International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 05, 2020
ISSN: 1475-7192

HYPER META-HEURISTIC IMPROVED PARTICLE SWARM OPTIMIZATION BASED SUPPORT VECTOR MACHINE (HMHIPSO-SVM) FOR BIG DATA CYBER SECURITY

[1]Mrs.G.A. Mylavathi, [2]Dr.B.Srinivasan

***ABSTRACT**--* *Cyber security in the domain of big data is regarded to be a crucial challenge for the research community. Machine Learning (ML) algorithms are recommended to be the representatives for dealing with big data security issues. Generally, Support Vector Machine (SVM) has been found successful for different problems of classification. The user has to specify the right SVM configuration much earlier, which again is cumbersome, needing the help of skilled experts in choosing the kernel function and a excessive amount of manual labor for experiments. In order to resolve this problem, a HMHIPSO-SVM classifier is proposed to be designed, whose performance is necessary for dealing with the right choice of the low-level heuristic. Hyper Meta-Heuristic Improved Particle Swarm Optimization based Support Vector Machine (HMHIPSO-SVM) framework comprises of a high-level mechanism and low-level heuristics. The high-level mechanism employs the search performance for controlling which low-level heuristic has to be utilized for the SVM configuration generation. It is designed with the aim of optimizing the SVM multi-objective optimization problem by including the hyper meta-heuristic and Improved Particle Swarm Optimization (IPSO) algorithm. The process of SVM configuration is designed as the multi-objective optimization problem by taking the false positive (fp), false negative (fn), true positive (tp) and true negative (tn) into consideration in which the results are attained based on metrics such as precision, recall, f-measure and accuracy, and here the model complexity is found to be the two contradicting objectives. The efficiency of the HMHIPSO-SVM framework has been assessed on two cyber security challenges, (1) malware big data classification, and then (2) anomaly intrusion detection. The novel framework shows improved classification performance of big data cyber security problems in comparison with other available algorithms.*

***KEYWORDS**--* *Machine Learning (ML), Support Vector Machine (SVM), Improved Particle Swarm Optimization (IPSO), Hyper Meta-Heuristic Improved Particle Swarm Optimization based Support Vector Machine (HMHIPSO-SVM), Big data, Cyber security.*

I. INTRODUCTION

Technologies have taken a big leap and networking concepts like mobile, social and Internet of Things generate digital information in excessive levels. In this regard, the term big data is used for describing these extreme amounts of digital information. Big data specifies huge and complicated datasets with both structured and unstructured data

[1] Assistant Professor of Computer ScienceGobi Arts & Science College, Gobichettipalayam, Mailid: mylavathiga@gmailcom

[2] Associate Professor of Computer Science Gobi Arts & Science College, Gobichettipalayam, Mailid: srinivasangasc4393@gmail.com

Received: 02 March 2020 | Revised: 21 April 2020 | Accepted:05 May 2020

created everyday, and their analysis has to be done in brief time duration. The term big data is diverse from the big database, where big data implies that the data is too vast, too quick, or too difficult for the available tools to deal with. Big data is a domain, which finds means for the analysis, systematic extraction of information from, or the management with data sets, which are too big or complicated to be worked with classical data-processing application software[1].

Data with several cases (rows) exhibits higher statistical potential, while data having a greater complexity (more attributes or columns) may result in the false discovery rate becoming high. The challenges of Big data are data capture, data storage, data analysis, search process, sharing, transfer, visualization, querying, updating, information privacy and data source. The important difference between conventional and big data lies with regard to volume, velocity and variation. Volume implies the amount of data generated; velocity indicates the speed with which the data gets generated and variation implies the kinds of structured and non structured data. The term big data refers to the enormous amount information, which is stored and transferred in a computer system [2].

Typically, Big data is defined using three features, which are volume, variety and velocity (3Vs): The complexity and problems faced in big data are primarily owing to the extension of all three features (3Vs) - and not just the volume [3].

1. The amount of data (Volume) - Size: the volume of datasets is a crucial aspect, which indicates the amount of data generated.
2. The rate of data generation and transmission (Velocity) - Complexity: the structure, behaviour and permutations of datasets is an important factor.
3. The kinds of structured and unstructured data (Variety) - Technologies: tools and approaches, which are utilized for processing a reasonably sized or complicated datasets is a crittical factor.

Big data has given rise to a new problem associated with not just the 3Vs characteristics, but also it is data security related. It has been shown that big data not just expands the level of the challenges associated with security, but also leads to new and diverse cyber-security dangers, which have to be dealt efficiently and smartly. Rather, security is considered to be the primary challenge for any organisation during big data learning. Big Data analytics tools can be utilized for detecting the threats in cyber security, which include malware/ransom ware attacks, endangered and weak devices, and adversarial insider programs [4-5]. Big data cyber-security problems include malwares detection, authentications and stegano analysis. Among these problems, malware detection is found to be the most important problem in big data cyber-security. Malware takes different names like malicious software, malicious code and malcode. Malware detection techniques can be vastly classified into two important groups: signature-based and anomaly based. Signature based detection is based on pre determined signatures to determine the malicious nature of a suspicious program. Anomaly-based detection determines the malicious behavior of a suspicious program depending on its advance information of what defines a normal behavior. Specification-based or rule-based detection is a branch of anomaly-based detection. Researchers and companies are aware of the problems, which can result due to this intrusive software (malicious computer programs) and hence novel techniques has to be designed to avert them [6-7].

Data mining exhibits fine performance in the form of a tool for malware detection. It permits the analysis of massive sets of information and extraction of new information out of it. The important advantage of making use

of data mining approaches for the detection of malicious software lies in the capability of identifying both known and zero-day attacks. In order to deal with the above challenges, Machine Learning (ML) algorithms have been introduced for the classification of unnamed patterns and malicious software [8]. ML has demonstrated potential results in the classification and identification of unnamed malware software. Support Vector Machine (SVM) is one of the most common ML algorithms and has been extremely successful in different practical applications. The widespread ubiquity of SVM is owing to their better performance and scalability. But, in spite of these benefits, the performance of an SVM hugely suffers due to its preferred configuration [9-10]. A common SVM configuration consists of selecting the soft margin parameter (or penalty) and the type of kernel and its parameters.

