Improving Electricity Theft Detection using Combination of Improved Crow Search Algorithm and Support Vector Machine

Hassan Ghaedi[1], Seyed Reza Kamel Tabbakh[2*], Reza Ghaemi[3]
1-Department of Computer, Neyshabur Branch, Islamic Azad University, Neyshabur, Iran.
Email: H.ghaedi@iau-neyshabur.ac.ir
2-Department of Computer, Mashhad Branch, Islamic Azad University, Mashhad, Iran.
Email: rezakamel@ieee.org(Corresponding author)
3-Department of Computer, Quchan Branch, Islamic Azad University, Quchan, Iran.
Email: rezaghaemi73@gmail.com

Received: March 2021 Revised: May 2021 Accepted: July 2021

ABSTRACT:
Advanced Metering Infrastructure (AMI) is an essential segment of the smart grids that is responsible for gathering, measuring and analyzing the electricity demand. Energy losses in the electricity distribution and transmission network and electricity theft detection are major challenges of electricity suppliers around the world. The analysis of consumption data related to the customers is one of the essential resources to identify electricity thieves. In this paper, the Crow Search Algorithm (CSA) is improved and the factors weight (w) and awareness probability (AP) are obtained dynamically and used to adjust the parameters C and γ related to the Support Vector Machine (SVM). The results illustrate that the ICSA-SVM framework has acceptable performance and detects fraudulent customers with high accuracy.

KEYWORDS: Data Mining, Theft Detection, Smart Grid, Classification, Crow Search Algorithm (CSA).

1. INTRODUCTION

A smart grid is constructed by communication and information technology in the electrical networks [1]. The smart grid is a large automated energy grid where the transmission of electrical power and the exchange of information take place bilaterally. It is capable of monitoring and responding to any network changes, from production sources to consumers and even individual equipment [2]. Energy losses in the electricity transmission and distribution of the power systems and predicting electricity theft are the major challenges that electricity suppliers are facing around the world. Energy losses are classified into technical and non-technical losses. Technical losses usually occur naturally in electricity transmission at transmission lines and transformers, and non-technical losses are mostly caused by electricity theft due to tampering the meters or faulty electric meter. The non-technical losses cause economic losses to electricity companies. For example, the annual loss of electricity in the United States is estimated to be about US$4.5 billion [3]. Furthermore, the financial losses of electrical theft may also be affected by the safety of electricity systems. Increasing the power load due to the electricity theft may cause system fire. Today, there are millions of smart meters in smart grids across Europe and United States that collect and report customer power consumption data over a period of time, and these readings help electricity companies to improve service quality [4]. AMI is a most vital part of the smart grids. The AMI system is not only a simple system to collect consumption information of the customers, but is also an integrated system comprised of hardware, software, network and telecommunication system. This integrated system receives information such as consumption, voltage, current and other information in real-time form consumer. This system, by creating a two-way telecommunication platform, has the capability to read, configure, monitor and remote control of meters, collect, manage, process and analyze the collected information [5]. The AMI system must be capable of managing over the network. Moreover, the system must be capable of controlling energy on the meters so that it can define the permissible amount of

Paper type: Research paper
DOI: https://doi.org/10.52547/mjee.15.4.63
How to cite this paper: H. Ghaedi, S. R. Kamel Tabbakh and R. Ghaemi, **"Improving Electricity Theft Detection using Combination of Improved Crow Search Algorithm and Support Vector Machine"**, *Majlesi Journal of Electrical Engineering*, Vol. 15, No. 4, pp. 63-75, 2021.

energy for each subscriber and alert to customer whenever exceeds the defined amount of energy. Despite the fact, the advantages of AMI in smart grids, the use of smart meters and the addition of a security layer to AMI have provided the condition for the electricity theft in new ways. Since traditional meters only provide non-physical casualties by physical manipulation, in AMI, data measurement can be manipulated both locally and remotely before sending data to smart meters [6]. There are various ways to detect electricity theft. Firstly, statistical methods were used and then techniques based on artificial intelligence such as neural networks, fuzzy and rough sets and today, data mining techniques have been widely adopted [7]. In the following, some studies are discussed in the field of electricity theft by the data mining approach. Angelos et.al. [7] presented a computational technique for classifying users' electricity consumption profiles, which consists of two steps. In the first step, a fuzzy clustering C-Means is presented to find customers with similar consumption profiles, and then a fuzzy classification has been done using fuzzy membership matrix and Euclidean distance for cluster centers. At the end, distance values are normalized and ordered and a unitary index score is created so that customers with the unconventional pattern of consumption have a highest number. The simulation results illustrate that the mentioned method has a highly efficiency. Huerta et.al. [8] have proposed a combined approach of genetic algorithm with Support Vector Machine (SVM) for classification of high amplitude data. This approach is a fuzzy logic based on the pre-filtering technique. In particular paper, a genetic algorithm has been used to extract a subset of data. By archiving information, a frequency-based technique has been presented for identifying data containing useful information. The results illustrate that the mentioned method recognizes better than other methods. One of the major challenges in the field of SVMs is the selection of optimal parameters that can affect SVM output performance. Zhang et.al. [9] have used the Ant Colony Optimization (ACO) algorithm to develop the ACO-SVM model in order to solve these challenges. in which that, the presented algorithm has been tested on several reference datasets, and the results indicate the high performance of the proposed algorithm. Given the importance of detecting electrical theft, Jindal et al. [10] have presented a top-down scheme based on the decision tree and SVM. The scheme is capable of detecting the theft with high performance and instantly at distribution and transmission levels and identifying its location. Indeed, the scheme is a two-level scheme that has used the processed data by decision tree as SVM input. The results indicate a false positive rate reduction of the proposed scheme. Due to the importance of detecting energy theft and the penetration issue to AMI, Yip et.al. [11] presented two algorithms based on the linear regression to study the customer energy consumption behavior and evaluate anomaly coefficients with the aim of preventing electricity theft caused by the manipulation of meters as well as meters that are physically damaged. In particular study, the main variables and diagnostic coefficients are responsible for introducing periods and fraudulent locations and meter failures. The simulation results describe that the presented method with high efficiency identifies fraudulent customers and meters that are damaged in an area. Given the importance of AMI in the smart grids and detecting unauthorized intruders, Jokar et.al. [12] presented an electricity theft detector based on consumption pattern of customer that uses customer's authorized and unauthorized consumption patterns to detect the electricity theft. Areas with high probability of electricity theft are quickly identified using distribution meters and fraudulent customers are identified by investigating abnormalities in consumption patterns. The detection rate of fraudulent customers is very high due to using classification and clustering techniques as well as the simultaneous use of distribution meters and anomaly detectors in the used method. One of the features of this algorithm is customer privacy. The results of the experiments illustrate that the presented algorithm is highly efficient for detection. In order to assist electricity generating companies, Li et.al. [13] proposed a random forest model based on the convolutional neural network for automatic detection of electricity theft. In the presented model, a convolutional neural network has been presented to learn features at different hours and days. In particular model a hybrid layer has been added for delaying the risk of over-fitting as well as the back-propagation algorithm to update the network parameters in the training phase. Random forest algorithm based on the obtained characteristics is trained to determine whether the consumer has committed theft or not. The results indicate that the presented model is more accurate and efficient than other algorithms. In order to detect electrical fraud and theft, Nagi et.al. [14] incorporated human knowledge into fraud detection model based on SVM by introducing a fuzzy inference system in the form of fuzzy IF-THEN rules. By implementing this improved system, the detection rate has increased by 12% compared to the previous work, which is economically cost effective. Lack of correct electricity consumption is a major challenge that energy producing companies are facing. Many researches have been done to detect unauthorized consumption of electricity. Nagi et.al. [15] have proposed a framework based on SVM for discovering non-technical losses using SVM. The presented framework examines customers whose consumption behavior is abnormal in a place where fraud has occurred. In fact, this framework is a data mining approach including the extraction of attributes from customer profiles over a period of time. The SVM uses this consumption profile to predict and detect fraud. The detection rate of fraud has increased to 60% in the presented method. Due to the importance of setting SVM parameters, many studies have been done in recent years. Pereira et.al. [16] proposed an extended meta-heuristic algorithm called Social Spider Optimization (SSO) for feature selection. In the presented study, model selection operations are performed by 3 scenarios of feature selection (TUNING), parameter setting and feature selection along with parameter setting. Experiments and their results describe that the best scenario is the combination of both feature selection and SVM parameter setting. Hassan et.al. [17] presented a combination of convolutional neural network and Long Short Term (LSM) memory architecture, an electricity theft detection system that classifies smart grid data. In particular paper, a preprocessing algorithm has been used to compute missing data and in order to enhance the performance of classifier, the dataset has been expanded with the manipulated data by using Synthetic Minority Over-Sampling Technique (SMOT) technique. The simulation results illustrate improvement in the values of recall, precision and F-score. Ahmad and Ul Hasan [18] conducted a comprehensive study on non-technical losses in electricity distribution systems. They have used SVM, central intelligent meter, electricity line connections, linear regression and Pearson correlation coefficient to detect electricity theft in a particular area. SVM has been used to classify unauthorized customers, the central meter for analyzing consumer data and linear regression to identify technical losses. The simulation results illustrate that the presented method is more efficient than other methods.

