followed by an evaluation of the algorithm's performance. Finally, Section 5 concludes the study with a discussion of the implications of this research, its limitations, and potential directions for future work.

2 Related Work

Hybrid approaches combining clustering and nearest neighbor methods have been effectively applied to various domains, including time series forecasting and instance selection for classification. Hnin et al. (2024) proposed a hybrid model integrating K-means clustering with K-nearest neighbors (KNN) for short-term load forecasting (STLF) in Thailand's electricity grid. The method clusters historical load patterns and uses KNN to classify days based on features such as day of the week, month, and holiday status, addressing non-linear variations during the holidays. Forecasting within each cluster employs models like linear regression, neural networks with Bayesian optimization, SVR with Bayesian optimization, and LSTM with Bayesian optimization, where the latter achieves the lowest MAPE. Evaluated on EGAT data using ANOVA and Tukey's HSD tests, the approach yields a 56.1% accuracy improvement over baselines, particularly excelling on holidays, though it remains sensitive to cluster quality. This clustering-classification paradigm is particularly effective for domain-specific imbalanced time series [13].

Similarly, Saha et al. (2022) introduced CIS, a cluster-oriented instance selection algorithm that applies K-means to partition training data, followed by selecting

central and border instances at user-controlled rates to reduce noise while maintaining representational accuracy. Unlike prior methods, CIS enables direct control over selection rates and better captures cluster characteristics. Tested on 24 benchmarks, it enhances KNN accuracy by 2-3% compared to alternatives like RIS and GDIS at equivalent reduction rates, demonstrating robustness across classifiers like GNB and LSVM. This framework supports prototype selection in high-dimensional classification tasks [14].

Advances in KNN-based classification for remote sensing images include works using support vector machines (SVM) with improved optimal index factor (OIF) for hyperspectral data, achieving high accuracy and optimal band selection through one-to-one strategies (Juan W.). Zhao L. employed an enhanced KNN variant, building on cropping concepts, for object-oriented classification of high-resolution images, showing improved accuracy over baselines. Sulianova (2023) compared classification outcomes, confirming the benefits of refined KNN approaches. Additionally, decision tree models with attribute analysis have been used to diagnose failures in high-reliability gas chromatographs (HRGCs), providing insights via confusion matrices and feature weights [15]

In limited-label scenarios, Gweon and Yu (2021) developed a nearest neighbor-based active learning strategy for time series classification, employing local uncertainty and utility metrics from inter-instance distances to select informative unlabeled samples. Supporting batch-mode and multi-class problems, it leverages 1NN strengths with soft probabilities. On datasets like WAFER and ECG5000, it surpasses random sampling and competitors like NETS in accuracy and stability, though sensitive to distance measures (recommending DTW for misaligned series). This advances efficient classification in label-scarce domains like healthcare [16].

Wan et al. (2021) proposed NCE-Net, embedding a nearest neighbor classifier within deep neural networks to improve active learning generalization under distribution biases, using prototypes and noise-conditioned embedding (NCE) loss. It progressively generalizes beyond softmax limitations, querying via rejection or confusion confidence. Subset information analysis validates reduced overestimation risks. On CIFAR-10/100 and PASCAL VOC, it gains 1-5% accuracy over baselines like LL4AL and Core-set with fewer labels, offering a task-agnostic solution aligned with prototype strategies in imbalanced data [17].

Hybrid metaheuristic optimization techniques have also advanced parameter tuning and complex problem-solving. Yuan and Gallagher (2005) introduced a meta-evolutionary algorithm combined with racing for genetic algorithm (GA) parameter optimization, adopting a variable-centric view with genetic operators and statistical performance evaluation across configurations. The approach proves reliable and efficient on benchmark problems, handling challenges in parameters lacking coherent distance metrics [18].

Martinez-de-Pison et al. (2017) developed a hybrid method merging Bayesian optimization (BO) with a constrained GA-PARSIMONY variant to derive

parsimonious models, mitigating computational costs. BO initializes parameters, followed by restricted GA for feature reduction, transformation, and selection. Tested with XGBoost on UCI datasets, it matches GA-PARSIMONY accuracy while significantly reducing runtime in most cases [19].

