Informatics and Control Problems 45 Issue 2 (2025)
journal homepage: www.icp.az

# Feature selection with genetic algorithm and particle swarm optimization from intrusion detection data

**Melek Ozsari[1], Sifa Ozsari[2*], İman Askerzade[2]**

[1]*Institute of Science, Ankara University, Ankara, Turkey*
[2]*Ankara University, Ankara, Turkey*

**ARTICLE INFO**

*Article history:*
Received 17.09.2025
Received in revised form 10.10.2025
Accepted 30.11.2025
Available online 26.12.2025

*Keywords:*
Intrusion detection
Feature selection
Optimization

**ABSTRACT**

*With the rapid advancement of technology and the widespread adoption of online services, ensuring the security of individuals and organizations within the digital environment has become a paramount concern. In this regard, the analysis of network traffic for the detection of cyberattacks represents a critical area of research within the domain of information security. This study investigates the efficacy of feature selection using Genetic Algorithm (GA), Particle Swarm Optimization (PSO) and Artificial Bee Colony (ABC) algorithms applied to network traffic data. The analysis focuses on binary classification, categorizing data as either "attack" or "normal" without providing granular details on specific attack types. Feature selection was performed on the USB_IDS_1 and CSE-CIC-IDS2018 datasets, with the k – Nearest Neighbor (k-NN) algorithm employed during the classification phase. Model performance was assessed using the F1-score metric. The experimental findings indicate that GA and PSO achieved favorable outcomes in feature selection.*

## 1. Introduction

With the rapid advancement of technology, access to information has become significantly easier, and many processes can now be executed rapidly. However, this progress has also led to a significant increase in malicious use and cyberattacks. Cyber threats are becoming more complex and diverse, targeting both individuals and organizations across various domains. Therefore, ensuring the security of systems has become critically important. One of the widely used tools to enhance information security is Intrusion Detection Systems (IDS). IDS primarily consists of devices or software designed to detect malicious attacks targeting systems [1]. In recent years, artificial intelligence-based methods are extensively employed in the development and efficient operation of IDS, with a significant amount of literature focused on this field [2-6].

In IDS systems that utilize machine learning algorithms, datasets are often high-dimensional and include a large number of features. Machine learning, as a branch of artificial intelligence, enables models to learn patterns from datasets and make predictions about future scenarios. The performance of these approaches is directly influenced by the structure and characteristics of the dataset. Specifically, the presence of irrelevant or redundant features can divert the model's focus to unnecessary details, adversely affecting its performance. Therefore, the feature selection process

*Corresponding author
*E-mail addresses*: mlkgmz1996@gmail.com (Melek Ozsari), ozsaris@ankara.edu.tr (Sifa Ozsari), imasker@eng.ankara.edu.tr (İman Askerzade)

www.icp.az/2025/2-04.pdf https://doi.org/10.54381/icp.2025.2.04

plays an important role in enhancing the performance of machine learning-based studies while also simplifying the analysis process. Feature selection generally refers to the identification of the most meaningful and impactful features within a dataset. The importance and advantages of this method can be summarized as follows:

- Irrelevant or unrelated features can complicate the learning process, reduce overall performance, and lead to overfitting by focusing on unnecessary details.
- Training models with fewer features optimizes time and resource utilization.
- Using fewer but more meaningful features facilitates easier interpretation of results.

Feature selection can be performed using manual or automated approaches. Manual feature selection involves identifying significant features based on expert knowledge, which can be time-consuming and heavily rely on human expertise. Automated methods, on the other hand, employ statistical computations or algorithms and are typically categorized into filtering, wrapper, and embedded methods. Additionally, optimization-based feature selection methods are frequently preferred in the literature, with numerous studies conducted on different IDS datasets [7-13].

In this study, feature selection was conducted on the USB_IDS_1 [14] and CSE-CIC-IDS2018 [15] datasets using GA, PSO and ABC. These techniques were followed by classification tasks to assess their effectiveness in improving model performance. The focus on these feature selection methods reflects their popularity and utility in optimizing the feature space for classification tasks in cybersecurity datasets.