As mentioned above, the SVM classifier plays an important role in identifying fraudulent customers and detecting electrical theft. This paper also uses the SVM classifier for electricity theft detection. The method is to build a model using the electricity consumption pattern of 5,000 Irish customers stored in the ISSDA dataset. Proper selection of the kernel function and proper adjustment of its parameters play crucial roles in performance of the final prediction. Many research works [8-10,19-25] have used Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Decision Tree (DT), Ant Colony (ACO), Artificial Bee Colony(ABC), etc. algorithms to adjust SVM parameters. Due to the crow's superb intelligence over other animals and it's high performance than other meta-heuristic algorithms such as GA, PSO and harmony search (HS), its combination with SVM improves detection performance. In this paper, the Crow Search Algorithm (CSA) is improved and used to adjust the parameters C and γ due to the importance of setting SVM parameters. This paper is organized as follows: Section 2 reviews the SVM. The CSA algorithm is reviewed in Section 3. Section 4 describes the improved CSA algorithm. The proposed framework for theft detection is explained in Section 5. Experiments and evaluations are discussed in Section 6 and Section 7 is dedicated to the paper conclusions.

2. SUPPORT VECTOR MACHINE

The support vector machine is actually a binary classifier that separates two classes using a linear boundary. In data linear segmentation, the goal is to achieve the function that determines the most marginal hyper-plane. By maximizing the margin of this hyper-plane, the separation between classes is maximized. Suppose that $\mathrm{S} = \{\mathrm{x_i}, \mathrm{y_i}\}$ is a training example that consists of two classes $\mathrm{y_i} = \mp 1$ and both classes has m attributes. As shown in Fig 1, the line $\mathrm{w^T x + b = 0}$ classifies the available data into two classes ∓ 1. This line is called a separator hyper-plane. The two lines $\mathrm{w^T x + b = +1}$ and $\mathrm{w^T x + b = -1}$ represent the region boundary of the classes y = + 1 and y = -1, respectively. The training data closest to the separating hyper-plane are called support vectors.

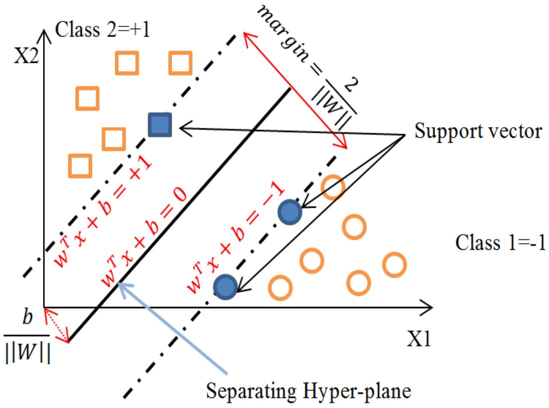


Fig. 1. Classification of data by SVM.

In the support vector machine, the set of points can be separated from each other in both linear and non-linear ways [26]. In the case where the data can be separated linearly, the support vector machine finds the optimal hyper-plane with maximum margin by considering the training data set, using the solution of the following optimization problem:

$$\begin{aligned} &Minimize \quad \tfrac{1}{2}||w||^2 \\ &S.t \qquad y_i(w^T x + b) \geq 1 \qquad i = 1,2,\dots,m \end{aligned} \tag{1}$$

Where, $||w||$ is Euclidean norm. Based on the above model, the minimum value of $||w||$ leads to maximizing the width of the plane margin.

It is difficult to solve this optimization problem. Therefore, in order to simplify its solution, Lagrangian coefficients are used. Then it's dual will be as follows according to the Karush[27]–Kuhn–Tucker[28] (KKT) condition:

$$L\left(\frac{w,b}{min},\frac{\alpha}{max}\right) =$$
$$\frac{1}{2}||w||^2 - \sum_{i=1}^{m} \alpha_i[y_i(wx_i + b) - 1] \qquad \alpha_i \geq 0 \qquad (2)$$

Now the above expression should be minimum for w and b and maximum for α. In this way, by obtaining the saddle point of the problem, the variables w, b and α are determined. Therefore, the decision function will be as follows:

$$y = sign(wx_i + b) = sign\left(\sum_{i,j=1}^{m} \alpha_i y_i \left(x_i . x_j\right) + b\right) \qquad (3)$$

Where, y is the output of the equation, y_i is the class value of the training sample x_i and represents the internal multiplication. x_j is an input data and the vectors x_i (i = 1,…, m) are support vectors. The support vector machine also has the ability to solve nonlinear classification problems. This is achieved by using the desired kernel function. To do this, the original data is mapped to a feature space with larger dimensions, where it is categorized linearly. Using the kernel function $K\left(x_i, x_j\right)$ the decision function is shown as follows:

$$y = sign\left(\sum_{i,j=1}^{m} \alpha_i y_i \, K\left(x_i, x_j\right) + b\right) \qquad (4)$$

Some of the kernel functions used in SVM were shown in Table 1.

Table 1. Common kernel function.

Kernel	$K\left(x_i, x_j\right)$
linear	${x_i}^T . x_j$
Polynomial	$({x_i}^T . x_j + 1)^d$
RBF	$exp\left(-\gamma \left\Vert x_i - x_j \right\Vert^2\right) \quad \gamma > 0$

As mentioned, SVM takes the data to a new space according to their predefined classes so that the data can be categorized linearly (or hyper-plane) and then by finding the support lines (support planes in multidimensional space), it tries to find the line that creates the greatest distance between the two categories.

Fig 1 shows the data in two groups of circles and squares, and the dotted lines show the support vectors corresponding to each category, denoted by filled circles and squares. The continuous black line is the same as the SVM. The support vectors (dotted lines) describe the boundary line of each category.

3. CROW SEARCH ALGORITHM

CSA is a meta-heuristic optimization algorithm based on the crow's artificial behavior that was developed by Askarzadeh [29]. The algorithm mimics the flock crows' collective intelligence in food gathering processes. Due to the improved performance of this algorithm compared to GA, PSO and HS algorithms, this algorithm is used to solve many complex engineering problems. It is considered that there is a d-dimensional space containing some crows. N is the number of crows and the location (X) of the crow i in iteration $iter$ is shown as the following vector:

$$X^{i,iter} \; i = 1,2, \dots, N \qquad iter = 1,2, \dots, Max_{iter}$$
$$X^{i,iter} = \left[X_1^{i,iter}, X_2^{i,iter}, \dots, X_d^{i,iter}\right]$$

Where, Max_{iter} represents maximum number of the iterations.

Fig.2 shows the steps of the CSA algorithm.