The existing optimisation techniques are grid search techniques, gradient-based techniques and meta-heuristic techniques. Grid search refers to a process, which carries out an exhaustive search through a manually defined subset of the hyperparameter space of the referenced algorithm [11]. Gradient-based techniques are iterative approaches, which make an extensive use of the gradient information of the objective function during iterations. Its important drawbacks are that they need the objective function to be unique and that they are highly dependent on the initial point [12]. Metaheuristic is a higher-level process or heuristic developed to seek, generate, or choose a heuristic (partial search algorithm), which may yield an adequately good solution to an optimization problem, particularly with missing or incorrect information. The performance of a meta-heuristic technique is hugely dependent on the parameters and operators chosen, the selection process of which is found to be a very hard and time-taking. Moreover, jusst one kernel is utilized in most of the works, and the search is carried out over the parameter space of that particular kernel [13].

This research work introduces a hyper meta-heuristic framework for SVM configuration optimisation. Hyper meta-heuristic exhibit more efficiency compared to other techniques as they are not dependent on the specific current task and can frequently get high potential configurations. The newly introduced HMHIPSO-SVM framework combines different key components, which distinguish it from the available works to get an efficient SVM configuration for big data cyber security. HMHIPSO-SVM framework comprises of a high-level mechanism and low-level heuristics. The high-level mechanism makes use of the search performance to control the choice of the low-level heuristic that needs to be utilized for the SVM configuration generation. The novel Framework comprises of three phases, which are explained in Section 3. At first, the framework takes a multi-objective definition of the SVM configuration problem into account, in which the accuracy and model complexity are considered to be two contrasting goals. Secondly, the framework manages the choice of the type of the kernel and kernel parameters and also the soft margin parameter. It is designed with the aim of optimizing the SVM multi-objective optimization problem by including the hyper meta-heuristic and Improved Particle Swarm Optimization (IPSO) algorithm. Thirdly, the hyper meta-heuristic framework integrates the potential of decomposition and Pareto-based techniques adaptively to get an approximate Pareto set of SVM configurations. The performance of the novel framework is assessed and then compared with that of benchmarked algorithms on two cyber security issues, which are malware big data classification and anomaly intrusion detection. The remaining sections of this work is organised as given. Section 2 provides a summary of relevant work. The specification and development of SVM and IPSO are studied, the hyper meta-heuristic framework and its important components are also discussed in section 3. Section 4 discusses the computational results of HMHIPSO-SVM framework and the framework is compared with other available algorithms. At last, Section 5 provides the conclusion and future work of this work.

II. LITERATURE REVIEW

Wu et al [14] proposed the big data analysis-based secure cluster management architecture for the optimized control plane, an Ant Colony Optimization (ACO). A security authentication mechanism is introduced for cluster management. Also, the ACO technique facilitates the big data analysis approach and the implementation system, which helps in optimizing the control plane. Security cluster management comprises of three steps, (i) Secure Authentication for Cluster Control, (ii) Ant System Based Modeling for Cluster Management, and finally (iii) Ant Colony Optimization Algorithm Based Cluster. Firstl, the username/password token for this authentication, a hash function and a Message Authentication Code (MAC) are submitted to increase the security during the clustering process. There are two stages in the secure authentication for clustering in control plane, which include "credential generation" and "credential authentication". The reduced computation in hash minimizes the complexity of the authentication protocol. Truly, the hash based MAC is a successful cryptology algorithm. Then, ant colony algorithm is a practical and popular approach for data analysis. But, implementation of the cluster of the SDN control plane is still an open challenge. This section studies the adaptation of the ant colony algorithm for the cluster of SDN control plane. At last, SDN clustering in the control plane, data objects (e.g., SDN traffic) are considered to be the ants with different attributes, and the clustering master node is considered to be the food source. The elaborate clustering process is about imitating the foraging behavior of ants. A secure authentication approach was introduced to guarantee the credibility of the data sources. The newly introduced work is successful in the performance and efficiency improvement of applications that run in SDN.

Terzi et al [15] introduced a NetFlow protocol for network anomaly and attack detection analysis on big data. NetFlow is basically a network protocol, which gathers traffic information like network users, network applications, and routing traffic. A public big network data was evaluated with a novel unsupervised anomaly detection mechanism on Apache Spark cluster in Azure HD Insight. At last, the results acquired from a case study were assessed, and and accuracy level of 96% was attained. The results were examined after dimension reduction employing Principal Component Analysis (PCA). It is evident from the results and the literature that timed and efficient detection of anomalies are required for improving the network security. A greater accuracy in anomaly detection yields superior quality of services and communication even when there is an increase in the complexity of attacks and analysis process. The detected anomalies may yield helpful outputs to know about the behavior of the network, differentiating between the attacks, rendering improved cyber security, and securing crucial architectures.

Kiss et al [16] introduced a clustering based mechanism for cyber attacks detection, which result in anomalies in Networked Critical Infrastructures (NCI). Different clustering approaches are examined to select the most desirable one for clustering the time-series data features, thereby categorizing the states and probable cyber on to the physical system. The Hadoop implementation of Map Reduce paradigm is helpful in rending an appropriate processing environment for massive datasets. A case study on a NCI comprising of several gas compressor stations is introduced. The design engineer of this kind of security modules has a much deep understanding on the processes of the cyber-physical system. However, the obvious benefit of this mechanism is that it is quite effective in strengthening the overall security of the deployment, as the same objective is applicable to the attacker also. Hence,

an attacker first needs to obtain access to the cyber system and has to stay sneaky for a specific time period in order to understand the particular features of the physical process.

Teoh et al [17] studied about Hidden Markov Models (HMM) for predicting the security attacks waged on big data. HMM in cyber security observed till date is in countable number. The characteristics of HMM such as prediction, probabilistic nature, and its capability of modelingvarious normally happening states is a good foundation for cyber security data modeling. Therefore, it becomes the inspiration for this research work to yield the initial results of efforts made towards the prediction of security attacks employing HMM. A massive network of datasets that represent cyber security attacks have been utilized for developing an expert system. The features of attacker's IP addresses can be obtained from combined datasets for the statistical datageneration. The cyber security expert yields the weight of every attribute and establishes a scoring system through the log historyannotation. In addition, HMM is used for distinguishing between a cyber security attack, unknown and no attack by first splitting the data into 3 cluster employing Fuzzy K mean (FKM), then manually labelling a small data (Analyst Intuition) and and later using HMM state-based mechanism. As per this, promising results are obtained in comparison with finding the anomaly in a cyber security log, which usually produces a massive amount of false detection data.