Several previous studies have explored similar approaches on related datasets. For instance, Farhan et al. conducted an analysis of the CSE-CIC-IDS2018 dataset, applying a Binary Particle Swarm Optimization (BPSO) approach in combination with Deep Neural Networks (DNN) [16]. Their work highlighted the synergy between evolutionary feature selection methods and deep learning models.

Similarly, Alzughaibi and colleagues proposed a methodology integrating PSO with Multi-Layer Perceptrons (MLP), displaying its effectiveness in handling complex classification problems [17].

Alzubi and colleagues extended the exploration of optimization-based feature selection by evaluating methods such as Grey Wolf Optimization (GWO), Bat Algorithm (BA), and Pigeon-Inspired Optimization (PIO). These methods were applied alongside classical models like decision trees, bayesian networks, and logistic regression, demonstrating the versatility of metaheuristic algorithms in diverse classification contexts [18].

Furthermore, Srivastava and Sinha conducted a noteworthy study where PSO was used for feature selection, while genetic algorithms were employed to optimize model parameters, thereby combining two powerful optimization paradigms for improved outcomes [19].

Specific to the USB_IDS_1 dataset, Veysel and researchers have investigated the application of traditional algorithms such as Decision Trees, Random Forest, k-NN, Naive Bayes, and Artificial Neural Networks, employing GA to optimize performance in a multi-class classification framework [1]. These efforts emphasize the importance of feature selection in enhancing the predictive capabilities of machine learning models.

While studies combining GA, PSO or ABC with machine learning classifiers do exist, to the best of our knowledge, there are no studies applying GA and ABC as a feature selection method on the CSE-CIC-IDS2018 dataset. Additionally, no prior work has examined feature selection on the USB_IDS_1 dataset using binary classification. In the present study, the performances of GA combined with k-NN, PSO combined with k-NN and ABC combined with k-NN were analyzed to evaluate their effectiveness in a binary classification framework. Both the USB_IDS_1 and CSE-CIC-IDS2018 datasets were used, with the binary classification approach focusing on distinguishing between "attack" and "non-attack" instances. This approach builds upon prior research while emphasizing the importance of feature selection in improving the accuracy and efficiency of intrusion detection systems.

## 2. Material and methods

### Dataset

In this study, two renowned datasets were employed to evaluate the proposed methodology: the USB_IDS_1 dataset and the CSE-CIC-IDS2018 dataset.

The USB_IDS_1 dataset contains 83 features and 16 classes, as documented in [14], and provides a detailed representation of various network activities. The class labels are grouped to represent the defense module as follows: "Hulk-NoDefense, Hulk-Reqtimeout, Hulk-Evasive, Hulk-Security2, TCPFlood-NoDefense, TCPFlood-Reqtimeout, TCPFlood-Evasive, TCPFlood-Security2, Slowhttptest-NoDefense, Slowhttptest-Reqtimeout, Slowhttptest-Evasive, Slowhttptest-Security2, Slowloris-NoDefense, Slowloris-Reqtimeout, Slowloris-Evasive, Slowloris-Security2".

Similarly, the CSE-CIC-IDS2018 dataset, developed by the Canadian Cybersecurity Institute and detailed in [15], comprises 80 features. The dataset includes various types of attacks such as brute-force attacks (SSH, FTP), web attacks (SQL injection, XSS), denial of service (DoS, DDoS), botnet activities, and the Heartbleed vulnerability. This dataset is widely recognized in the cybersecurity community for its comprehensive representation of modern network traffic and attack scenarios. Both datasets encompass records representing a diverse range of attack types, allowing for a robust evaluation of intrusion detection systems.

Despite the richness of the datasets and their potential for multi-class classification, this study focused on a binary classification approach. In this framework, each instance in the datasets was categorized as either "attack" or "non-attack". Subsequently, data preprocessing was applied to address the imperfections in the dataset, ensuring that the data was cleaned, transformed, and prepared for analysis.