Shang et al. (2006) presented a hybrid ant colony and particle swarm optimization algorithm for the traveling salesman problem (TSP). It generates initial solutions statistically, disseminates pheromones, applies ant colony for multi-solution creation via pheromone updates, and refines via PSO crossover/mutation. Across 16 variants, it outperforms simulated annealing, standard GA, and ant colony algorithms, with specific crossover/mutation strategies proving particularly efficient [20].

The literature review reveals that numerous algorithms have been employed for classification purposes, yielding varying outcomes. The CCA algorithm exhibits certain limitations in terms of its performance when compared to alternative algorithms. The proposed algorithm enhanced the performance of the CCA by incorporating an A_S. Additionally, a parameterization technique was used to represent the number of points that signify the class. The experimental results demonstrate superior accuracy performance in comparison to other algorithms across four distinct data sets.

3 Proposed Algorithm

The proposed algorithm is derived from the CCA algorithm in that it is based on studying the similarities of all points with an A_S that contains the points with the most connectivity of the class and its variance; it represents the parameterization of this algorithm, and the A_S numbers are the number of points in the A_S. Beta " β", as shown in Figure 1. The aforementioned point is commonly regarded as the accurate depiction of the class given that it encompasses the majority, if not all, of its defining attributes. The algorithm under consideration has the capability to surmount the obstacles posed by the CCA algorithm, leading to enhanced accuracy and superior performance, as evidenced by the results presented in Figure 2.

The objective of this study is to combine the algorithm obtained from CCA with a partition-based unsupervised learning algorithm named K-means. The hybridization process was intended to enhance the robustness of the derived algorithm and enable it to handle large datasets comprising multiple domains and cases with greater efficiency. Machine learning algorithms exhibit varying strengths and weaknesses in their mechanisms, rendering them efficacious in specific scenarios and inadequate in others.

In numerous scenarios, combining two algorithms through hybridization is crucial for enhancing their performance and increasing efficiency. Notably, it proves to be

an effective solution to overcome shortcomings and challenges that one of the algorithms may face. Historically, hybrid algorithms were developed by leveraging the strengths of one algorithm to improve the performance and efficiency of another. Nevertheless, our research reveals that using the weaknesses of one algorithm to enhance the efficiency of the other is equally, if not more, essential.

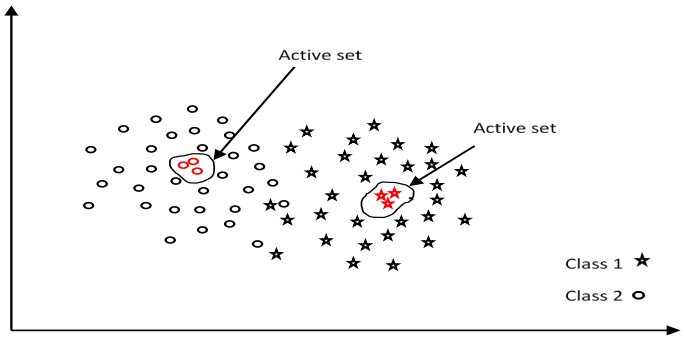

Figure 1
ICCA classification algorithm

The suggested algorithm (ICCA) relies on the following concepts for its fundamental mechanism:

i. 1: The present study involves the simulation of the Canonical Correlation Analysis (CCA) algorithm to identify an A_S t for each class, which is characterized by its unique features. This approach is aimed at addressing the challenge posed by the high data distribution concept.

ii. 2: The utilization of clustering algorithms can effectively address challenges related to the distribution of datasets, including but not limited to nonlinear classification, overlapping, and noise.

3.1 Mathematical Notation of the Study

i. X: Data matrix

ii. w^1: Transformation vector for view 1

iii. w^2: Transformation vector for view 2

iv. ρ: Correlation coefficient

v. A: Active set matrix

vi. A_ij: Element of the Active set matrix (A), indicating the connection between data points i and j

vii. DM^1c: Distance matrix for class c

viii. DM^1c_ij: Distance between data points i and j in class c

ix. β: Threshold value for A_S selection

x. x: New data point

3.2 Improved Core Classification Algorithm ICCA

Using the Active Set (A_S) concept, the Improved Core Classification Algorithm (ICCA) extends the capabilities of core classification algorithms. The basic mathematical framework of the ICCA is explained in this section.