Each crow has a memory (Mem) containing a location to hide its food. The location of the crow i in iteration $iter$ is indicated by $Mem^{i,iter}$, which is the best location that the crow i has ever recorded. Crows move around the search space and looking for the best hidden locations of food. Suppose in iteration $iter$, the crow j decides to go to its hidden location where it has hidden food, in this iteration, the crow i decides to follow the crow j to discover its hidden food. In this situation, two decisions may be made:

Situation (1): crow j does not realize that the crow i follows it, consequently the crow i discovers the hidden location of the crow j and the new location of the crow i, $X^{i.iter+1}$ is updated as follows:

$$X^{i.iter} + r_i \times fl^{i,iter} \times (Mem^{j,iter} - X^{i,iter}) \qquad (5)$$

Where, r_i is a random number with a uniform distribution in the range (0,1), and also, fl represents the flight length of crow i in iteration $iter$.

Situation (2): The crow j realizes that the crow i is looking for it. As a result, the crow j tricks crow i to protect its food and then move to a new location. In general, the location of $X^{i,iter+1}$ is updated as follows:

$$\begin{cases} X^{i.iter} + r_i \times fl^{i,iter} \times \left(Mem^{j,iter} - X^{i,iter}\right) & r_j \geq AP^{j,iter} \\ a\ random\ location & otherwise \end{cases} \qquad (6)$$

Where, AP is the awareness probability of the crow j at iteration $iter$ and, r_j is a random number with a uniform distribution in the range (0,1).

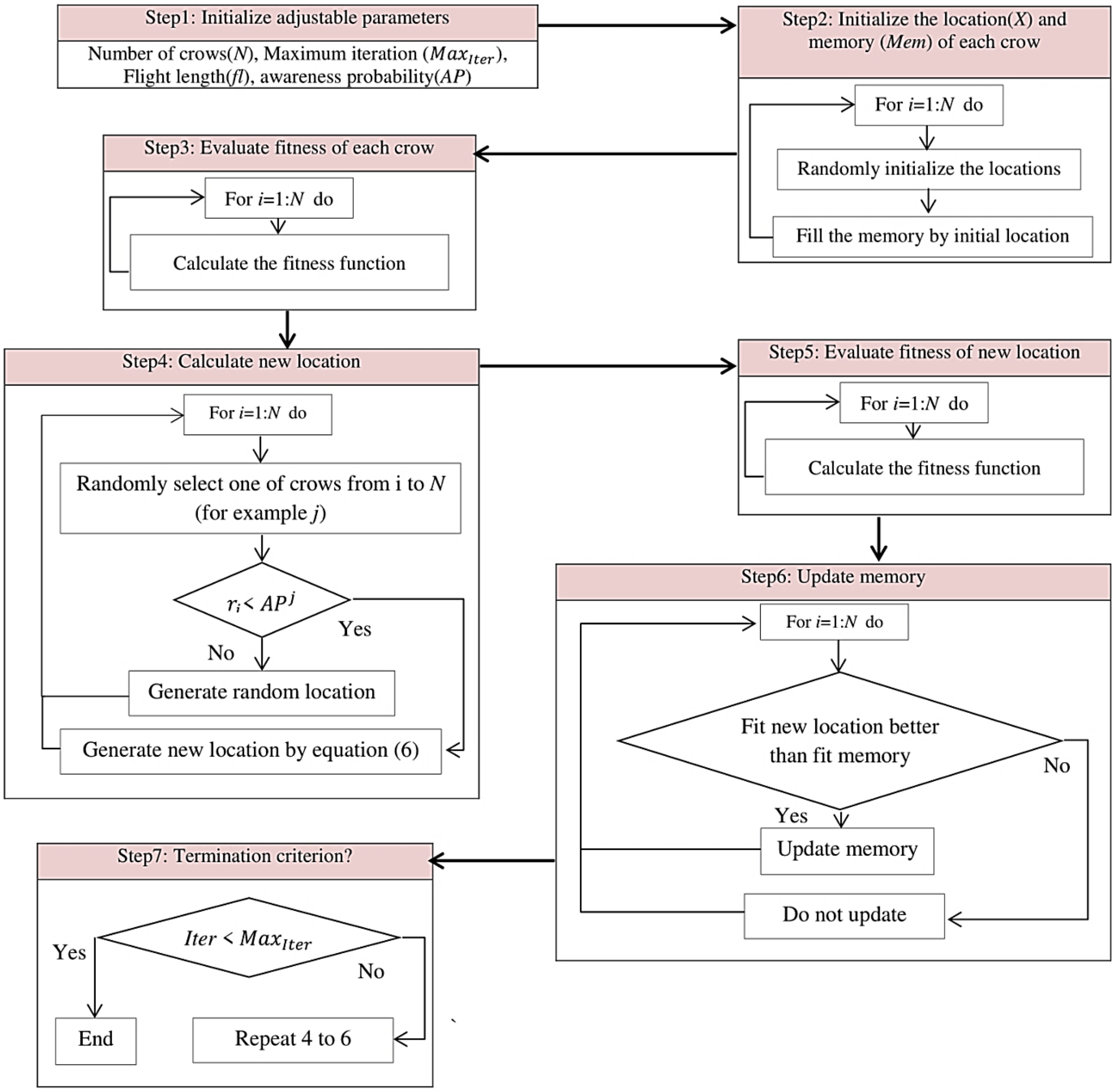

Step1: Initialize adjustable parameters
Number of crows(N), Maximum iteration (Max_{Iter}), Flight length(fl), awareness probability(AP)
Step2: Initialize the location(X) and memory (Mem) of each crow
For i=1:N do
Randomly initialize the locations
Fill the memory by initial location
Step3: Evaluate fitness of each crow
For i=1:N do
Calculate the fitness function
Step4: Calculate new location
For i=1:N do
Randomly select one of crows from i to N (for example j)
$r_i < AP^j$
Yes
No
Generate random location
Generate new location by equation (6)
Step5: Evaluate fitness of new location
For i=1:N do
Calculate the fitness function
Step6: Update memory
For i=1:N do
Fit new location better than fit memory
No
Yes
Update memory
Do not update
Step7: Termination criterion?
$Iter < Max_{Iter}$
Yes
No
End
Repeat 4 to 6

Fig. 2. The flowchart of the CSA algorithm.

4. IMPROVED CROW SEARCH ALGORITHM

In this section, the CSA algorithm is improved. One of the superior features of improved CSA (ICSA) algorithm compared to other meta-heuristic algorithms such as PSO, ACO, HS, etc. is its low number of adjustment parameters. This algorithm requires setting only the flight length (fl) parameter. In the ICSA algorithm, the crows' ability to find suitable food locations is determined by their weight characteristics in the search space. The weight of each crow is obtained according to the fitness function of each crow and the best and worst crows at each iteration. Indeed, each crow is affected by the weight of the other crows, and a crow with a good weight should have one more weight than a crow with a worse weight. The best and worst fitness of the crows' population in iteration $iter$ are calculated as follows:

$$Best\_F(iter) = \underset{i\epsilon\{1,2,\dots,N\}}{\boldsymbol{MIN}} F\left(Crow_{i,iter}\right) \quad (7)$$

$$Worst\_F(iter) = \underset{i\epsilon\{1,2,\dots,N\}}{\boldsymbol{MAX}} F\left(Crow_{i,iter}\right) \quad (8)$$

Where, $F\left(Crow_{i,iter}\right)$ is the fitness for crow i in iteration $iter$ that is obtained based on equation (14). N is number of the crows. The weight of crow i in iteration $iter$ is also obtained as follow:

$$w_{i,iter} = \frac{F(Crow_{i,iter}) - Worst\_F(iter)}{Best\_F(iter) - Worst\_F(iter)} \tag{9}$$

Due to the importance of AP to ensure diversification and intensification in the ICSA algorithm, the value of AP crow j in iteration $iter$ is adjusted as follows:

$$AP^{j,iter} = \frac{F(Crow_{i,iter})}{Worst\_F(iter)} \tag{10}$$

In various experiments, adjusting the AP values in the range (0,0.3] has led to the best results. After calculating the weight of each crow at the iteration $iter$, crow j is randomly selected and the movement probability(MP) of crow i towards crow j in iteration $iter$ is obtained as follows:

$$MP_{ij,iter} = \begin{cases} 1 & \dfrac{w_{i,iter} - Best\_w(iter)}{w_{j,iter} - w_{i,iter}} > rand() \\ & and \ w_{j,iter} > w_{i,iter} \\ 0 & otherwise \end{cases} \tag{11}$$

Where, *rand()* is a random number with a normal distribution between 0 and 1. If the conditions $r_j > AP^{j,iter}$ and $MP_{ij,iter} = 1$ are satisfied, crow i will move to crow$j$. Otherwise, it randomly searches for food in the search space. As a matter of fact, attraction and movement of a bad crow towards a good crow result in acceleration of convergence, and attracting a good crow to a bad crow causes a pervasive search in the search space. According to the description given, the parameters AP and w balance the diversification and intensification criteria and give the best solution to the output. At the end, the number of crows that follow crow j in iteration $iter$ based on its weight is obtained as follows:

$$C_{f\,j,iter} = C_{max} * w_{j,iter} \tag{12}$$

Where, C_{max} represents the maximum number of crows that can search around the crow j.

