Arabian Journal for Science and Engineering

**The name(s) of the author(s): Tahira Khorram, Nurdan Baykan**

**A concise and informative title: Feature Selection in Network Intrusion Detection using Metaheuristic Algorithms**

**The affiliation(s) and address(es) of the author(s): Faculty of Engineering, Selcuk University, Selcuklu, Konya, Turkey**

**The e-mail address, and telephone number(s) of the corresponding author: , +905349655008**

**If available, the 16-digit ORCID of the author(s):**

**Abstract:**

Network intrusion detection mechanism is a primary requirement in the current fast growing network systems. Data mining and machine learning approaches are widely used for network anomaly detection during past few years. Machine learning based intrusive activity detector is getting popular, But still, they produce a high volume of false alarms. One of the main reasons for generating false signals is the usage of datasets with redundancy. To resolve this problem, efficient feature selection is necessary to improve the intrusion detection system performance. In this article, we use Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), and K-Nearest Neighbors. The three mentioned metaheuristic algorithms are used to select the most relevant feature set for identify network attacks. And KNN algorithm is used as the classifier. The standard NSL-KDD dataset is used as training set and testing set in this experiment. We used different performance metrics to find out which of the mentioned algorithms provide a better overall performance when they are used for feature selection in intrusion detection. Our experiments show that all these three algorithms for attribute selection perform better than other approaches. Feature selection based on ABC provides a 98.6 accuracy rate with a false alarm of 1.2, which is the best result among the examined algorithms.

**Keywords**

*Intrusion Detection System, Feature Selection, Ant Colony Optimization, Particle Swarm Optimization, Artificial Bee Colony*

# Introduction

Recently the internet and computer networks have become the inseparable part of everyday life. A statistic until 2017 shows that there are 20.35 billion devices connected to the internet all over the world and this number will be increased up to 31.73 billion devices through 2020 [1]. By connecting more devices to the computer systems, the risk of unauthorized activities such data destruction, data modification, and data theft from both internal intruders and external intruders will be increased.

Several types of security appliances and protocols are designed to protect our distributed systems from a variety of internet attacks. Firewalls, Intrusion Detection System (IDS), and Intrusion Prevention System (IPS) are the most widely used appliances. In this study, our focus is on IDS. Intrusions are a set of actions that try to overrule the security aspect of a system and violate the confidentiality, integrity, and availability of that computer network [2]. Intruders always try to find a vulnerability in the system to launch an attack; it is intrusion detection system that monitors and analyzes all events happening on the computer network, identifies intrusive activities and searches for a sign of security problem [3]. In case of abnormal behavior, or an attack, IDS sends an alarm to the system administrator to react immediately before the intrusion affects the network. IDS can be deployed a network-based to monitor all network events or can be set up on a PC as host-based to audit all incidents happening on that PC [4]. Anomaly-based detection and misuse-based detection are two standard approaches to network anomaly detector. Misuse-based works based on signature generates an alarm when an intrusive activity matches the signature. Unlike misuse detection, anomaly-based warns system admins when there is an event deviating from the normal behavior of the system., and thus it is capable of detecting unknown attacks [4,5]. IDS is used to protect a computer network system. A secure network system is defined as a barrier, which prevents violations of availability, confidentiality, and integrity of information and resources [6].

Researchers presume that anomaly traffic behavior differs from regular traffic and unknown network traffic patterns are similar to known traffic instances. Based on the mentioned hypothesis intrusion detection can be considered as data analysis problem [6]. To specify features for network traffic records, Data Mining and Machine Learning (DM-ML) algorithms are widely used. DM-ML algorithms help to define samples for intrusive traffic and regular traffic. Due to the massive amount of network data produced daily, determining related and useful patterns of data is a difficult task. Generated datasets are usually noisy and contain unnecessary and correlated features that confuse the intrusion classifier engine and reduce the overall performance of the system [7]. Therefore, feature selection is a major issue in detecting network anomalies. Feature space that is fed to an intrusion classifier as training examples have a significant impact on the system performance [7]. Thus, plenty of works had been done on the selection of the right, and related features for network traffic records to boost the performance of the IDS and reduce the computational cost. A large number of that literature focus on heuristic searches for selecting the right features. In the recent approaches, metaheuristic algorithms such as Particle Swarm, Ant Colony, and Artificial Bee Colony algorithms are used for feature selection and classification of intrusions.