Teoh et al [18] suggested a Fuzzy K Mean (FKM) and Multi-Layer Perceptron (MLP) security monitoring system that can be used for big data analysis malware attack detection. It helps in monitoring the big network datasets that this process generates. A massive network datasets with characteristic malware attacks have been utilized for forming an expert system. The features of attacker's IP addresses can be obtained from ensemble datasets for statistical data generation. The cyber security expert gives to the weight of every attribute and creates a scoring system through the log history annotation. A specific semi supervise technique is adopted for classifying the cyber security log into attack, unknown and no attack by first dividing the data into 3 cluster employing FKM, then manually labelling a small data (Analyst Intuition) and at last, training the neural network classifier MLP base on the manually labelled data. The results achieved is quite promising in comparison with getting the anomaly in a cyber security log, which usually leads to the generation of massive amount of false detection. The technique of incorporating Artificial Intelligence (AI) and Analyst Intuition (AI) is also called as AI2. The classification results exhibit much potential in classifying the kinds of attacks.

Liu et al [19] suggested a framework of cyber security condition awareness depending on data mining. The framework can be visualized from two aspects, one involves data flow, which provides the abstraction of of cyber data, and the other one involves the logic view, which demonstrates the process of situation knowledge. The core component of the frame work is correlation state machine, an extended version of state machine. The correlation state machine is basically a data structure for attaining situation knowledge, which is established depending on the data mining technology. Once it is formed, it can be utilized to evaluating and predicting the threat conditions to attain cyber information. In addition, it is concluded with an example of the way in which the framework can be used for practical purposes to yield cyber security condition for administrators.

Ruan et al [20] introduced a hash algorithm, a weight table, and sampling technique to tackle with the intrinsic issues resulting from the analysis of big data, which include volume, variety, and velocity of the KDD99 data set for big data security. Three key aspects summarize this concept: the new sampling technique, evidence of the visualization algorithm in the KDD99 data set, and the concept of reducing the data points within a figure rather

than their inclusion. The benefit of proposed sampling technique is the control of the intrinsic faults present in big data sets comprise in terms of the volume, variation and velocity. The sampling technique helps in considerably reducing the time consumed as a measure towards velocity. By using a visualization algorithm, it was possible to obtain the perspectives into the KDD99 data set with a right detection of "normal" clusters and defined unique clusters of efficient attacks.

Sabar et al [21] introduced a Hyper-Heuristic Support Vector Machine (HH-SVM) for attaining big data cyber security. The new hyper-heuristic framework comprises of a high-level mechanism and low-level heuristics. The high-level mechanism makes use of the search performance for controlling the selection and low-level heuristic has to be utilized for the generation of a novel SVM configuration. Each low-level heuristics uses multiple rules for efficiently exploring the SVM configuration search space. In order to deal with bi-objective optimisation, the novel framework intuitively combines the potential of decomposition- and Pareto-based techniques for the approximation of the Pareto set of SVM configurations. The efficiency of the novel framework has been assessed on two cyber security issues: Microsoft malware big data classification and anomaly intrusion detection. The results attained show that the novel framework is quite efficient, if not the best, in comparison with its contemporaries and other algorithms.

III. PROPOSED METHODOLOGY

The primary objective of the proposed approach is to generate and develop a classifier with the objective of dealing with cyber security issues in big data. The proposed classifier uses a particular search performance in the aspect of control choice in which low-level heuristic has been utilized for creating the Support Vector Machine (SVM) configuration. HMHIPSO-SVM framework combines various key components, which distinguish it from the available works to get an useful SVM configuration for achieving big data cyber security.

The novel framework comprises of three phases as explained. (i) HMHIPSO-SVM framework takes a multi-objective derivation of the SVM configuration problem into the consideration, with the accuracy and model complexity as parameters, and (ii) framework regulates the choice of type of kernel and kernel parameters in addition to the soft margin parameter, then SVM multi-objective optimization problem is solved by including hyper meta-heuristic and Improved Particle Swarm Optimization (IPSO) algorithm, then (iii) hyper meta-heuristic framework integrates the merits of decomposition and Pareto-based techniques adaptively to get an approximated Pareto set of SVM configurations.

A. Support Vector Machine (SVM)

Support Vector Machine (SVM) come under the category of supervised learning techniques utilized for classification and regression tasks, which is founded from statistical learning theory. In the form of a classification technique, SVM is a universal classification model, which produces non-overlapping partitions and generally use a all the attributes. Linear Classification, SVM can effectively carry out a non-linear classification process using a strategy known as the kernel trick, inherently mapping their inputs onto high-dimensional feature spaces. If data is unnamed, supervised learning cannot be carried out, and an unsupervised learning mechanism is needed, which tries to get the natural clustering of the data onto groups, and then the new data are mapped onto these new groups.

The support-vector clustering algorithm, formulated by Hava Siegelmann and Vladimir Vapnik, uses the statistics of support vectors, designed in the support vector machine algorithm, for classifying the unlabeled data, and is one of the most extensively applied clustering algorithms in industrial applications [22].

Classification of data is a generic task in machine learning. Assume, few data points which belong to either of the two classes, and the objective is to determine which class a new data point will belong to. In support-vector machines, a data point is considered to be a p-dimensional vector (a list made up of p numbers), to know if can such points can be separated with a $(p-1)$ −dimensional hyperplane. This is known as a linear classifier. Several hyperplanes exists,which might help in the data classification. One possible selection for the best hyperplane is the one, which characterizes the biggest separation, or margin, between the two classes. Therefore, the hyperplane is chosen, such that the distance from it to the closest data point on every side is increased. If there is such a hyperplane , it is called as the maximum-margin hyperplane and the linear classifier defined by it is called as a maximum-margin classifier; or in other terms, the perceptron of optimum stability.

SVM builds a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be utilized for classification, regression, or other tasks such as outliers detection. Generally, a good isolation is attained using the hyperplane, which has the greatest distance to the closest training-data point of any class (so-known functional margin), as typically the greater the margin, the lesser will be the generalization error of the classifier. SVM approaches are designed with the intent that dot products of pairs of input data vectors may be computed withe ease in terms of the variables in the real space, by expressing them in terms of a kernel function $k(x, y)$chosen for the problem.The hyperplanes in the higher-dimensional space are specified to be the set of points whose dot product with a vector in that space is fixed, where this set of vectors is an orthogonal (and therefore minimal) set of vectors, which is characteristic of a hyperplane. The vectors that define the hyperplanes can be selected to be linear combinations with parameters $\alpha_i$of images of feature vectors $x_i$occurring in the data base.With this selection of a hyperplane, and the points x in the feature space, which are mapped onto the hyperplane are defined by the relation $\sum_i \alpha_i k(x_i, x) = constant$.It is to be noted that if $k(x, y)$tends to become small with y growing further away from x, and every term in the sum provides a measure the extent of closeness of the test point x to the respective data base point x_i. In this manner, the sum of kernels given above can be utilized for measuring the relative closeness of every test point to the data points that originate in one or the other of the sets that have to be differentiated. Also, another fact also has to be noted that the set of points x mapped onto any hyperplane can be highly convoluted and consequently, allows much more sophisticated discriminate the sets, which are not convex at all in the real space [23].