5. ICSA-SVM PROPOSED FRAMEWORK

One of the effective factors in improving the performance of SVM classifier is the correct adjustment of parameters C and γ. Parameter C represents the penalty cost that affects the output of the SVM. If parameter C is set too large, the classification accuracy rate in the training and testing phases will be very high and low, respectively. Also, if this parameter is set too small, the classification accuracy rate will be inappropriate and will cause SVM inefficiency. The effect of the parameter γ on the SVM results is greater than the parameter C because its value affects the output segmentation in the feature space. Very high and low values of the parameter γ cause over-fitting and under-fitting, respectively [30]. To adjust mentioned parameters, meta-heuristic algorithms such as PSO, ACO, BA, etc. are used. In order to increase the performance of prediction, the base CSA algorithm has been improved and the factors weight (w) and AP of crows have been used dynamically to improve SVM performance. In fact, the weight of each crow indicates its ability to attract other crows. The heavier crow has a greater ability to attract other crows and the search around it is more. The AP factor is also set in the range (0,0.3], which guarantees the criteria of diversification and intensification and causes the best result to be output. Algorithm 1 shows the steps of the ICSA- SVM framework. Also, Fig 3 represents the flowchart of proposed framework. The proposed framework consists of 11 steps.

(1) Read data from dataset and data preprocessing:

Some data may not be within the normal range of customers' consumption pattern and may be too small or too large. For this purpose, there is a need to normalize the data. In this study, the MIN-MAX normalization method falls all data within the [0,1] range.

MIN-MAX normalization method is done as follows:

$$X' = \frac{x_i - x_{min}}{x_{max} - x_{min}} \qquad \mathrm{i} = 1,2,\dots,\mathrm{N} \tag{13}$$

Where, N is the total number of data, X' is the normalized value, $\mathrm{x_i}$ is the $\mathrm{i^{th}}$ input data, $\mathrm{x_{min}}$ and $\mathrm{x_{max}}$ are the minimum and maximum values of the data, respectively.

(2) Production of abnormal samples

Given that the Irish social science data archive (ISSDA) dataset contains only normal and real electricity consumption data for Ireland's 5,000 domestic and commercial customers, there are no abnormal or tampered samples. In order to balance the normal and abnormal samples, the abnormal samples are added to the ISSDA dataset by using the artificial attacks [12], as presented in Table 2. The sampling rate is reduced to one per hour and the file is divided into a dataset of 535 vectors for each customer, each with 24 features. Then, based on the normal samples, for each sample ($y = y_1, \dots, y_{24}$), 6 types of abnormal samples is produced according to Table 2. Given the Table 2, the function $\mathrm{f_1(y_t)}$ multiplies all the samples in a random value and the function $\mathrm{f_2(y_t)}$ multiplies each value read in a different random value. The function $\mathrm{f_3(y_t)}$ is a by-pass attack that sends a value of zero at a specified time interval [ts, tf], otherwise it sends the actual amount of consumption. ts and tf are the beginning and end of the theft period, respectively. The functions $f_4(y_t)$ and $f_5(y_t)$ focus on sending the average daily consumption. The function $f_6(y_t)$ reverses the order of the readings in a day. By addition of abnormal samples, the performance of the proposed framework ICSA-SVM is increased greatly.

Table 2. Artificial attacks to generate abnormal samples in ISSDA dataset.

Attack	Formula
$f_1(y_t)$	$\alpha \times y_t$ α=random(0.1,0.8)
$f_2(y_t)$	$\alpha_t \times y_t$ α_t=random(0.1,0.8)
$f_3(y_t)$	$\begin{cases} 0 & \forall t \in [ts, tf] \\ y_t & \text{otherwise} \end{cases}$
$f_4(y_t)$	$a_t \times \text{mean}(y)$ a_t=random(0.1,0.8)
$f_5(y_t)$	$\text{mean}(y)$
$f_6(y_t)$	y_{24-t}

(3) At this step, the initial adjustments of the parameters of the ICSA algorithm and the SVM classifier are performed.

The parameters C and γ, along with the other parameters, are initialized.

(4) At this step, the location of each crow and it's memory are initialized.

(5) Using the SVM classifier, the fitness of each crow is calculated and the best local solution (Mem_{sol}) and the best global solution ($Best_{sol}$) are obtained. In this paper, the fitness of each crow is calculated based on $Accuracy$ of the classifier.

$$Fitness = Min \frac{1}{Accuracy_{ave}} \qquad (14)$$

To obtain an optimal combination of C and γ, 10-fold cross-validation is used. Cross-validation is used to test the performance of classifier in terms of different combinations of C and γ. If the training data set is randomly divided into k subsets (Fold) with the same size, in each step of the CV process, $k-1$ of these subsets can be used as the training data set and one as the testing data. By selecting $k = 10$, the number of repetitions of the CV process will be equal to 10 and the final result will be obtained from the average of the results of 10 steps.

(6) At this step, each crow searches a range with radius R_{sense} around the location previously initialized by itself. This search makes the initial location of the crows more realistic and the fitness and weight of the crows more acceptable. R_{sense} is calculated based on the number of the crows, the dimension of the problem and the range of the search space. In this paper, R_{sense} is set to 12.

At each iteration, steps 7 to 9 are repeated:

(7) The best and worst fitness of all crows are obtained.

(8) The weight and the number of follower crows of each crow are obtained.

(9) In this step, the location of each crow is updated according to the parameters AP and MP. Also, the fitness of the new locations are re-evaluated and the Mem_{sol} and Best_{sol} updated as needed.

(10) The best solution and the most optimal parameters C and γ are extracted.

(11) The SVM classifier is retrained with optimized parameters and labels of the new samples are predicted.

6. EXPERIMENTS AND ANALYSIS

This section examines the proposed framework and compares the results with other frameworks. The results of experiments demonstrate that normalization the features increases the classification performance of the SVM. In general, the range of each feature scaled to the range [0,1]. In this paper, parameters C and γ are set in ranges $0.01 \leq C \leq 100$ and $0.001 \leq \gamma \leq 20$, respectively.

Increasing the ranges of these parameters extends the search space and increases the computational time and slows down the convergence speed.

6.1. Dataset

This paper uses the ISSDA [31], Iris, Sonar, Vehicle and Wine [32] datasets to test and analyze the results of the proposed framework. The ISSDA dataset stores information on the electricity consumption of Ireland's 5,000 household and commercial customers, which are publicly available to researchers. Indeed, this dataset contains readings of every half hour of customer's meters in one day that are sent to the data center.

Table 3 shows the information of a meter, including the meter ID, consumption value in each hour (kWh), and consumption date that are sent to the center.

Table 3. Consumption information of a meter.