In this paper, we aim to use these three metaheuristic algorithms for feature selection, and KNN as an evaluator for the selecting of features by metaheuristic algorithms. We seek to discover the most efficient algorithm for feature space selection of intrusive detector system, and we will show how feature selection improves the overall performance of the intrusion detection system. This paper is structured as follows. In section 2 we provide brief information on methods and materials used in this research. Part 3 is result analysis and discussion and in section 4 we conclude the paper.

# Experimetal Methods

**A.** **Related Works**

In detecting network anomalies, an accurate dataset leads to optimal performance. Feature space of an intrusion classifier needs to be preprocessed. To preprocess and remove redundant and unnecessary features that degrade the performance, several methods have been proposed. Here we review some of those works.

In [8], a different intrusion detector is proposed by applying ant colony optimization for feature selection, and SVM algorithm for classification. In this study, they mapped the feature into a connected graph, in which each ant can select one feature. The selection of the next feature depends on pheromone value and heuristic information. The proposed method applied to a KDD99 dataset which includes 10,000 data records. The overall performance of the proposed technique is 97.7%. In [9], an ant colony algorithm is applied to the KDD99 dataset as feature selector, it selects 14 features among 41 features, and SVM is used as a detection method. 5,823 records of the KDD99 dataset is used for training and 77,287 used for testing in this binary classification problem. The experimental result shows 98.5% of detection accuracy with ACO-SVM and 98.2% with SVM. Aghdam et al. [10], presented ACO as feature selection and select 19 features of the KDD99 dataset and NSL-KDD datasets. The experimental result shows 98.9% detection accuracy.

A hybrid method using multi-objective particle swarm optimization and the random forest is recommended to detect Probe attack. Their objective is to improve detection rate and decrease the false alarm discovery rate while identifying Probe attacks. PSO eliminates the unnecessary features, and RF detects the Probe attacks. The detection rate of this proposed method is 90.7% [11]. Ahmad [7], used PCA and PSO for feature reduction. He used PCA to reduce features. The features are selected based on their eigenvalues. It is not guaranteed that attribute with higher eigenvalue provides optimal sensitivity for the classifier. To make sure that optimal features are chosen for the intrusion detector the author proposes optimization methods. He used PSO and for optimization to improve the performance of PCA. Artificial Neural Network used for detection. The proposed method detection rate upgraded from 94.50 to 99.40 and false alarm reduced from 5.5 to 0.6. Srinoy in [12], proposed PSO for feature reduction and SVM for intrusion detection.

In [13], the authors introduced Artificial Bee Colony algorithm for intrusion classification. They used the CART and BNMB algorithms to select the most relevant features for classifier from the KDD99 dataset.  By the proposed method, the accuracy rate of 97% is achieved for the known attack, and for unknown attacks, 93.25% accuracy has been measured. Ghanem et al. [14] use a new approach to design an efficient intrusive detector tool. This method uses multiobjective ABC algorithm to minimize the number features of IDS dataset. Then PSO-ABC optimized NN is used to determine intrusive packets and standard traffic packets. In [15], ABC and SVM are used to design an IDS. There ABC algorithm is used for two purposes; First it is used to select the necessary features from the KDD99 dataset, and secondly, it is used to optimize the SVM parameters. For anomaly detection, ABC-SVM is proposed. The overall performance of this method is higher than PSO and GA-SVM at the same time.

**B.** **Particle Swarm Optimization (PSO)**

PSO is a population-based optimization method developed by Dr. Eberhard and Dr. Kennedy in 1995. PSO inspired by a scenario where a group of birds is searching for a piece of food in a specific area, none of the birds know where the food is but in each iteration the birds know how far the food is. The optimal way to find the food is to follow the nearest bird to the food. [16]

 In PSO, the birds called particles. Each particle of the group keeps track of their attributes. These attributes include particle’s current position, particle’s current velocity that keeps track of speed and direction in which the particle is flying. Each particle has fitness value that is obtained by calculating the fitness function at particle’s current position [16].