Several researchers have designed novel and hybrid kernel functions by merging the elementary kernels. The available kernel functions can be categorized to be either local or global kernel functions. The local kernel functions exhibit a superior learning capability, though with no remarkable generalisation capability; on the contrary, global kernel functions exhibit good generalisation ability but their learning capability is poor [24].

The important problem is in deciding the kernel function that has to be utilized for the problem instance at hand or the decision point at hand. The process of Kernel selection is hugely dependent on the distribution of the input vectors and association between input vector and output vector (predicted variables). But, there is no information of the feature space distribution in prior and it may vary while the solution process progresses, particularly in big

data cyber security. In order to resolve this problem, the novel work employs several kernel functions to boost the accuracy of the novel HMHIPSO-SVM algorithm and prevent the drawbacks involved in the usage of one individual kernel function.

SVM Configuration Formulation

The kernel function is utilized for computing the dot product of two sample points in the high-dimensional space. The kernel functions are defined by the input vectors and kernel parameters α, β and dthat the user specifies. A classical SVM configuration provides the suitable values forC,type of kernel and kernel parameters. SVM configurations from the space of all probable configurations, which reduce the anticipated error when tested on entirely new data [25]. This can be defined to be a black-box optimisation problem, which tries to find an optimal cross-validation error (I) and can be defined in the form of a tuple as$< SVM, \Theta, D, C, S >$given as follows.

1. SVM refers to the parametrised algorithm
2. Θindicates the search space of the probable SVM configurations (C, type of kernel and kernel parameters)
3. Dstands for the distribution of the set of instances
4. Crefers to the cost function, and
5. Sindicates the statistical information
6. The objective is the optimization the cost function $C: \Theta \times D \mapsto R$ of the SVM over a set of problem instances $\pi \in D$ to get

$$\theta^* \in \arg\min_{\theta \in \Theta} \frac{1}{|D|} \sum_{\pi \in D} C(\theta, \pi) \quad (1)$$

$\theta \in \Theta$indicates one probable configuration of SVM, cost function C indictes one single execution of the SVM utilizing $\theta$for resolving a problem instance $\pi \in D$. The statistical information S (e.g., a mean value) provides a summary of output of Creceived while testing the SVM across a group of instances. The primary purpose of the novel hyper meta-heuristic framework is to get a $\theta \in \Theta$so that $C(\theta)$gets optimal.

Multi-objective Formulation

A multi-objective optimisation problem consists of multiple one objective function, which all have to be optimized at the same time. The formulation of multi-objective optimization problem is done depending on three pre-defined objective parameters. SVM, the accuracy can be considered to be a compromise between the complexity Number of Support Vectors (*NSV*) and the margin (*C*) [26]. A huge number of support vectors may result in over-fitting, while a greater value of Cintended to improve the generalisation capability may lead to improper classification of some samples. This compromise can be suppressed by selecting SVM configuration (C, kernel type and kernel parameters). The training instances has two contrasting goals:

- **Accuracy**: The accuracy characterizes the classification performance on a certain problem instance.
- Complexity: The complexity indicates the number of support vectors (*NSV*) or the upper bound on the anticipated number of errors.

The false positive (fp), false negative (fn) and model complexity are chosen to be the constrasting goals. fp and fn refer to the expectancy rates associated with the accuracy, precision, recall and f-measure values. The

complexity is indicated by the Number of Support Vectors (NSV). The objectives that need optimization (m=3) can be defined as given

$$\min F(X) = |f_1(x), f_2(x), f_3(x)| \tag{2}$$

where $f_1(x) = fp; f_2(x) = fn; f_3(x) = NSV$.

B. Hyper Meta-Heuristic Improved Particle Swarm Optimization (HMHIPSO)

The novel framework of HMHIPSO-SVM is illustrated in Figure 1. The technique has two sections: the SVM and the hyper meta-heuristic framework having Improved Particle Swarm Optimization (IPSO) for multi-objective optimization. The primary purpose of the hyper meta-heuristic framework is to create a configuration (C, type of kernel and kernel parameters) and provide it to the SVM.

FIGURE 1: THE PROPOSED FRAMEWORK OF HMHIPSO-SVM

The novel framework integrates the merits of decomposition- and Pareto (dominance)- based techniques for efficiently approximating the Pareto set of SVM configurations. The novel framework carries out the procedures involving the selection of cost optimization based configuration for the SVM. SVM model consists of the SVM configuration and formulation of kernel function and kernel parameters in addition to the margin c and other parameters. The cost function that is modelled with the help of the SVM configuration parameters needs to get optimized with the multi-objective optimization function. The solution represents a single configuration ($\theta \in \Theta$) of SVM, which is optimal and is defined in the form of a one-dimensional array with the margin,type of kernel and kernel parameters to be chosen. The IPSO population is initialized in random through the assignment of any random value to all decision variables.

$$x_i^p = l_i^p + Rand_i^p(0,1) \times \left(u_i^p - l_i^p\right), p = 1,2,..,|PS|; I = 1,2,..,d \tag{3}$$

where $i$represents the index of the decision variable, $d$stands for the overall number of decision variables, $p$represents the index of the solution, $|PS|$stands for the population size, $Rand_i^p$represents the random value in the range [0,1] for the i^{th}decision variable. The fitness computation is carried out in the IPSO through the estimation of the rules generated for optimizing the cost function depending on the objective function given in equation 2. Training instances T decides the most useful configuration with reduced cost function. The fitness function is first divided into several single-objective sub-problems that are resolved collaboratively to create the objective values.

The fitness function is defined as

$$fitness(X) = \left|\frac{1}{T}\right| \sum_{i=1}^{|T|} z_i(X) \tag{4}$$

Received: 02 March 2020 | Revised: 21 April 2020 | Accepted:05 May 2020

$$subject\ to\ z_i(X) = \min F(X)$$

High-level strategy**:**It helps in automatically carrying out the heuristic selection by selecting the heuristics consecutively and using it for the solutions.During the selection stage, heuristics are selected from the available set of heuristics created by the low-level heuristics. In this selection phase, two variables are important. Those two variables include empirical reward and confidence level. The rewards acquired in the earlier performance are known as empirical reward whereas the frequency of the heuristic represents the confidence level. Depending on these two variables, heuristics is regarded to be either fit or unfit for the present state of operation. The first solution for which the heuristic has to be used in the particle mating procedure is chosen. The solution is chosen depending on the particle position and velocity of the IPSO algorithm. The particle that is submitted as g_{best}solution has the solution, which has to be used. The heuristic is used with the chosen solution for the formation of newer set of solutions. Then the new solutions are compared and analysed using their properties in terms of configuration to decide if they can be included in the available set of solutions or discard them to bring in newer solution from the next iterations.