Date	Hour1(kWh)	…	Hour24(kWh)	Meter ID
29/8/2009	0.094	…	0.149	1184

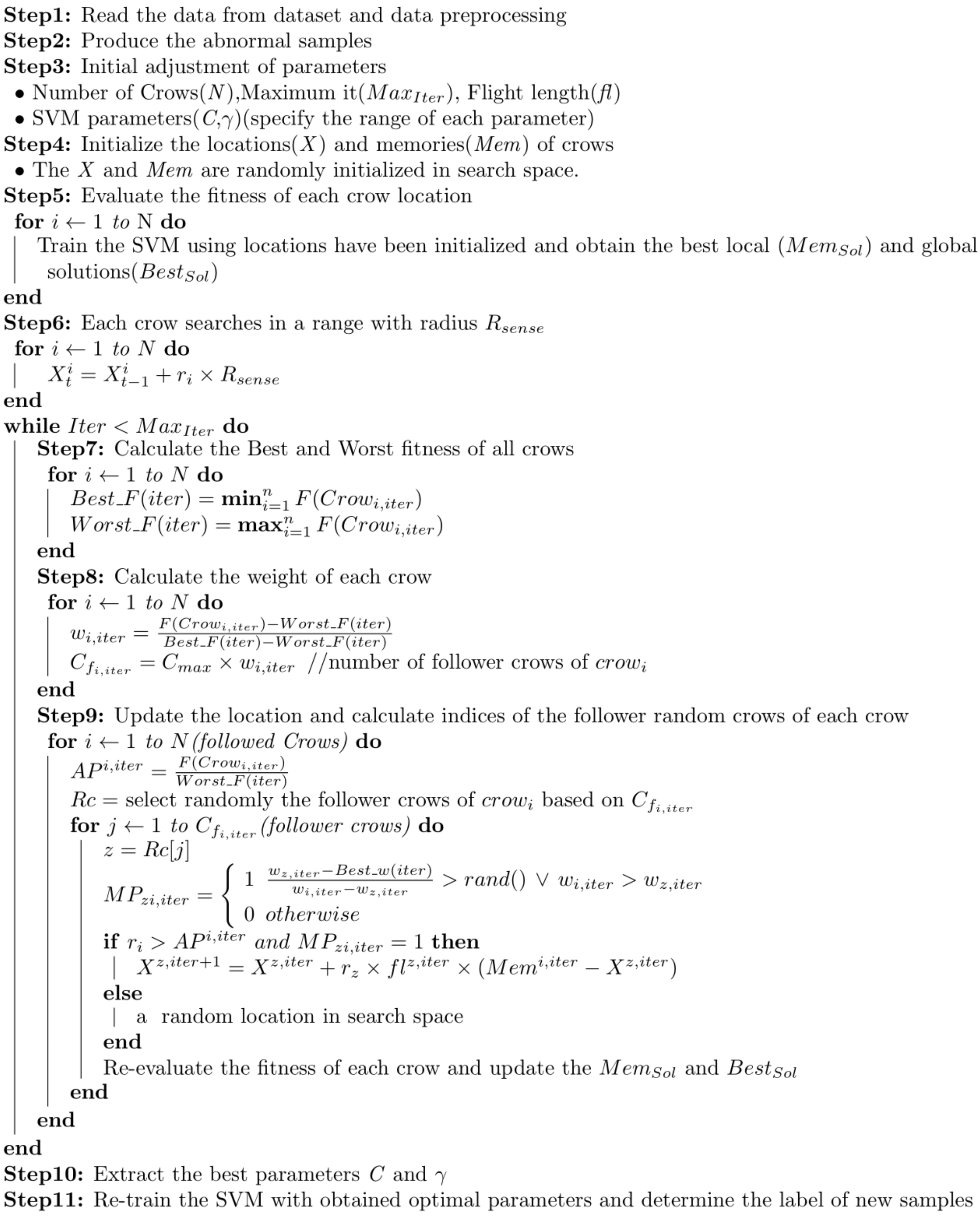
Hour1 (kWh) represents the amount of consumption electricity at 1 o'clock in the morning.

The Iris, Sonar, Vehicle and Wine datasets are reference datasets which are located in UCI repository that also used in this paper and more researches. Table 4 shows the details of these datasets.

Table 4. The reference datasets from UCI repository.

Dataset	No. of classes	No. of instances	No. of features
Iris	3	150	4
Sonar	2	208	60
Vehicle	4	846	8
Wine	3	175	13

For example, in Iris dataset, which contains 150 instances, each data falls into one of three classes: Iris Setosa, Iris Versicolour, and Iris Virginica. Each data has 4 features including sepal length in cm, sepal width in cm, petal length in cm and petal width in cm. Using classification, category of the new data is identified.



```
Step1: Read the data from dataset and data preprocessing
Step2: Produce the abnormal samples
Step3: Initial adjustment of parameters
 • Number of Crows($N$),Maximum it($Max_{Iter}$), Flight length($fl$)
 • SVM parameters($C$,$\gamma$)(specify the range of each parameter)
Step4: Initialize the locations($X$) and memories($Mem$) of crows
 • The $X$ and $Mem$ are randomly initialized in search space.
Step5: Evaluate the fitness of each crow location
 for $i \leftarrow 1$ to N do
     Train the SVM using locations have been initialized and obtain the best local ($Mem_{Sol}$) and global
      solutions($Best_{Sol}$)
 end
Step6: Each crow searches in a range with radius $R_{sense}$
 for $i \leftarrow 1$ to $N$ do
     $X_t^i = X_{t-1}^i + r_i \times R_{sense}$
 end
while $Iter < Max_{Iter}$ do
    Step7: Calculate the Best and Worst fitness of all crows
     for $i \leftarrow 1$ to $N$ do
         $Best\_F(iter) = \mathbf{min}_{i=1}^{n} F(Crow_{i,iter})$
         $Worst\_F(iter) = \mathbf{max}_{i=1}^{n} F(Crow_{i,iter})$
    end
    Step8: Calculate the weight of each crow
     for $i \leftarrow 1$ to $N$ do
         $w_{i,iter} = \frac{F(Crow_{i,iter})-Worst\_F(iter)}{Best\_F(iter)-Worst\_F(iter)}$
         $C_{f_{i,iter}} = C_{max} \times w_{i,iter}$ //number of follower crows of $crow_i$
    end
    Step9: Update the location and calculate indices of the follower random crows of each crow
     for $i \leftarrow 1$ to $N$ (followed Crows) do
         $AP^{i,iter} = \frac{F(Crow_{i,iter})}{Worst\_F(iter)}$
         $Rc$ = select randomly the follower crows of $crow_i$ based on $C_{f_{i,iter}}$
         for $j \leftarrow 1$ to $C_{f_{i,iter}}$ (follower crows) do
             $z = Rc[j]$
             $MP_{zi,iter} = \begin{cases} 1 & \frac{w_{z,iter}-Best\_w(iter)}{w_{i,iter}-w_{z,iter}} > rand() \vee w_{i,iter} > w_{z,iter} \\ 0 & otherwise \end{cases}$
             if $r_i > AP^{i,iter}$ and $MP_{zi,iter} = 1$ then
                 $X^{z,iter+1} = X^{z,iter} + r_z \times fl^{z,iter} \times (Mem^{i,iter} - X^{z,iter})$
             else
                 a  random location in search space
             end
             Re-evaluate the fitness of each crow and update the $Mem_{Sol}$ and $Best_{Sol}$
         end
    end
end
Step10: Extract the best parameters $C$ and $\gamma$
Step11: Re-train the SVM with obtained optimal parameters and determine the label of new samples
```

Algorithm 1. The steps of ICSA-SVM framework.