PSO algorithms work as follows:

*Step1:* initialize population. Particle’s position and velocity initialized randomly.

*Step 2:* evaluate each particles fitness value

          If fitness *X > pbest*

            then *pbest = X*

 if fitness *X > gbest*

 then *gbest = X*

*Step 3:* update the velocity and position of particle i

|  |  |
| --- | --- |
|                                                                                     :$v\_{i}d^{(}t+1)=w\*v\_{i}d^{t}+c\_{1}\*r\_{1}i\*(p\_{i}d−x\_{i}d^{t})+c\_{2}\*〖r〗\_{2}i\*(p\_{g}d−x\_{i}d^{t}).$ | (1) |

   Update the position of particle i.

|  |  |
| --- | --- |
|  | (2) |

Where  shows the th iteration of PSO,  indicates search space dimension, w is inirtia wieght and and  are accelation factors,    and  are random numbers between (0,1).  and  are pbest and gbest.

*Step 4:* return to step 2 and three if stopping criterion (max iteration) has not been met.

*Step 5:* return global best value. (here is the selected feature).

 For the experiment, first, we set the number of particles and number of generations. Each particle represents a feature in the swarm search space. Each particle is initialized by random 1 or 0 values. The feature with value 1 is selected, and feature with 0 value will be eliminated. Thus, every particle illustrates a different subset of features. The particles are randomly initialized and then start moving in the search space to search for the best subset of features by updating its position and velocity. For example out of 41 features, 10 attributes will be selected, this selected feature set might include attributes: x1, x4, x5, x6, x12, x25, x26, x29, x30, x33, x37. So after any generation, a particle might look like (1,1,1,0,0,1,0,1,1,0,1). As we mentioned before, value 1 indicates selected attributes while 0 shows ignored attributes. Now we can say that attributes selected by this particles are x1,x4,x5,x25,x29,x30,x37. In the next generation, because of the pbest and gbest of the other particles, this particle’s position will be changed, and this time it will select a different set of attributes among these 11 attributes. The dimension of the particles will be updated according to the equation (1) and (2).

The KNN classification algorithm will be used for classification of attacks. The PSO selected features will be used for detecting attacks.

The PSO sequence for feature selection is a follows:

|  |
| --- |
|  |
| Figure1: PSO algorithm for feature selection |

**C.** **Ant Colony Optimization  (ACO)**

ACO, which is developed by Dorigo in 1992, has been inspired by social behavior of ants searching for a food source all together in a group. The ants in the swarm communicate along with a chemical matter called pheromone. Ants spray pheromone on their route to food origin. Each ant can follow the path marked by pheromone. The pheromone concentration differs from the primary random direction. Ants always follow the roads with higher pheromone concentration as the number of ants increases on that specific route the number of pheromones also will be expanded, and that route will be selected as an efficient direction to the food source [17, 18]

For the experiment, we set up the initial population of ants, some generations, and initial pheromone value for each feature (ant). The generated ants placed randomly on a graph node. Each ant represents a random feature. All ants start to move to the next feature. For ant  that is placed randomly on the node (feature) *i*.  the probability of selecting next feature  *j* is:

|  |  |
| --- | --- |
|    | (3) |

Where is symbol for heuristic information (number of time a feature has been visited), *j* is the set of neighbor nodes that has not been visited by ant k yet.  And  parameters determine the amount pheromone with the respect to the heuristic information. After every generation, the amount of local pheromone value is updated:

|  |  |
| --- | --- |
|  | (4) |

Where

*N* is the set of visited neighbors’ nodes for that process.  control . After completing tour, the ants pass their selected features to the classifier. The dataset that produce by ants get global pheromone update by the following equation:

|  |  |
| --- | --- |
|  | (5) |

Where

All these steps repeated until the termination condition (max iteration or until the accuracy cannot improve more) is met. Figure 2 shows the flow of ant feature selection.

|  |
| --- |
|  |
| Figure2: Ant colony feature selection |

**D.** **Artificial Bee Colony  (ABC)**

ABC is a swarm-based optimization method that was developed by Karaboga in 2005. The algorithms are inspired by the social behavior of honeybees [19]. The algorithm consist of these components:

·        Employed bees: They find a food source, store information about the quality of the food and share the information with others.