3.3 Determining A_S

3.3.1 Distance Matrix Construction

Each class c has a distance matrix (DM^1c) that has been computed. Pairwise distances for each class C data point are contained in this matrix. In mathematical terms, the distance between data points i and j in class c is represented by the element DM^3c_ij. To calculate this distance, a certain distance metric such as the Euclidean distance can be used:

Equation 1

The formula

$DM^1c\_ij = sqrt((X\_ci^1 - X\_cj^1)^2 + (X\_ci^2 - X\_cj^2)^2 + ... + (X\_ci^d - X\_cj^d)^2)$

where X_ci^k represents the kth feature value of data point i in class c, and d is the total number of features.

3.3.2 A_S Selection

A threshold value (β) is established in order to identify the data points in class C with the highest degree of interconnectivity . If the total distance of a data point from every other point in the class is below this threshold, then that point is assigned to the Active Set (A_S). This ensures that the A_S comprises the most representative “core” points of the class — those that are closest on average to all other points in the same class, capturing the class’s central tendencies and reducing the influence of outliers or peripheral points. The selection procedure can be expressed mathematically as follows:

Class C data point i is included in the Active Set matrix A (A_ic = 1) or not (A_ic = 0). According to

Equation 2

$A\_ic = 1 \text{ if } \Sigma\_j DM^c\_ij < \beta, \quad \text{for all } i \in c$

where A_ic is an element of the Active Set matrix (A), indicating whether data point i in class c belongs to the A_S (A_ic = 1) or not (A_ic = 0). Here, β serves as a tunable upper bound on the cumulative distance: lower values of β produce a smaller, more tightly connected A_S (fewer but highly central points), while higher values include more points but may slightly reduce centrality. This parameterization allows the algorithm to adapt to dataset characteristics such as noise level or class density.

3.3.3 Classification

Each dataset A_S's distance to a fresh data point (x) is calculated as part of the categorization procedure. Next, the class whose A_S displays the least distance is assigned the new data point. The mathematical nuances of this phase will be incorporated in a later iteration.

3.4 Convergence Metrics

This work aimed to establish the fundamental principles and characteristics of the Improved Core Classification Algorithm (ICCA). However, evaluating the ICCA's convergence behavior is an important consideration for further research. Convergence measurements can provide valuable insights on the scalability and efficiency of an algorithm, particularly for large datasets. As performance metrics, accuracy, precision, and recall will be analyzed along with this convergence statistic, which is expressed as a percentage of all the data points. The purpose of incorporating convergence metrics into the evaluation process is to gain a deeper understanding of the performance characteristics of ICCA. This data is crucial for assessing the system's suitability for real-world classification tasks, especially ones that call for complex and large datasets.

3.5 The Pseudo-Code of the Proposed Algorithm

The present study will demonstrate the methodology for training a dataset and subsequently partitioning the outcomes into two distinct categories. The construction of the distance matrix (DM)ci is carried out for each class, whereby the summation of each row of the matrix indicates the degree of similarity between a given point (row) and the other points belonging to the same class. The present class contains a collection of entities that exhibit the greatest degree of interconnectivity, referred to as the active set A_S. This A_S is recorded in the Active Set Matrix, and the Beta value is established as a variable quantity that represents the number of points within the A_S. Therefore, it can be argued that it provides the most accurate depiction of the aforementioned category. The test data will be categorized based on their resemblance to the A_S.

i. Compute the distance matrix between the two classes.
ii. Set the value of Beta as the cardinality of the A_S, representing the number of objects within it.
iii. The A_S matrix for each class can be obtained by measuring the distance matrix.
iv. Compute the distance between the variable x and each set of active objects.
v. In accordance with the minimum distance criterion, x shall be assigned to the class that exhibits the lowest distance.
vi.

The following procedures are depicted in pseudocode as presented below, while the corresponding symbols are illustrated in Table 1

Algorithm: Algorithm for ICCA

```
1: Input: training dataset
2: Output: classify the test point into its class
3: Initialization:
4: Find the  A_S for each class
5: loop process
6: for i=1 to length of test_ds do
7:     find the dis between test_ds[i] and  A_S objects
8:     if dis(1)  is minimum
9:         Classify test_ds[i] to class(1)
10:    else
11:        Classify test_ds[i] to class(2)
12: end loop
```