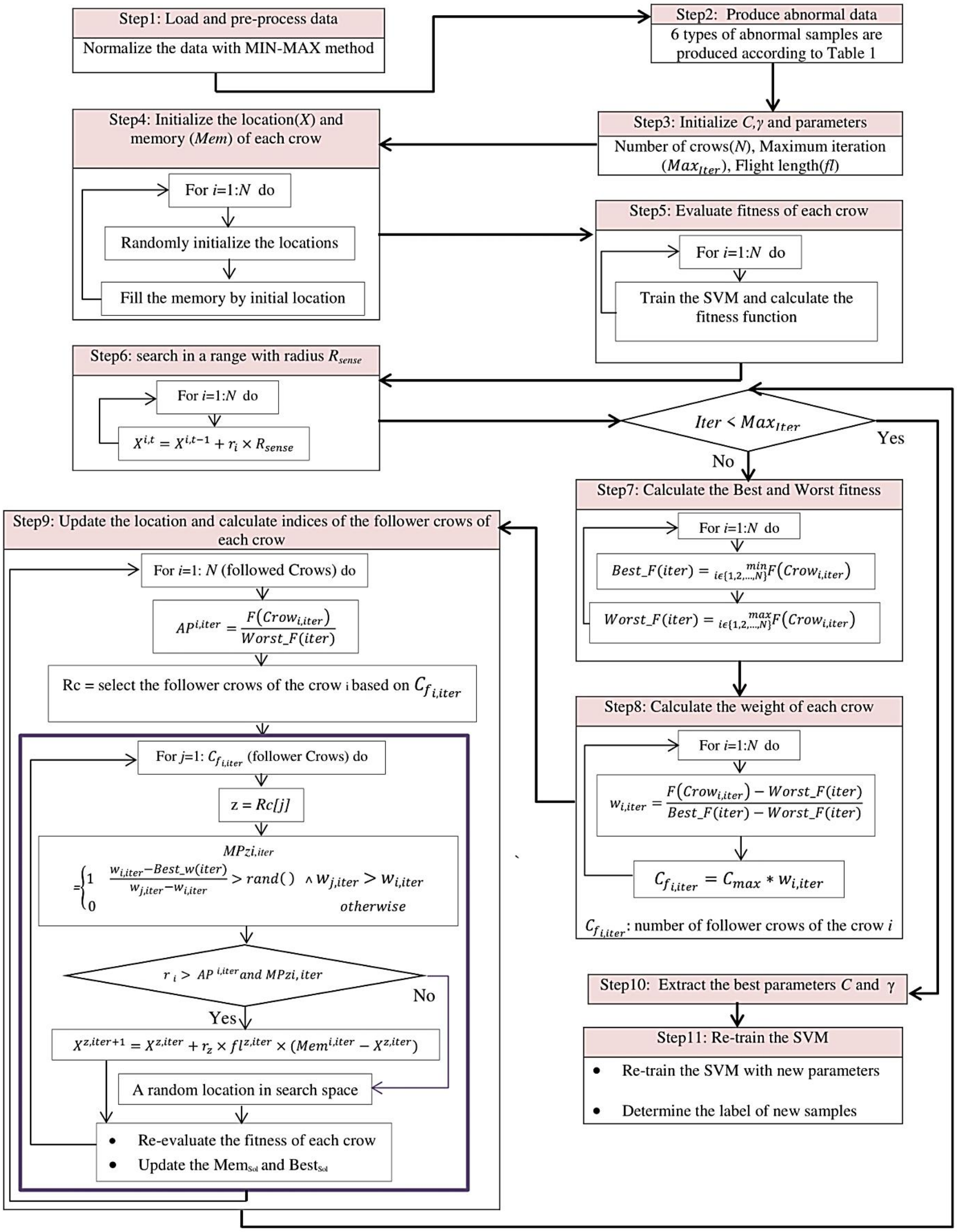

Step1: Load and pre-process data
Normalize the data with MIN-MAX method
Step2: Produce abnormal data
6 types of abnormal samples are produced according to Table 1
Step3: Initialize C,γ and parameters
Number of crows(N), Maximum iteration (Max_{Iter}), Flight length(fl)
Step4: Initialize the location(X) and memory (Mem) of each crow
For i=1:N do
Randomly initialize the locations
Fill the memory by initial location
Step5: Evaluate fitness of each crow
For i=1:N do
Train the SVM and calculate the fitness function
Step6: search in a range with radius R_{sense}
For i=1:N do
$X^{i,t} = X^{i,t-1} + r_i \times R_{sense}$
$Iter < Max_{Iter}$
Yes
No
Step7: Calculate the Best and Worst fitness
For i=1:N do
$Best\_F(iter) = \min_{i\epsilon\{1,2,\dots,N\}} F(Crow_{i,iter})$
$Worst\_F(iter) = \max_{i\epsilon\{1,2,\dots,N\}} F(Crow_{i,iter})$
Step8: Calculate the weight of each crow
For i=1:N do
$w_{i,iter} = \frac{F(Crow_{i,iter}) - Worst\_F(iter)}{Best\_F(iter) - Worst\_F(iter)}$
$C_{f_{i,iter}} = C_{max} * w_{i,iter}$
$C_{f_{i,iter}}$: number of follower crows of the crow i
Step9: Update the location and calculate indices of the follower crows of each crow
For i=1: N (followed Crows) do
$AP^{i,iter} = \frac{F(Crow_{i,iter})}{Worst\_F(iter)}$
Rc = select the follower crows of the crow i based on $C_{f_{i,iter}}$
For j=1: $C_{f_{i,iter}}$ (follower Crows) do
z = $Rc[j]$
$MPzi_{,iter} = \begin{cases} 1 & \frac{w_{i,iter} - Best\_w(iter)}{w_{j,iter} - w_{i,iter}} > rand() \wedge w_{j,iter} > w_{i,iter} \\ 0 & otherwise \end{cases}$
$r_i > AP^{i,iter}$ and $MPzi,iter$
Yes
No
$X^{z,iter+1} = X^{z,iter} + r_z \times fl^{z,iter} \times (Mem^{i,iter} - X^{z,iter})$
A random location in search space
• Re-evaluate the fitness of each crow
• Update the Mem$_{Sol}$ and Best$_{Sol}$
Step10: Extract the best parameters C and γ
Step11: Re-train the SVM
• Re-train the SVM with new parameters
• Determine the label of new samples

Fig. 3. The flowchart of the ICSA-SVM framework.

6.2. Evaluation Measurement

In order to evaluate the performance of ICSA-SVM framework and compare its results with PSO-SVM and CSA-SVM frameworks, classification performance measures including $Accuracy$, $Precision$, $Recall$ and $F-score$ are used. These criteria are calculated based on confusion matrix. Table 5 shows the confusion matrix for problem with two classes.

Table 5. The confusion matrix

		The predicted label	
		Positive	Negative
The actual label	Positive	TP	FN
	Negative	FP	TN

If the data is considered in two classes of Positive and Negative:
TP: Data that belongs to the Positive class and is correctly categorized.
FN: Data that belongs to the Positive class but incorrectly is assigned to the Negative class.
TN: Data that belongs to the Negative class and correctly is categorized.
FP: Data that belongs to the Negative class but incorrectly is assigned to the Positive class.
These criteria are defined as follows:

$Accuracy$ is the degree of correct detection of the classifier in a total of two classes. This parameter actually indicates the amount of patterns that have been correctly detected.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (15)$$

$Precision$ criterion evaluates the ratio of the number of correct predictions made for samples of a particular class to the total number of predictions for samples of that particular class (this number includes the sum of all true and false predictions).

$$Precision = \frac{TP}{TP+FP} \quad (16)$$

$Sensitivity$ ($Recall$) means the proportion of correct items that the classifier has correctly identified as the correct sample.

$$Sensitivity(Recall) = \frac{TP}{TP+FN} \quad (17)$$

$\mathrm{F-score}$ is an appropriate criterion to evaluate the performance of a classifier. This criterion considers $precision$ and $recall$ together. This criterion is 1 in the best case and 0 in the worst case.

$$F-score = \frac{2*(Precision*Recall)}{Precision+Recall} \quad (18)$$

6.3. Experiment

Five experiments are done to analyze the performance of the proposed algorithm.

(1) The first experiment:

In order to evaluate the performance of the proposed algorithm, $Accuracy$ criterion of the ICSA algorithm is compared with PSO and CSA algorithms on the Iris, Sonar, Vehicle and Wine datasets.

The obtained results were compared with results of research [33]. Moreover, in order to compare fairly the results of the PSO-SVM, CSA-SVM and ICSA-SVM frameworks, the number of crows and particles is adjusted to 20, the number of algorithm iterations to 2000, the number of objective function evaluation to 40000 and the number of algorithm runs to 30. To evaluate the results of these algorithms, after classification, the $Accuracy$ criterion is compared.