·        Unemployed bee: they are two types: Onlooker bees receive information about food source and choose the food source with higher quality. The other type is scout bees when the existing food is over; scout bees try to find new food sources [20].

In our experiment, we use ABC to select the best features for KNN classifier. The algorithm is initialized by N feature vector, where the elements are placed in different positions of the vector. If the value of the vector at that position is 1 that attribute will be selected otherwise the attribute will be dropped. Equation 6 shows the bee swarm initialization:

|  |  |
| --- | --- |
|  | (6) |

Where *i*=1,2…N, is the number of features (food sources), *j*=1,.., D, is some optimization parameters.

As we mentioned the employee bee, search for food sources in the neighborhood. This exploration is defined in equation 7:

|  |  |
| --- | --- |
|  | (7) |

For each , there a food source  is determined.  is a number between (-1,1), *j* and *k* are random variables. After  is produced we can obtain the fitness value for the feature (food origin)

|  |  |
| --- | --- |
|  | (8) |

 is the fitness function. As we mentioned after employed bees have found the food source they will share the information about the food source and its quality with the onlooker bees the probability of selecting that food source (feature) by onlooker bees is shown in equation 9.

|  |  |
| --- | --- |
|  | (9) |

A general flow for ABC is shown in figure 3

|  |
| --- |
|  |
| Figure3: Bee Colony general flow |

**E.** **K-Nearest Neighbors (KNN)**

KNN is one of supervised machine learning algorithms that are very simple to understand and implement. It works based on minimum distance of the new instance from the training sample that determine the nearest neighbors [21]. After nearest neighbors are gathered, the majority vote of the neighbors decides what the new instance belongs to. For example, if k=5, this algorithm will look to five nearest neighbors and find out the class of the new instance. KNN classifier works based on Euclidean distance, which is the distance between the test sample and a specific training sample [22]. Below equation calculates the ED.

|  |  |
| --- | --- |
|  | (10) |

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus in sagittis nisi. Aenean tristique cursus lorem, sed pretium mauris scelerisque vel. Morbi aliquet ultrices lorem quis sodales. Praesent elementum felis quis quam ultrices, quis mollis urna interdum. Nam imperdiet eget ex molestie dapibus. Suspendisse consectetur felis venenatis turpis scelerisque placerat. Mauris sodales libero eget libero faucibus, id porttitor ante aliquam. Curabitur molestie mi vel augue scelerisque aliquam. Pellentesque vel nibh pellentesque, ullamcorper lorem a, pharetra justo.Nam a erat sit amet odio ultrices porttitor. Nulla eget luctus leo. Vivamus ullamcorper facilisis tortor, ut cursus velit consequat facilisis. Maecenas posuere odio eget sem lobortis euismod. Donec vulputate scelerisque dapibus. Nam nisl purus, pulvinar vel lacus et, accumsan venenatis enim. Nullam et erat massa. Aliquam sit amet justo mollis, feugiat enim id, gravida lectus. Etiam dolor diam, tincidunt vitae pulvinar in, consectetur quis ante. Suspendisse venenatis ut nisi a sagittis. Maecenas consectetur sem non ex rhoncus efficitur. Morbi urna tellus, imperdiet sed finibus vitae, faucibus quis sem. Morbi gravida, lorem cursus faucibus pretium, metus erat sodales arcu, sit amet tempor ex nisl at metus. Vivamus sem eros, egestas et metus eget, elementum eleifend ligula.

# Acknowledgements

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus in sagittis nisi. Aenean tristique cursus lorem, sed pretium mauris scelerisque vel. Morbi aliquet ultrices lorem quis sodales.