Table 1
Symbols of the pseudocode

Symbol	Explanation
D_M	Distance matrix
c1,c2	Class1 , class2
ds_{c1}, ds_{c2}, ds_{c3}	The number of points in each class
$(D_M)_{c1}$,$(D_M)_{c2}$	Distance matrix for each class
B	Number of objects in the class
A_S	Active set Ative

Figure 2 showed a flowchart presented to depict the Improved Core Classify Algorithm (ICCA), which aims to enhance the classification process. The algorithm is designed to handle a scenario with two classes, each of which is divided into two clusters. The testing dataset is represented by the variable "n", while "j" is used as a counter. The model is trained using "I" as a counter for the number of iterations. During each iteration, the accuracy and A_S are recorded. The A_S of clusters that exhibit high accuracy is selected as the appropriate A_S for a perfect model.

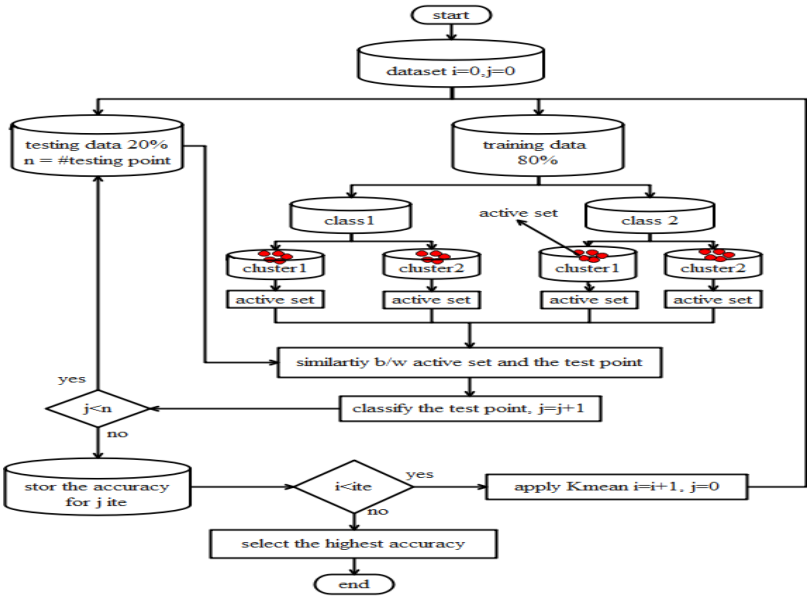

Figure 2
Flowchart of ICCA algorithm

4 Experimental Analysis and the Results

To ensure reproducibility and provide a clear evaluation framework, this section details the datasets used, their characteristics, preprocessing steps, and the experimental configuration.

The experiments were conducted on four publicly available datasets: one for phishing detection and three for healthcare (liver disease, cardiovascular disease, and heart disease prediction). These datasets were selected to evaluate ICCA's performance on high-dimensional, potentially imbalanced classification tasks relevant to intrusion detection and medical diagnostics. Table 2 summarizes the key characteristics of each dataset, including the number of instances, features, class distributions, and imbalance ratios [21-24].

Table 2
Dataset Characteristics

Dataset	Source/ Reference	Instances	Features	Classes	Distribution	Imbalance Ratio
Phishing Dataset	Tan (2018) [21]	10,000	48	Phishing (1), Legitimate (0)	5,000 (Phishing), 5,000 (Legitimate)	1:1 (balanced)
Indian Liver Patient Records	UCIML (2019) [22]	583	10	Liver Disease (1), No Liver Disease (2)	416 (Liver Disease), 167 (No Liver Disease)	2.49:1
Cardiovascular Disease Dataset	Sulianova (2023) [23]	70,000	11	CVD Present (1), CVD Absent (0)	35,021 (Present), 34,979 (Absent)	~1:1 (nearly balanced)
Heart Disease Prediction	Dileep (2023) [24]	4,238	15	Heart Disease (1), No Heart Disease (0)	642 (Heart Disease), 3,596 (No Heart Disease)	5.6:1

Table 3 provide the results of an analysis of four separate datasets; these datasets are designated "1-4." In any event, the subsequent trials demonstrate that the findings of the ICCA have undergone a significant improvement, which is particularly noticeable when the number of cores used in each class is increased. The ICCA was analyzed with the use of the confusion matrix, and the F1-score, precision, and recall were all computed. Considering Datasets 1-4, ICCA provided varying outcomes depending on the dataset; each experiment was executed with 20, 50, and 100 iterations. In Table 3, the ICCA was analyzed without the use of clustering, and the Beta was equal to 5.