Low-level heuristics: The low-level heuristics has the set of problem associated rules produced to yield solutions for every chosen problem instance. It takes one or multiple solutions into consideration and either merges or changes them to create new set of solutions. The solutions are generated with the help of several search based operations. IPSO based search procedure is one of the search operations used for the formation of new solutions from the available set of solutions. Once the new solutions are formed by low-level heuristics and the high-level strategy is selected, they are archived in the non-dominant set of solutions. The IPSO choose the solutions from this archive depending on the Pareto Front (PF) and gives back the best configuration to be the final solution. Pareto-optimal solutions is known as the Pareto-optimal Set (PS), and its image in the objective space is known as Pareto Front (PF). The primary objective of optimisation algorithms is to get the optimumPS.

Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is a computational technique, which helps in optimizing a problem by iteratively attempting to modify a candidate solution with respect to a certain quality metric. It resolves a problem using a population of candidate solutions, dubbed particles, and then transporting these particles around in the search-space based on simple mathematical formulae over the position and velocity particle. The movement of each particle depends on its local best known position, but is also encouraged to move toward the best known positions found in the search-space, which are updated to be the better positions that are found by the remaining particles. This is hoped to transport the swarm toward the best solutions. The selection of PSO parameters can strongly influence the optimization performance. Therefore, selection of the PSO parameters, which yields much better performance has remained the topic of research primarily. The PSO parameters can also be adjusted with the the help of another overlaying optimizer, a concept called as meta-optimization or even refined during the optimization process, e.g., using fuzzy logic. Parameters have also been adjusted for different optimization conditions[27].

Improved Particle Swarm Optimization (IPSO)

The novel improvement done to PSO algorithm is the local search capability improvement. The velocity and position update of PSO is expressed by

$$v_{id} = w * v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (5)$$

$$x_{id} = x_{id} + v_{id} \quad (6)$$

v_{id}represents velocity of particle i at D-dimensional vector, x_{id}stands for position of particle i at D-dimensional vector, $w$indicates the inertia weight, $c_1$and c_2 refers to positive acceleration factors, $r_1$and $r_2$refers to the random functions differing between [0,1]. $p_{id}$and p_{gd}stands for the local optimal extreme, global optimal extreme.

Conventional PSO algorithm's global search capability reduces when the number of iterations increase. To deal with this problem, local minimum value is reduced to improve the global search ability. IPSO targets atgetting over this drawback by improving the inertia weight at every iteration and exchange the high frequency non-linear function to be the inertia weight attenuation function to increase the speed of convergence.

The velocity is changed depending on the new inertia weight is $w(t)$defined as

$$v_{id} = w(t) * v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (8)$$

$$w(t) = w_{max} - \left(\frac{w_{max} - w_{min}}{2^{\frac{t}{T_{max}}}} \right) * \left(\frac{t}{T_{max}} \right)^2 \quad (9)$$

$w(t)$represents the inertia weight after t iteration, T_{max}indicates the maximum iteration, $w_{max}$and w_{min}stand for the maximum and minimum inertia weights. The t value initially is small, inertia weight is near to $w_{max}$and improves the global search capability. With the increase in the number of iterations, the inertia weight remains balanced and improves local search capability and averts the local minimum value.

IV. RESULTS AND DISCUSSION

The assessment of the novel HMHIPSO-SVM framework is carried out with the help of two standard instances of cyber security problems, which includemalware big data classification and anomaly intrusion detection. It is an approach used for comparing algorithms against standard ones to guarantee the accuracy and model complexity. The SherLock data [28-29] collection agent is dependent on the Google Funf framework. This Funf Open Sensing Framework is fundamentally a flexible sensing and data processing framework utilized for mobile devices, developed by the MIT Media Lab. Sensors can get the data from either physical or virtual sources (e.g., external temperature or memory usage). Usually, there are two types of sensors, which include PUSH and PULL. The data collected by SherLock is stored temporarily on the volunteer's device as a text file in JSON format. SherLock dataset for cyber security research involves Malware Detection & App Profiling. The dataset can be used for detecting malware and profile applications using inherent application activity (network traffic, CPU/memory usage, etc.). In addition, the dataset includes multiple contextual properties such as location of device, motion, and battery usage, which can be helpful in making the detection of intrusive threats better.

Theperformance analysis is done in terms of metrics such as precision, recall, f-measure and accuracy of the HMHIPSO-SVM proposed framework. The metric utilized for two class classification task is based on the confusion matrix given in table 1. A confusion matrix depicts a table, which is mostly utilized for describing the performance of a classification model (or "classifier") on a set of test data for which there are known true values. It permits the visualizing the performance that an algorithm achieves [30]. In predictive analytics, a confusion

matrix (Table 1), consists of a table with two rows and two columns with respect to the number of false positive (fp), false negative (fn), true positive (tp), and true negative (tn). The results of the novel HMHIPSO-SVM are compared with FKM, HH-SVM. The Proposed HMHIPSO-SVM classifier yields much better accuracy in comparison with other available classification algorithms such as FKM, HH-SVM.

TABLE 1: CONFUSION MATRIX

		Predicted	
		Positive	**Negative**
Actual	**Positive**	tp	fp
	Negative	fn	tn

- True Positive (tp) refers to correct prediction of a label (predicted “yes”, and it’s “yes”),
- True Negative (tn) refers to correct prediction of the other label (predicted “no”, and it’s “no”),
- False Positive (fp) indicates the false prediction of a label (predicted “yes”, but it's “no”),
- False Negative (fn) is called as missing and incoming label (predicted “no”, but it’s “yes”).

Precision (P) is known as the ratio of the expected positive cases, which were right, as defined by equation (10)

$$\text{Precision (P)} = tp/(tp+fp) \quad (10)$$

Recall or True Positive Rate (TPR) stands for the ratio of positive cases, which were rightly obtained, defined by equation (11)

$$\text{TPR} = tp/(tp+fn) \quad (11)$$

F-measure or balanced F-score (F1 score) is called as the harmonic mean of precision and recall. By multiplying it with the constant 2, it makes the score to 1 when recall and precision are 1 according to equation (12)

$$\text{F-score} = 2tp/(2tp+fp+fn) \quad (12)$$

Accuracy is defined to be the ratio of the total number of predictions, which were correct. It is expressed by equation (13)

$$\text{Accuracy} = (tp+tn)/(tp+tn+fp+fn) \quad (13)$$

Theoverall results of the techniques with the performance evaluation metrics against classification techniques are explained in table 2.