### GA, PSO and ABC

Inspired by nature and rooted in the principles of biological evolution, GAs hold a significant position among optimization and problem-solving methodologies. This approach, characterized by its relatively straightforward implementation, models biological mechanisms such as natural selection, crossover, and mutation. Genetic algorithms have a broad range of applications, spanning fields from engineering and biology to artificial intelligence and economics.

The basic steps of GA, introduced by John H. Holland [20], can be summarized as follows:

- Depending on the problem, a population of candidate solutions is randomly created.
- The fitness of each individual in the population is calculated.
- New individuals are generated through selection, crossover, and mutation processes.
- The new population is evaluated.
- These steps are repeated until a predefined termination criterion is met.

Particle swarm optimization, developed by James Kennedy and Russell Eberhart in 1995 [21], is an optimization algorithm inspired by the collective behavior of natural swarms, such as bird flocks or shoals of fish. PSO employs a population-based approach to solve optimization problems and can be applied to both continuous and discrete problems.

The fundamental steps of the PSO algorithm are as follows:

- An initial population (swarm) is randomly generated.
- The fitness of each particle is evaluated by using a fitness function.
- The personal best (pbest) and global best (gbest) values are updated. Here, pbest refers to the best solution found by a particle until then, while gbest represents the best solution identified by the entire swarm.
- The velocity of each particle is updated. The position of each particle is adjusted based on its updated velocity.

- The process iterates until a termination condition is met, at which point gbest gives the best solution found by the swarm.

The ABC algorithm, a biological optimization method inspired by the foraging behavior of honey bees, was developed by Karaboga in 2005 [22]. The colony consists of three groups: employed bees, onlooker bees, and scout bees:

- Employed bees explore specific food sources (solutions) and share the information they gather with onlooker bees.
- Onlooker bees evaluate this information and decide which solution to explore based on its quality.
- If a food source (solution) cannot be improved within a certain time, scout bees abandon it and search for new solutions randomly.

Its simplicity, adaptability, and low parameter requirements make the algorithm suitable for various optimization problems, such as feature selection [23], function optimization, and data clustering.

## 3. Experimental results

This section presents the experimental results obtained to evaluate the effectiveness of the proposed approaches when applied to two distinct datasets. The performance evaluation was conducted using a range of standard metrics, including accuracy, precision, recall, and the F1-score. Among these, the F1-score was emphasized as the primary objective function for assessing model performance due to its ability to provide a balanced measure of a model's precision and recall. The mathematical definitions of the evaluation metrics employed in this study—accuracy, precision, recall, and F1-score—are detailed in Equations (1), (2), (3), and (4), respectively.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

$$Precision = \frac{TP}{TP+FP} \tag{2}$$

$$Recall = \frac{TP}{TP+ FN} \tag{3}$$

$$F1 - score = \frac{2*Precision*Recall}{Precision+Recall} \tag{4}$$

In Equations (1), (2), and (3), TP (True Positive), FP (False Positive), TN (True Negative), and FN (False Negative) denote the counts of respective classification outcomes. A TP represents instances correctly identified as belonging to the positive class, such as accurately detecting an attack in a cybersecurity system. FP, on the other hand, occur when instances are incorrectly classified as positive, such as mistakenly identifying normal behavior as an attack. TN refer to instances correctly identified as belonging to the negative class, such as accurately recognizing normal behavior as non-attack. Finally, FN occur when the model fails to identify positive instances and classifies them as negative, such as missing an actual attack.

The parameter values for the GA, PSO, ABC and k-NN algorithms were carefully configured to ensure optimal performance during the experimental studies. The selection of parameters was based on preliminary experimental results conducted during the development phase of the study. For greater clarity and readability of the study, the preliminary experimental results have not been included. Instead, only the parameter values obtained under the most optimal conditions during this process have been presented. These parameter settings are as follows:

- For GA, PSO and ABC the number of iterations was set to 20, and the population size was fixed at 50.
- In GA, the mutation rate was configured as 0.1, and single-point crossover was employed.