Table 3
Results of ICCA, where β = 5, Number of Cluster = 0

Data set	**Accuracy**	**Precision**	**recall**	**F1 score**
1	77.12	77.31	76.08	76.08
2	68.96	67.44	66.83	65.56
3	60.83	60.91	60.90	60.10
4	62.81	54.52	52.42	61.21

By making use of the Cluster, setting the Beta to 5, and iterating for a total of 20, 50, and 100 times (Table 4), the data suggested increases when the number of iterations was increased. The results were much improved with clustering as compared to when it was not used.

Table 4
Results of ICCA where β = 5, Number of Cluster = 2, iteration = 20,50, 100

Data set	**iteration**	**Accuracy**	**Precision**	**Recall**	**F1 score**
1	20	78.47	70.42	71.77	71.09
	50	80.95	71.37	72.68	72.02
	100	79.53	67.88	71.15	69.47
2	20	61.49	56.94	55.60	56.26

	50	65.52	62.98	61.25	62.10
	100	67.24	62.66	60.65	61.64
3	20	62.33	51.75	54.35	53.02
	50	60.17	56.21	56.21	56.21
	100	61.67	60.60	61.50	61.05
4	20	80.49	60.91	56.55	58.65
	50	81.22	61.99	57.10	59.45
	100	80.22	60.95	56.95	58.88

Additionally, Table 5 showed that the findings were much better when the cluster size was raised to three; this resulted in a more evenly distributed dataset, which in turn led to superior outcomes.

Table 5

Results of ICCA where β = 5, Number of Cluster = 3, iteration = 20,50,100

Data set	**iteration**	**Accuracy**	**Precision**	**Recall**	**F1 score**
1	20	83.36	80.45	80.03	80.24
	50	85.35	74.39	75.44	74.91
	100	88.15	75.84	76.72	76.28
2	20	72.41	55.06	67.13	60.50
	50	75.86	64.20	65.38	64.79
	100	68.97	61.24	60.33	60.78
3	20	67.00	44.07	44.50	44.64
	50	61.50	57.07	57.10	57.09
	100	64.50	62.91	63.50	63.21
4	20	72.23	51.87	57.22	51.54
	50	80.71	47.35	48.71	48.02
	100	80.75	57.33	54.16	55.70

In Table 6, the Beta value is 9 and th clustering was not provided; the results for this case indicate more improvements compared to the comparative methods when the lambda value was equal to 5.

Table 6

Results of ICCA where β = 9, Number of Cluster = 0

Data set	Accuracy	Precision	Recall	F1 score
1	78.66	77.36	77.29	76.31
2	70.69	68.62	66.67	67.63
3	61.83	61.87	61.98	61.92
4	63.45	58.80	55.01	56.84

It may be concluded that using the cluster and raising the value of Beta yields more accurate results as seen in Table 7.

Table 7
Results of ICCA where β = 9, Number of Cluster = 2, iteration = 20,50,100

Data set	iteration	Accuracy	Precision	Recall	F1 score
1	20	80.52	71.62	72.84	72.22
	50	88.79	71.38	72.56	71.97
	100	89.32	70.43	71.59	71.00
2	20	63.79	64.38	62.43	63.39
	50	68.39	67.18	65.25	66.20
	100	71.26	62.51	59.88	61.17
3	20	60.83	60.81	60.80	60.81
	50	63.33	58.76	58.87	58.81
	100	63.33	55.11	55.29	55.20
4	20	79.58	60.11	56.05	58.01
	50	68.28	61.66	57.58	59.55
	100	79.76	64.19	58.82	61.39

Finally, the greatest values of Beta and clustering were used, as well as the best outcomes in all datasets, and the results are presented in Table 8.