Table 6. $Accuracy$(%) comparison of the PSO-SVM, CSA-SVM and ICSA-SVM frameworks,

Dataset	PSO-SVM[33]	CSA-SVM	ICSA-SVM
Iris	98.00	97.17	99.58
Sonar	88.32	91.11	97.15
Vehicle	88.71	90.42	95.32
Wine	99.56	98.3	100

According to the Table 6, the results of ICSA-SVM framework are more appropriate. For example, the values of $Accuracy$ in the Iris dataset for the PSO-SVM, CSA-SVM, and ICSA-SVM frameworks are 98, 97.17, and 99.58, respectively. The simulation results show that the ICAS algorithm has better performance.

(2) The second experiment:

In this experiment, the ISSDA dataset is used and the evaluation of the proposed ICSA-SVM framework is compared to the PSO-SVM and CSA-SVM frameworks based on $Accuracy$, $Precision$, $Recall$ and $F-score$ criteria. The more values of these criteria, is demonstrated that the proposed framework has higher performance in detection of fraudulent customers. Also, the value of fl for CSA and ICSA algorithms are adjusted 2, and in the PSO algorithm, the constants $C1$ and $C2$ are set to 2, and the inertial weight decreases linearly throughput the iterations from 0.4 to 0.9.

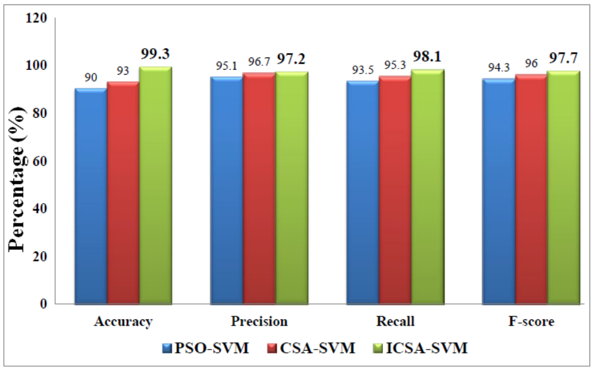


Fig. 4. Comparing the results of ICSA-SVM, PSO-SVM and CSA-SVM frameworks on ISSDA .

The 10-flod cross validation search method is used to calculate the parameters C and γ, and the optimal values of parameters C and γ that yield the high performance are 10 and 1, respectively. According to Fig.4, the value of $Accuracy$ for ICSA-SVM is 99.3 which is more perfectible than the other two frameworks. The value of $F-score$ criterion for ICSA-SVM framework is 97.7 however for those of PSO-SVM and CSA-SVM frameworks are 94.3 and 96, respectively. By comparing all four criteria in the three frameworks, it was found that the performance of the proposed ICSA-SVM framework is perfectible in electricity theft detection.

(3) The third experiment:

$Accuracy$, $Precision$, $Recall$, and $F-score$ criteria are unable to estimate efficiently the model's ability when the number of samples in a class is much higher than another [34]. Hence, the receiver operating characteristics (ROC) curve is used to detect the efficiency of a classifier. In fact, the ROC curve is a measure of the performance evaluation of classification problems. For this purpose, two-dimensional diagrams with horizontal axis FPR and vertical axis TPR were used. FPR is the ratio of false positives to positive class. TPR is a proportion of positive samples that the modeler correctly identified as positive sample.

FPR and TPR are calculated as follows:

$$FPR = \frac{FP}{FP+TN} \tag{19}$$

$$TPR = \frac{TP}{TP+FN} \tag{20}$$

The Area under curve is used as a criterion for evaluating classifier performance. Ideally, the area under curve is equal to its maximum value of 1. Therefore, if the number is close to 1, it is an representation of higher performance of classification.

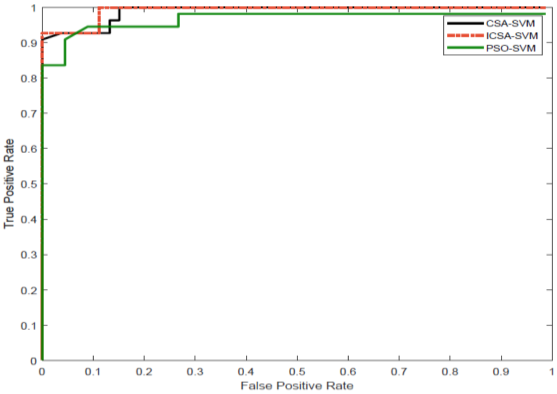


Fig. 5. ROC Criterion for PSO-SVM, CSA-SVM and ICSA-SVM frameworks on ISSDA.

In this experiment, after classification, the ROC values of the PSO-SVM, CSA-SVM and ICSA-SVM frameworks are measured and compared.

Fig.5 shows the ROC of the ICSA-SVM framework is 98.12, which is higher than the ROC values of the PSO-SVM and CSA-SVM frameworks. The ROC values of the PSO-SVM and CSA-SVM frameworks are 90 and 96.2, respectively. With comparing these results, it is clear that the proposed framework detects electricity theft with high efficiency and is acceptable for theft detection.

(4) The fourth experiment:

In this experiment, the results of the proposed framework were compared with results of Convolutional Neural Network Random Forest (CNN-RF), Gradient Boosting Decision Tree (GBDT) and SVM methods were proposed by Shuan et al. [35]. The experiment has been performed on ISSDA dataset. Fig 6 shows the comparison of results. With comparison, we found that the performance of the proposed framework was higher than research of Shuan. According to Fig 5, the $F-score$ criterion of ISSDA dataset by proposed framework is 97.7, while, for CNN-RF, GBDT and SVM classifiers are 97, 71 and 73.5 respectively.

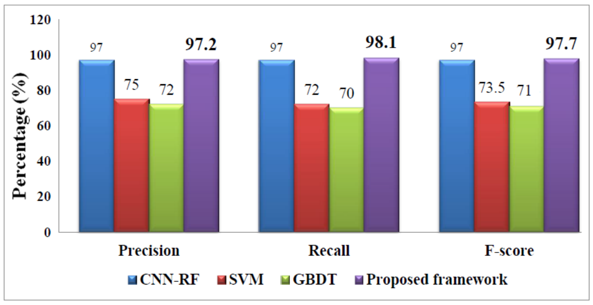


Fig. 6. Comparing the results of ICSA-SVM with CNN-RF, GBDT and SVM frameworks [35] on ISSDA dataset.

(5) The fifth experiment:

In this experiment, the $Accuracy$ of the proposed framework was compared with the results of literature articles [36] and [37] on the ISSDA dataset. In this experiment, Jokar's research attacks [12] (6 attacks) were used. According to Table 7, it can be seen the $Accuracy$ of the proposed method is higher than papers [36] and [37].

Table 7. $Accuracy$ Comparison of the proposed framework with literature research.

Method	$Accuracy$(%)
ETDFE[37]	93.36
Model1[36]	91.8
Model2[36]	90.2
Model3[36]	90.3
Proposed Framework	99.33

As can be seen, the $Accuracy$ value of the proposed framework is 99.3, which is higher than other methods.

7. CONCLUSION

In this paper, a hybrid ICSA-SVM framework for electricity theft detection was presented which used the ICSA algorithm to adjust the parameters C and γ in SVM classifier. For increasing the performance of classification, the CSA algorithm was improved. In ICSA algorithm based on weight, the weight of the crows was used to search for around the crows, and the crows with higher-weight were selected as optimal crows to search for food around them.

Therefore, the ICSA algorithm was used as an optimizer to improve the performance of SVM classifier. Then, the performance of proposed algorithm was evaluated on four reference datasets as Iris, Sonar, Vehicle and Wine. Furthermore, it yielded more preferable values in 4 criteria of $Accuracy$, $Precision$, $Recall$ and $F-score$ than the PSO-SVM and CSA-SVM frameworks. The ROC criterion for three PSO-SVM, CSA-SVM and ICSA-SVM frameworks was calculated, which proved that the performance was higher for the ICSA-SVM framework than other two frameworks. Overall, the results indicated that the proposed framework is highly efficient in identifying customers who are being manipulating their consumption patterns.