- For PSO, the inertia weight (w) was set to 0.7, while the cognitive and social coefficients (c1 and c2) were both set to 1.5.
- In k-NN, the Euclidean distance metric was used. During feature selection, the value of k was set to 50, while for classification, it was set to 100.
- Each experiment was conducted five times independently to ensure robustness and reliability of the results.

Table 1 provides a detailed presentation of the experimental results. The abbreviations used in the table are defined as follows: Algorithm is abbreviated as “Alg.”; Features as “Fea.”; Accuracy as “Acc.”; Precision as “Pre.”; and Recall as “Rec.”.

Upon examining Table 1, it is evident that the USB_IDS_1 dataset achieved the highest performance (0.98) using the PSO algorithm with 9 selected features. Similarly, the GA algorithm demonstrated comparable performance on the same dataset with 7 features. The ABC algorithm, achieving a success of 0.98 and selecting the 18 features, demonstrated relatively lower performance in comparison to the GA and PSO algorithms.

For the CIC-IDS2018 dataset, the PSO algorithm generally delivered consistent results with a larger number of features; however, a noticeable decline in metric values (0.94) was observed when only 4 features were used. On the other hand, the GA algorithm achieved its highest success (0.98) on this dataset with just 5 features. The ABC algorithm, while producing consistent results in terms of the number of selected features, was able to select a minimum of 18 features, as seen with the USB_IDS_1 dataset. However, although its performance is notable, it falls behind other algorithms with a success rate of 0.96.

These findings underscore the significant impact of feature selection on classification performance, demonstrating that high accuracy can be achieved even with a reduced feature set. Notably, the GA algorithm proved to be particularly effective in achieving high performance with fewer features, thereby offering a cost-effective approach for data processing. For both algorithms, the reduction of irrelevant features improved the accuracy of classification models and facilitated a more efficient analysis process.

**Table 1**

Experimental results

| *Run* | *Dataset* | *Alg.* | *Fea.* | *Acc.* | *Pre.* | *Rec.* | *F1* |
|---|---|---|---|---|---|---|---|
| 1 | USB_IDS_1 | PSO | 35 | 0.97 | 0.97 | 0.97 | 0.97 |
| 2 | | | 19 | 0.97 | 0.97 | 0.97 | 0.97 |
| 3 | | | 42 | 0.97 | 0.97 | 0.97 | 0.97 |
| 4 | | | 9 | 0.98 | 0.98 | 0.98 | 0.98 |
| 5 | | | 33 | 0.97 | 0.97 | 0.97 | 0.97 |
| 1 | USB_IDS_1 | GA | 9 | 0.97 | 0.97 | 0.97 | 0.97 |
| 2 | | | 10 | 0.97 | 0.97 | 0.97 | 0.97 |
| 3 | | | 7 | 0.98 | 0.98 | 0.98 | 0.98 |
| 4 | | | 14 | 0.97 | 0.97 | 0.97 | 0.97 |
| 5 | | | 5 | 0.97 | 0.97 | 0.97 | 0.97 |
| 1 | USB_IDS_1 | ABC | 21 | 0.97 | 0.97 | 0.97 | 0.97 |
| 2 | | | 23 | 0.98 | 0.98 | 0.98 | 0.98 |
| 3 | | | 18 | 0.98 | 0.98 | 0.98 | 0.98 |
| 4 | | | 22 | 0.98 | 0.98 | 0.98 | 0.98 |
| 5 | | | 23 | 0.98 | 0.98 | 0.98 | 0.98 |
| 1 | CIC-IDS2018 | PSO | 36 | 0.96 | 0.96 | 0.96 | 0.96 |
| 2 | | | 31 | 0.96 | 0.96 | 0.96 | 0.96 |
| 3 | | | 40 | 0.96 | 0.96 | 0.96 | 0.96 |
| 4 | | | 38 | 0.96 | 0.96 | 0.96 | 0.96 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 5 | | | 4 | 0.94 | 0.94 | 0.94 | 0.94 |
| 1 | CIC-IDS2018 | GA | 7 | 0.97 | 0.97 | 0.97 | 0.97 |
| 2 | | | 5 | 0.97 | 0.97 | 0.97 | 0.97 |
| 3 | | | 5 | 0.96 | 0.96 | 0.96 | 0.96 |
| 4 | | | 8 | 0.97 | 0.97 | 0.97 | 0.97 |
| 5 | | | 5 | 0.98 | 0.98 | 0.98 | 0.98 |
| 1 | CIC-IDS2018 | ABC | 19 | 0.98 | 0.98 | 0.98 | 0.98 |
| 2 | | | 18 | 0.96 | 0.96 | 0.96 | 0.96 |
| 3 | | | 19 | 0.97 | 0.97 | 0.97 | 0.97 |
| 4 | | | 20 | 0.98 | 0.98 | 0.98 | 0.98 |
| 5 | | | 20 | 0.98 | 0.98 | 0.98 | 0.98 |