Table 8
Results of ICCA where β = 9, Number of Cluster = 3, iteration = 20,50,100

Data set	Iteration	Accuracy	Precision	Recall	F1 score
1	20	83.93	76.87	81.00	78.89
	50	80.42	80.66	80.13	80.40
	100	90.32	86.41	89.17	87.77
2	20	70.69	59.04	66.34	62.47
	50	71.55	62.29	63.93	63.10
	100	73.28	57.64	61.71	59.60
3	20	67.75	62.60	62.64	62.62
	50	66.25	56.79	58.33	57.55
	100	68.00	66.54	66.62	66.58
4	20	81.81	57.04	53.72	55.33
	50	81.53	57.11	53.48	55.23
	100	81.67	61.51	57.11	59.23

5 Comparison with Other Classification Algorithms

In this section, the performance of the proposed ICCA is evaluated against several widely recognized machine learning algorithms, including CCA, Support Vector Machines (SVM), and Decision Trees (DT). Table 9 summarizes the comparative performance. ICCA consistently surpasses the original CCA across all datasets. It

achieves the highest accuracy on the Phishing (high-dimensional) and Cardiovascular Disease datasets. On the Indian Liver Patient and Heart Disease datasets, SVM and Decision Trees perform slightly better, likely due to SVM's effectiveness in lower-dimensional spaces with clearer linear boundaries. This highlights ICCA's strength in complex, high-dimensional scenarios typical of intrusion detection. However, the results validate ICCA's potential as a powerful alternative for improving classification accuracy in diverse applications.

Table 9

The comparison between ICCA and other classification algorithms

No.	CCA	SVM	DT	ICCA	High accuracy
1	84.0%	93.1%	96.5%	90.32	DT
2	63.2%	71.3%	69.5%	75.86	ICCA
3	61.8%	64.3%	64.7%	68.00	ICCA
4	77.2%	84.7%	75.5%	81.81	SVM
Avg	77.2%	82.7%	81.2%	78.99%	

A chart plot of the performance of ICCA compared to other classification algorithms is presented in Figure 3.

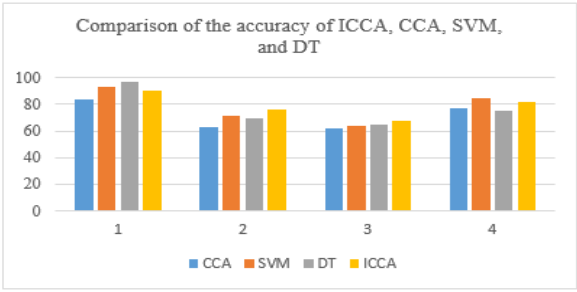

Figure 3

Performance of ICCA compared to other classification algorithms

Conclusion and Future Work

This study introduced the Improved Core Classification Algorithm (ICCA), which aimed to address the shortcomings of traditional core classification methods. ICCA uses the concept of an Active Set to carefully select the most representative data points for each class. The categorization process is optimized and accuracy is greatly increased by focusing on useful data points. The experimental results provide compelling evidence for ICCA's effectiveness. The results show that ICCA outperforms the traditional Core Classification Algorithm (CCA) in handling high-dimensional datasets. In some cases, ICCA even performed better than popular algorithms like Random Forests (RF) and Decision Trees (DT). One key area for improvement in ICCA is its temporal complexity, which can cause issues when

working with huge datasets. In this work, an Improved Core Classification Algorithm (ICCA) that leverages the most effective points to enhance the representation of data was presented. By employing the active set technique and optimizing the K-mean clustering algorithm, significant improvements were achieved in the classification accuracy of the proposed ICCA over the traditional CCA. The experimental results demonstrate that ICCA outperformed CCA, particularly when dealing with high-dimensional datasets, and in some cases, it even outperformed RF and DT algorithms. However, it is acknowledged that ICCA still has limitations, such as its time complexity, which can be a bottleneck when dealing with large-scale datasets. Addressing this limitation is an important direction for future work. Additionally, other clustering algorithms could be explored instead of K-mean to further improve the data distribution and classification accuracy of ICCA. Overall, the proposed ICCA algorithm provides a promising approach for improving classification accuracy, particularly in high-dimensional datasets. With further development and optimization, it has the potential to be a valuable tool in various real-world applications, such as medical diagnosis, fraud detection, and spam filtering.