TABLE 2: PERFORMANCE COMPARISON METRICS VS. CLASSIFICATION METHODS

Methods	**Metrics**			
	Precision (%)	**Recall (%)**	**F-measure (%)**	**Accuracy (%)**
FKM	81	89	84	80
HH-SVM	85	91	88	84
HMHIPSO-SVM	87	94	91	87

Received: 02 March 2020 | Revised: 21 April 2020 | Accepted:05 May 2020

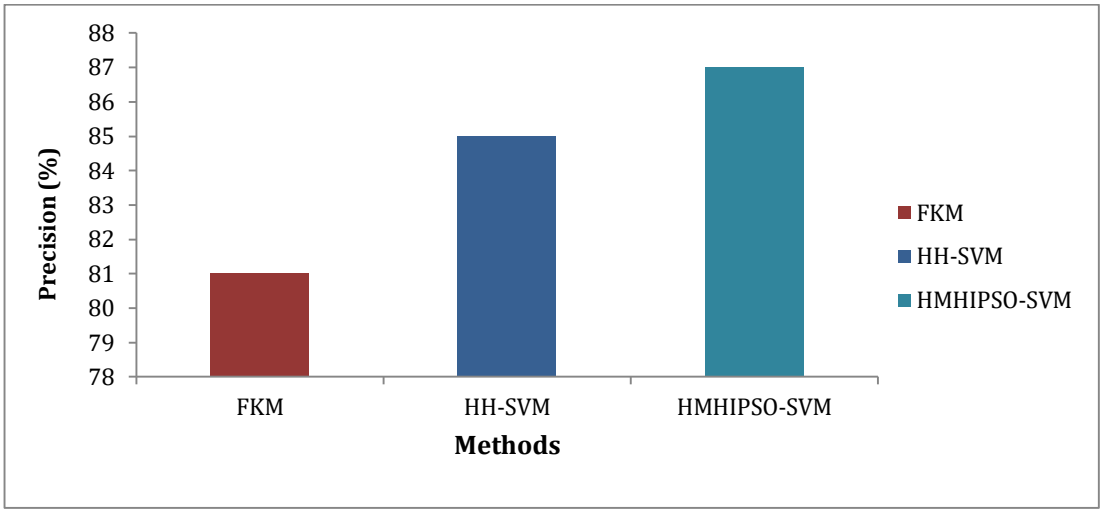

FIGURE 2: PRECISION RESULTS EVALUATION OF CLASSIFICATION METHODS

Figure 2 demonstrates the performance results of precision metrics with regard to three classifiers such as the proposed FKM, HH-SVM and HMHIPSO-SVM. The results prove that the novel HMHIPSO-SVM classifier yields much better precision value of 87%, while the other available techniques such as FKM, HH-SVMyields much lesser precision value of 81%, 85% correspondingly.

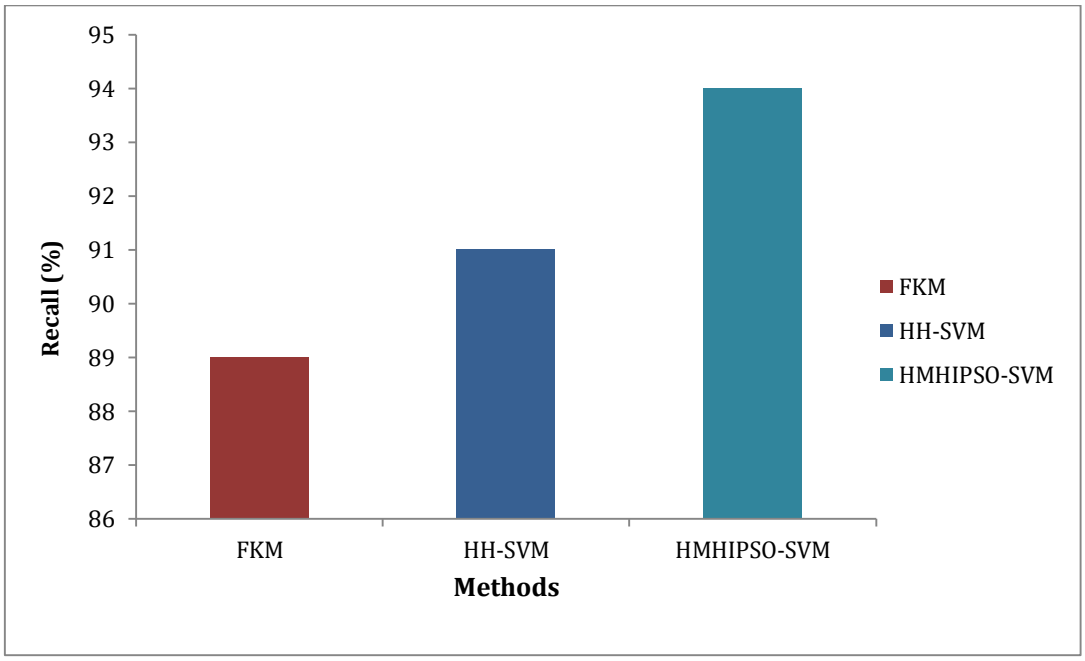

FIGURE 3: RECALL RESULTS EVALUATION OF CLASSIFICATION METHODS

Figure 3 shows the recall results metrics with regard to three classifiers such as the proposed FKM, HH-SVM and HMHIPSO-SVM. It is revealed from the figure 3 that the proposed HMHIPSO-SVM classifier yields much better recall value of 94%, while the other available techniques like FKM, HH-SVM yields lesser recall value of 89%, 91% correspondingly.