## 4. Conclusion

In this study, feature selection was performed using GA, PSO and ABC on the USB_IDS_1 and CIC-IDS2018 datasets. Experiments were conducted with the feature sets obtained through these methods using the k-NN classifier.

The results demonstrate that feature selection significantly impacts classification performance. Achieving high performance with fewer features on both datasets highlights the efficiency of the algorithms employed and reveals that irrelevant features in the datasets can negatively affect classification success. For the USB_IDS_1 dataset, the PSO algorithm achieved the highest F1-score (98%) with 9 features, while the GA algorithm achieved the same score with 7 features. Regarding the CIC-IDS2018 dataset, the GA algorithm delivered the best performance (98%) with 5 features, while the PSO algorithm produced consistent results with a higher number of features. This indicates that both algorithms can flexibly and effectively operate across different datasets. The ABC algorithm has been more consistent in the number of selected features across both datasets compared to GA and PSO. However, it has produced similar results while using a larger number of features. The variability in the number of selected features across executions in GA and PSO indicates that sometimes these algorithms may be prone to getting stuck in local optima.

The findings confirm the critical role of feature selection in improving model accuracy and reducing computational costs in classification problems. In particular, the ability of the GA algorithm to achieve high performance with fewer features suggests promising cost-effective solutions for large datasets. Future research can expand the scope of these methods by employing more complex datasets and different classifiers, providing a more detailed analysis of the algorithms' performance across diverse problems.

## References

[1] M.V. Özsarı, Ş. Özsarı, A. Aydın, M.S. Güzel, USB-IDS-1 dataset feature reduction with genetic algorithm, Commun. Fac. Sci. Univ. Ank. Series A2-A3: Phys. Sci. and Eng. 66 No.1 (2024) pp.26-44.
doi: 10.33769/aupse.1320795.

[2] A.A. Aburomman, M.B.I. Reaz, Ensemble of binary SVM classifiers based on PCA and LDA feature extraction for intrusion detection, in Advanced Information Management, Communications, Electronic and Automation Control Conference (IMCEC). (2016) pp.636-640.

[3] O. Y. Al-Jarrah, Y. Al-Hammdi, P. D. Yoo, S. Muhaidat, M. Al-Qutayri, Semi-supervised multi-layered clustering model for intrusion detection, Digital Communications and Networks. 4 No.4 (2018) pp.277-286.

[4] W.L. Al-Yaseen, Z. A. Othman, M.Z.A. Nazri, A hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system, Expert Systems with Applications. 67 (2017) pp.296-303.

[5] X. An, J. Su, X. Lü, F. Lin, Hypergraph clustering model-based association analysis of DDOS attacks in fog computing intrusion detection system, EURASIP Journal on Wireless Communications and Networking. No.249 (2018) pp.1-9.