References

[1] Salva, S. and L. Regainia, *An Advanced Approach for Choosing Security Patterns and Checking their Implementation.* arXiv preprint arXiv:2007.03275, 2020

[2] Uluer, Abdullah Fahreddin, et al. "BGP Anomali Tespitinde Hibrit Model Yaklaşımı." *2022 30th Signal Processing and Communications Applications Conference (SIU)* IEEE, 2022

[3] Esteva, A., et al., *A guide to deep learning in healthcare.* Nature medicine, 2019. **25**(1): pp. 24-29

[4] Brown, A. J., et al., *Augmenting adaptive immunity: progress and challenges in the quantitative engineering and analysis of adaptive immune receptor repertoires.* Molecular Systems Design & Engineering, 2019, **4**(4): pp. 701-736

[5] Chovancová, E. and N. Ádám, *A clustered hybrid honeypot architecture.* Acta Polytechnica Hungarica, 2019. **16**(10): pp. 173-189

[6] Altunay, Hakan Can, Zafer Albayrak, and Muhammet Çakmak. "Autoencoder-based intrusion detection in critical infrastructures." *Current Trends in Computing* 2.1 (2024): 1-12

[7] Said, Aden Ali, et al. "Performance of Ad-Hoc Networks Using Smart Technology Under DDoS Attacks." *The Proceedings of the International Conference on Smart City Applications*. Cham: Springer International Publishing, 2021

[8] Issa, A. S. A. and Z. Albayrak, *DDoS attack intrusion detection system based on hybridization of CNN and LSTM.* Acta Polytechnica Hungarica, 2023, **20**(2): pp. 1-19

[9] ISSA, Ahmed Sardar Ahmed, and Zafer ALBAYRAK. "CLSTMNet: a deep learning model for intrusion detection." *Journal of Physics: Conference Series*. Vol. 1973, No. 1, IOP Publishing, 2021

[10] Altamimi, Mubarak, et al. "BGP Anomaly Detection Using Association Rule Mining Algorithm." *Avrupa Bilim ve Teknoloji Dergisi* 42 (2022): 134-139

[11] Jain, A. K. and B. B. Gupta. *PHISH-SAFE: URL features-based phishing detection system using machine learning*. in *Cyber Security: Proceedings of CSI 2015*. 2018, Springer

[12] Alarbi, A. and Z. Albayrak, *Core Classifier Algorithm: A Hybrid Classification Algorithm Based on Class Core and Clustering.* Applied Sciences, 2022, **12**(7): p. 3524

[13] Hnin, S. W., et al., *A hybrid K-means and KNN approach for enhanced short-term load forecasting incorporating holiday effects.* Energy Reports, 2024, **12**: pp. 5942-5959

[14] Saha, S., et al., *Cluster-oriented instance selection for classification problems.* Information Sciences, 2022, **602**: pp. 143-158

[15] Zheng, Z., P. Lu, and D. Tolliver, *Decision tree approach to accident prediction for highway–rail grade crossings: Empirical analysis.* Transportation Research Record, 2016. **2545**(1): pp. 115-122

[16] Gweon, H. and H. Yu, *A nearest neighbor-based active learning method and its application to time series classification.* Pattern Recognition Letters, 2021, **146**: pp. 230-236

[17] Wan, F., et al. *Nearest neighbor classifier embedded network for active learning*. in *Proceedings of the AAAI conference on artificial intelligence*. 2021

[18] Yuan, B. and M. Gallagher. *A hybrid approach to parameter tuning in genetic algorithms*. in *2005 IEEE Congress on Evolutionary Computation*. 2005, IEEE

[19] Aldama, A., J. Ferreiro, and E. Fraile. *Hybrid Methodology Based on Bayesian Optimization and GA-PARSIMONY for Searching Parsimony Models by Combining Hyperparameter Optimization and Feature Selection*. in *Hybrid Artificial Intelligent Systems: 12th International Conference, HAIS 2017, La Rioja, Spain, June 21-23, 2017, Proceedings*. 2017, Springer

[20] Shang, G., et al. *Hybrid algorithm combining ant colony optimization algorithm with particle swarm optimization*. in *2006 Chinese Control Conference*. 2006, IEEE

[21] Tan, C. L., *Phishing dataset for machine learning: Feature evaluation.* Mendeley Data, 2018, **1**: p. 2018

[22] Repository, U. M. L. *Indian Liver Patient Records*. 2019, Available from: https://www.kaggle.com/datasets/uciml/indian-liver-patient-records

[23] Sulianova, A. *Cardiovascular Disease Dataset*. 2023 ,Available from: https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset

[24] Dileep, K. *Heart Disease Prediction using Logistic Regression*. 2023 , Available from: https://www.kaggle.com/datasets/dileep070/heart-disease-prediction-using-logistic-regression