Received: 02 March 2020 | Revised: 21 April 2020 | Accepted:05 May 2020

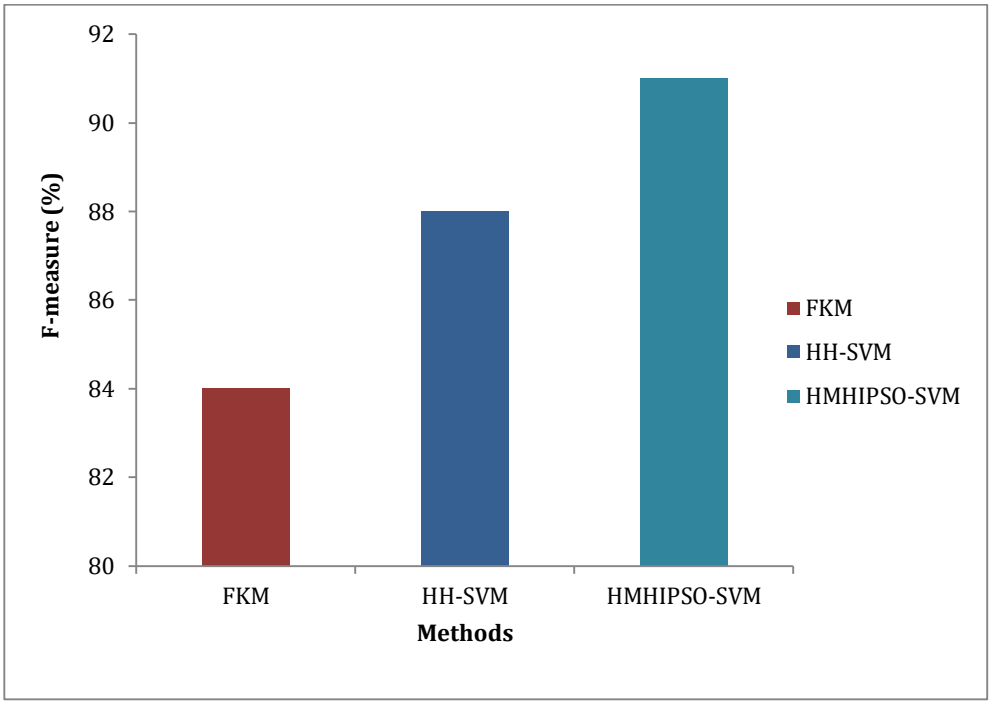

FIGURE 4: F-MEASURE RESULTS EVALUATION OF CLASSIFICATION METHODS

Figure 4 illustrates the results of F-measure comparison among the three classification techniques. Those techniques include FKM, HH-SVM and HMHIPSO-SVM. The results show that the novel HMHIPSO-SVM classifier yields greater f-measure results of 91%, while other available techniques like FKM, HH-SVM yields 84% and 88% correspondingly.

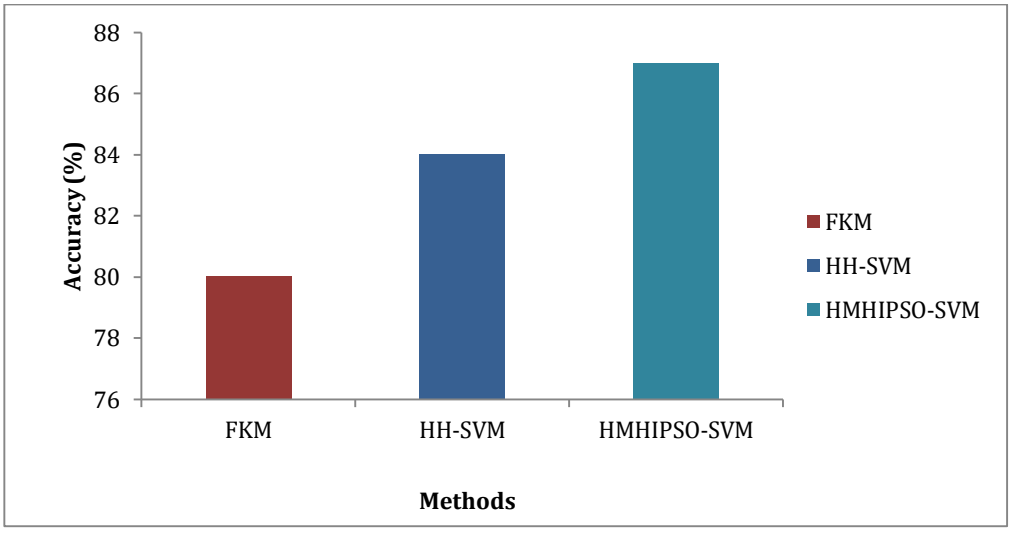

FIGURE 5: ACCURACY RESULTS EVALUATION OF CLASSIFICATION METHODS

Figure 5 shows the results of the accuracy comparison among the three classification techniques. The results reveals that the novel HMHIPSO-SVM classifier yields much better accuracy results of 87%, while other available techniques like FKM, HH-SVM yields 80% and 84% correspondingly.

V. CONCLUSION AND FUTURE WORK

Received: 02 March 2020 | Revised: 21 April 2020 | Accepted:05 May 2020

Hyper Meta-Heuristic Improved Particle Swarm Optimization based SVM (HMHIPSO-SVM) configuration framework is introduced to find a solution to the problems faced in big data cyber security. The high-level mechanism makes use of the search performance to manage the choice of the low-level heuristic that has to be utilized for the creation of a SVM configuration. This SVM configuration is brought into use and the data has been taken individually. SVM configuration is modelled in the form of a multi-objective optimization problem having false positive (fp), false negative (fn), true positive (tp), true negative (tn) and the model complexity taken to be the multiple objective parameters. Multi-objective optimization problem is resolved through the introduction of HMHIPSO framework which uses the high-level mechanism and low-level heuristics of hyper meta-heuristic approach and a modified PSO algorithm. It is designed with the intent of optimizing the SVM parameters of multi-objective optimization problem by including the hyper meta-heuristic and Improved Particle Swarm Optimization (IPSO). The framework combines the potential of decomposition- and Paretobased techniques for the approximation of the Pareto set of configurations. This novel HMHIPSO framework helps improving the choice of SVM margin, type of kernel and kernel parameters for getting a superior configuration of SVM forproblems involving cyber security big data. The results of the novel framework demonstrate in terms of metrics such as precision, recall, f-measure and accuracy is analyzed, and along with it the model complexity is considered to be the two contradicting objectives.The novel framework has been validated on two standard cyber security problem instances, such as malware big data classification and anomaly intrusion detection. The results show that the novel HMHIPSO-SVM classifier attained the highest value accuracy, which is 87%, in comparison with other benchmarkedtechniques, which attained much lesser classification accuracy. The futuristic approach can consider the same model of real life conditions and the various objectives of protection from thedanger of malicious intruders.