REFERENCES

[1] V. G. Vilas, M. Venkat, S. M. Bakre, and V. Velhal, **"Smart meter modelling and fault location communication in Smart Grid,"** *Majlesi Journal of Electrical Engineering*, Vol. 12, pp. 55–62, 2018.

[2] N. R. Babu, "**Smart Grid Systems: Modeling and Control**", *Apple Academic Press*; 1 edition, June 18, 2018.

[3] P. McDaniel, S. McLaughlin, "**Security and Privacy Challenges in the Smart Grid**", *IEEE Security & Privacy*, Vol. 7, No. 3, pp. 75-77, 2009.

[4] A. Hasanalizadeh-Khosroshahi, H. Shahinzadeh, "**Security Technology by using Firewall for Smart Grid**", *Bulletin of Electrical Engineering and Informatics*, Vol. 5, 2016.

[5] R. Sowndarya, V. Latha, "**An Artificial Intelligent Algorithm for Electricity Theft Detection in AMI**", *International Journal of Engineering Science and Computing*, Vol. 7, No. 3, pp. 5222- 5227, 2017.

[6] G. G. Dranka, P. Ferreira, **"Towards a smart grid power system in Brazil: Challenges and opportunities"**, *Energy Policy*, Vol. 136, 2020.

[7] E. W. S. Angelos, et al., **"Detection and Identification of Abnormalities in Customer Consumptions in Power Distribution Systems"**, *IEEE Transactions on Power Delivery*, Vol. 26, No. 4, pp. 2436-2442, 2011.

[8] E. B. Huerta, et al., **"A hybrid GA/SVM approach for gene selection and classification of microarray data"**, *presented at the Proceedings of the 2006 international conference on Applications of Evolutionary Computing, Budapest, Hungary*, 2006.

[9] X. Zhang, et al., **"An ACO-based algorithm for parameter optimization of support vector machines"**, *Expert Syst. Appl.*, Vol. 37, No. 9, pp. 6618-6628, 2010.

[10] A. Jindal, et al., **"Decision Tree and SVM-Based Data Analytics for Theft Detection in Smart Grid"**, *IEEE Transactions on Industrial Informatics*, Vol. 12, pp. 1-1, 2016

[11] S.-C. Yip, et al., **"Detection of energy theft and defective smart meters in smart grids using linear regression"**, *International Journal of Electrical Power & Energy Systems*, Vol. 91, pp. 230-240, 2017.

[12] P. Jokar, et al., **"Electricity theft detection in AMI using customers' consumption patterns"**, *IEEE Transactions on Smart Grid*, Vol. 7, No. 1, pp. 216-226, 2015.

[13] S. Li, et al., **"Electricity Theft Detection in Power Grids with Deep Learning and Random Forests"**, *Journal of Electrical and Computer Engineering*, Vol. 2019, pp. 1-12,2019.

[14] J. Nagi, et al., **"Improving SVM-Based Nontechnical Loss Detection in Power Utility Using the Fuzzy Inference System"**, *IEEE Transactions on Power Delivery*, Vol. 26, No. 2, pp. 1284-1285, 2011.

[15] J. Nagi, et al., **"Nontechnical Loss Detection for Metered Customers in Power Utility Using Support Vector Machines"**, *Power Delivery, IEEE Transactions on*, Vol. 25, pp. 1162-1171, 2010.

[16] D. R. Pereira, et al., **"Social-Spider Optimization-based Support Vector Machines applied for energy theft detection"**, *Computers & Electrical Engineerin*g, Vol. 49, pp. 25-38, 2016.

[17] M. Hasan, et al., **"Electricity Theft Detection in Smart Grid Systems: A CNN-LSTM Based Approach"**, *Energies*, Vol. 12, No. 17, p. 3310, 2019.

[18] T. Ahmad, Q. Ul Hasan, **"Detection of Frauds and Other Non-technical Losses in Power Utilities using Smart Meters: A Review"**, *International Journal of*

Emerging Electric Power Systems, Vol. 17, No. 3, pp. 217-234, 2016.

[19] A. A. Aburomman, M. B. Ibne Reaz, **"A novel SVM-KNN-PSO ensemble method for intrusion detection system"**, *Applied Soft Computing*, Vol. 38, pp. 360-372, 2016.

[20] P. Shi, et al., **"A novel intelligent fault diagnosis method of rotating machinery based on deep learning and PSO-SVM"**, *Journal of Vibro engineering*, Vol. 19, 2017

[21] J. Du, et al., **"A Prediction of Precipitation Data Based on Support Vector Machine and Particle Swarm Optimization (PSO-SVM) Algorithms"**, *Algorithms*, Vol. 10, No. 2, 2017.

[22] R. J. Kuo, et al., **"Artificial bee colony-based support vector machines with feature selection and parameter optimization for rule extraction"**, *Knowledge and Information Systems*, Vol. 55, No. 1, pp. 253-274, 2018.

[23] E. Pourbasheer, et al., **"Application of genetic algorithm support vector machine (GA-SVM) for prediction of BK-channels activity"**, *European Journal of Medicinal Chemistry*, Vol. 44, No. 12, pp. 5023-5028, 2009.

[24] Y. Prasad, et al., **"SVM Classifier Based Feature Selection Using GA, ACO and PSO for siRNA Design"**, *In: Tan Y., Shi Y., Tan K.C. (eds) Advances in Swarm Intelligence. ICSI 2010. Lecture Notes in Computer Science*, Vol 6146. Springer, Berlin, Heidelberg, pp. 307-314, 2010.

[25] P. Wang, et al., **"PSO-SVM Model Based Prediction and Analysis for the Formation of Navigation Channel Silt"**, *Applied Mechanics and Materials*, Vol. 543-547, pp. 4133-4136, 2014.

[26] E. Avci, "**Selecting of the optimal feature subset and kernel parameters in digital modulation classification by using hybrid genetic algorithm–support vector machines: HGASVM**", *Expert Systems with Applications,* Vol. 36, No. 2, Part 1, pp. 1391–1402, 2009.

[27] W. Karush, "**Minima of Functions of Several Variables with Inequalities as Side Conditions**", *Master thesis, Department of Mathematics, University of Chicago, Chicago*, IL, USA, 1939.

[28] H. W. Kuhn and A. W. Tucker, "**Nonlinear Programming**", *in Jerzy Neyman (ed.), Proceedings of the Second Berkeley Symposium on mathematical Statistics and Probability (Berkeley, U. of Calif. Press, 1950)*, 481-492.

[29] A. Askarzadeh, **"A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm"**, *Computers & Structures*, Vol. 169, pp. 1-12, 2016.

[30] M. Pardo, and G. Sberveglieri, **"Classification of electronic nose data with support vector machines"**, *Sensors and Actuators B:Chemical*, Vol. 107, No. 2, pp. 730-737, 2005/06/29/, 2005.

[31] Irish Social Science Data Archive(the smart energy data from the Irish Smart Energy Trial) [Online] Available:http://www.ucd.ie/issda/data/commissionforenergyregulationcer.

[32] UCI datasets://archive.ics.uci.edu/ml/index.php

[33] S.W. Lin, et al., **"Particle swarm optimization for parameter determination and feature selection of support vector machines"**, *Expert System. Appl.*, Vol. 35, No. 4, pp. 1817-1824, 2008.

[34] J. Huang and C. X. Ling, "**Using AUC and accuracy in evaluating learning algorithms**", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 3, pp. 299–310, 2005

[35] L. Shuan., et al., **"Electricity Theft Detection in Power Grids with Deep Learning and Random Forests"**, *Journal of Electrical and Computer Engineering*, Vol. 2019, pp. 1-12, 2019.

[36] M. Nabil., et al., **"PPETD: Privacy-Preserving Electricity Theft Detection Scheme With Load Monitoring and Billing for AMI Networks"**, *IEEE Access*, Vol. 7, pp. 96334-96348, 2019.

[37] M. Ibrahem., et al., **"Efficient Privacy-Preserving Electricity Theft Detection with Dynamic Billing and Load Monitoring for AMI Networks"**, *arXiv*, 2020.