[6] M. C. Belavagi, B. Muniyal, Performance evaluation of supervised machine learning algorithms for intrusion detection, Procedia Computer Science. 89 (2016) pp.117-123.

[7] M. G. Raman, N. Somu, K. Kirthivasan, R. Liscano, V.S. Sriram, An efficient intrusion detection system based on hypergraph-genetic algorithm for parameter optimization and feature selection in support vector machine, Knowledge-Based Systems. 134 (2017) pp.1-12.

[8] H. Li, W. Guo, G. Wu, Y. Li, A RF-PSO based hybrid feature selection model in intrusion detection system, in 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC), Guangzhou, China. (2018) pp.795-802. doi: 10.1109/DSC.2018.00128.

[9] R. Vijayanand, D. Devaraj, A novel feature selection method using whale optimization algorithm and genetic operators for intrusion detection system in wireless mesh network, IEEE Access. 8 (2020) pp.56847-56854. doi: 10.1109/ACCESS.2020.2978035.

[10] R. Vijayanand, D. Devaraj, B. Kannapiran, Intrusion detection system for wireless mesh network using multiple support vector machine classifiers with genetic-algorithm-based feature selection, Computers & Security. 77 (2018) pp.304-314.

[11] J. Maldonado, M.-C. Riff, E. Montero, Improving attack detection of C4.5 using an evolutionary algorithm, in 2019 IEEE Congress on Evolutionary Computation (CEC), Wellington, New Zealand. (2019) pp.2229-2235. doi: 10.1109/CEC.2019.8790199.

[12] T. Dokeroglu, A. Deniz, H.E. Kiziloz, A comprehensive survey on recent metaheuristics for feature selection, Neurocomputing. 494 (2022) pp.269-296.

[13] M. Salati, İ. Askerzade, G. Bostancı, Convolutional neural network models using metaheuristic based feature selection method for intrusion detection, Journal of the Faculty of Engineering and Architecture of Gazi University. 40 No.1 (2024). doi: 10.17341/gazimmfd.1287186.

[14] M. Catillo, et al., USB-IDS-1: A public multilayer dataset of labeled network flows for IDS evaluation, in 2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), IEEE, (2021).

[15] I. Sharafaldin, A.H. Lashkari, A.A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization, in ICISSp. (2018) pp.108-116.

[16] R.I. Farhan, A.T. Maolood, N.F. Hassan, Hybrid feature selection approach to improve the deep neural network on new flow-based dataset for NIDS, Wasit Journal of Computer and Mathematics Science. (2021) pp.49-61.

[17] S. Alzughaibi, S. El Khediri, A cloud intrusion detection systems based on DNN using backpropagation and PSO on the CSE-CIC-IDS2018 dataset, Applied Sciences. 13 No.4 (2023) p.2276. doi: 10.3390/app13042276.

[18] Q.M. Alzubi, S.N. Makhadmeh, Y. Sanjalawe, Optimizing intrusion detection: Advanced feature selection and machine learning techniques using the CSE-CIC-IDS2018 dataset, Journal of Advances in Information Technology. 16 No.3 (2025).

[19] A. Srivastava, D. Sinha, PSO-ACO-based bi-phase lightweight intrusion detection system combined with GA optimized ensemble classifiers, Cluster Comput. 27 (2024) pp.14835-14890.

doi: 10.1007/s10586-024-04673-3.

[20] J.H. Holland, Genetic algorithms, Scientific American. 267 No.1 (1992) pp.66-73.

[21] J. Kennedy, R. Eberhart, Particle swarm optimization, in Proceedings of ICNN'95-International Conference on Neural Networks. 4 (1995) pp.1942-1948.

[22] D. Karaboga, An idea based on honey bee swarm for numerical optimization. (2005) pp.1-10.

[23] Y. Lin, J. Wang, X. Li, Y. Zhang, S. Huang, An improved artificial bee colony for feature selection in QSAR, Algorithms. 14 (2021) pp.120. doi: https://doi.org/10.3390/a14040120