VI. REFERENCES

1. A. Ju, Y. Guo, Z. Ye, T. Li and J. Ma, HeteMSD: A Big Data Analytics Framework for Targeted Cyber-Attacks Detection Using Heterogeneous Multisource Data, Security and Communication Networks, 2019.
2. C.W. Tsai, C.F. Lai, H.C. Chao and A.V. Vasilakos, Big data analytics: a survey, Journal of Big data, Vol.2, No.1, Pp.21-50, 2015.
3. J. Hu and A.V. Vasilakos, Energy Big Data Analytics and Security: Challenges and Opportunities, IEEE Transactions on Smart Grid, Vol.7, No.5, Pp. 2423-2436, 2016.
4. N.R. Sabar, J. Abawajy and J. Yearwood, Heterogeneous cooperative co-evolution memetic differential evolution algorithm for big data optimization problems, IEEE Transactions on Evolutionary Computation, Vol.21, No.2, Pp.315-327, 2016.
5. M. Chen, S. Mao and Y. Liu, Big data: A survey, Mobile networks and applications, Vol.19, No.2, Pp.171-209, 2014.
6. Z. Cui, F. Xue, X. Cai, Y. Cao, G.G. Wang and J. Chen, Detection of malicious code variants based on deep learning, IEEE Transactions on Industrial Informatics, Vol.14, No.7, Pp.3187-3196, 2018.
7. Y. Ye, L. Chen, S. Hou, W. Hardy and X. Li, DeepAM: a heterogeneous deep learning framework for intelligent malware detection, Knowledge and Information Systems, Vol.54, No.2, Pp.265-285, 2018.

Received: 02 March 2020 | Revised: 21 April 2020 | Accepted:05 May 2020

8. Y. Ye, T. Li, D. Adjeroh and S.S. Iyengar, A survey on malware detection using data mining techniques, ACM Computing Surveys (CSUR), Vol.50, No.3, Pp.1-40, 2017.
9. S. Suthaharan, Big data classification: Problems and challenges in network intrusion prediction with machine learning, ACM SIGMETRICS Performance Evaluation Review, Vol.41, No.4, Pp.70-73, 2014.
10. J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-an and H. Ye, Significant permission identification for machine-learning-based android malware detection, IEEE Transactions on Industrial Informatics, Vol.14, No.7, Pp.3216-3225, 2018.
11. L. Haghnegahdar and Y. Wang, A whale optimization algorithm-trained artificial neural network for smart grid cyber intrusion detection, Neural Computing and Applications, Pp.1-15, 2019.
12. F. Matern, C. Riess and M. Stamminger, Gradient-Based Illumination Description for Image Forgery Detection, IEEE Transactions on Information Forensics and Security, Vol.15, Pp.1303-1317, 2019.
13. M.H. Etesami, D.M. Vilathgamuwa, N. Ghasemi and D.P. Jovanovic, Enhanced metaheuristic methods for selective harmonic elimination technique, IEEE Transactions on Industrial Informatics, Vol.14, No.12, Pp.5210-5220, 2018.
14. J. Wu, M. Dong, K. Ota, J. Li and Z. Guan, Big data analysis-based secure cluster management for optimized control plane in software-defined networks, IEEE Transactions on Network and Service Management, Vol.15, No.1, Pp.27-38, 2018.
15. D.S. Terzi, R. Terzi and S. Sagiroglu, Big data analytics for network anomaly detection from netflow data, International Conference on Computer Science and Engineering (UBMK), Pp.592-597, 2017.
16. I. Kiss, B. Genge, P. Haller and G. Sebestyén, Data clustering-based anomaly detection in industrial control systems, IEEE 10th International Conference on Intelligent Computer Communication and Processing (ICCP), Pp.275-281, 2014.
17. T.T. Teoh, Y.Y. Nguwi, Y. Elovici, N.M. Cheung and W.L. Ng, Analyst intuition based Hidden Markov Model on high speed, temporal cyber security big data, 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Pp.2080-2083, 2017.
18. T.T. Teoh, Y. Zhang, Y.Y. Nguwi, Y. Elovici and W.L. Ng, Analyst intuition inspired high velocity big data analysis using PCA ranked fuzzy k-means clustering with multi-layer perceptron (MLP) to obviate cyber security risk, 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Pp.1790-1793, 2017.
19. J. Liu, X.W. Feng, J. Li and D.X. Wang, Cyber Security Situation Awareness Based on Data Mining, Advanced Materials Research, Vol.756, Pp.4336-4342, 2013.
20. Z. Ruan, Y. Miao, L. Pan, N. Patterson and J. Zhang, Visualization of big data security: a case study on the KDD99 cup data set, Digital Communications and Networks, Vol.3, No.4, Pp.250-259, 2017.
21. N.R. Sabar, X. Yi and A. Song, A bi-objective hyper-heuristic support vector machines for big data cyber-security, IEEE Access, Vol.6, Pp.10421-10431, 2018.
22. Y. Ma, W. Chen, X. Ma, J. Xu, X. Huang, R. Maciejewski and A.K. Tung, EasySVM: A visual analysis approach for open-box support vector machines, Computational Visual Media, Vol.3, No.2, Pp.161-175, 2017.

Received: 02 March 2020 | Revised: 21 April 2020 | Accepted:05 May 2020

23. I. Ahmad, M. Basheri, M.J. Iqbal and A. Rahim, Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection, IEEE Access, Vol.6, Pp.33789-33795, 2018.
24. W. Zhao, T. Fan, Y. Nie, F. Wu and H. Wen, Research on attribute dimension partition based on SVM classifying and MapReduce, Wireless Personal Communications, Vol.102, No.4, Pp.2759-2774, 2018.
25. S.M. Othman, F.M. Ba-Alwi, N.T. Alsohybe and A.Y. Al-Hashida, Intrusion detection model using machine learning algorithm on Big Data environment, Journal of Big Data, Vol.5, No.1, Pp.34-45, 2018.
26. H. Wang, W. Wang, L. Cui, H. Sun, J. Zhao, Y. Wang and Y. Xue, A hybrid multi-objective firefly algorithm for big data optimization, Applied Soft Computing, Vol.69, Pp.806-815, 2018.
27. T. Su, H. Xu and X. Zhou, Particle Swarm Optimization-Based Association Rule Mining in Big Data Environment, IEEE Access, Vol.7, Pp.161008-161016, 2019.
28. http://bigdata.ise.bgu.ac.il/sherlock/#/
29. http://bigdata.ise.bgu.ac.il/sherlock/#/download
30. M. Ohsaki, P. Wang, K. Matsuda, S. Katagiri, H. Watanabe and A. Ralescu, Confusion-matrix-based kernel logistic regression for imbalanced data classification, IEEE Transactions on Knowledge and Data Engineering, Vol.29, No.9, Pp.1806-1819, 2017.

Received: 02 March 2020 | Revised: 21 April 2020 | Accepted:05 May 2020