International Journal of Electrical and Computer Engineering (IJECE)
Vol. 11, No. 5, October 2021, pp. 3704~3717
ISSN: 2088-8708, DOI: 10.11591/ijece.v11i5.pp3704-3717

Improving power theft detection using efficient clustering and ensemble classification

Hassan Ghaedi[1], Seyed Reza Kamel Tabbakh Farizani[2], Reza Ghaemi[3]
[1]Department of Computer, Neyshabur Branch, Islamic Azad University, Neyshabur, Iran
[2]Department of Computer, Mashhad Branch, Islamic Azad University, Mashhad, Iran
[3]Department of Computer, Quchan Branch, Islamic Azad University, Quchan, Iran

Article Info

Article history:

Received Aug 8, 2020
Revised Mar 28, 2021
Accepted Apr 9, 2021

Keywords:

Classification
Clustering
Crow search algorithm
Machine learning
Power smart grid
Theft detection

ABSTRACT

One of the main concerns of power generation systems around the world is power theft. This research proposes a framework that merges clustering and classification together in order to power theft detection. Due to the fact that most datasets do not have abnormal samples or are few, we have added abnormal samples to the original datasets using artificial attacks to create balance in the datasets and increase the correct detection rate. We improved the crow search algorithm (CSA) and used the weight feature of Crows to improve performance of clustering phase. Also, to create balance between diversification and intensification, we calculated the awareness probability parameter (AP) dynamically at iterations of the algorithm. To evaluate the performance, we used the cross validation technique have used the stacking technique in its training phase. The results of extensive experiments on three reference datasets showed high performance to detect power theft. The evaluation results showed that if the data is collected correctly and sufficiently, this framework can effectively detect power theft in any actual power grid. Also, for new attacks, if their patterns can be detected from the data, it is easily possible to implement these types of attacks.

Corresponding Author:

Seyed Reza Kamel Tabbakh Farizani
Department of Computer
Mashhad Branch, Islamic Azad University
Mashhad, Iran
Email: rezakamel@ieee.org

1. INTRODUCTION

Smart grid develops the traditional power grid into a new grid where it correlates too many electronic devices to each other through asterisk manager interface (AMI) [1]. In power smart grids, it is important to know about the amount of power consumed by customers in a moment. Because, it is necessary to accurately predict and plan the electricity demand in the future [2]. Power theft has become a major concern in developing countries due to the lack of metering, charges and billing infrastructure because it has imposed very high economic and technical losses [3]. Many research have been done to explore electricity theft using machine learning and data mining technologies through customer consumption patterns, but each has some kind of challenges. One of these challenges is the lack or shortcoming of abnormal consumption samples. For example, there are no or a few abnormal samples for a customer. The presence of unusual and manipulated consumption patterns can make it easier to identify incorrect customers for classification. Thus, the lack of a comprehensive dataset, which includes both normal and abnormal samples, limits the detection rate of theft. That is why a comprehensive and balanced dataset is required. Choosing the type and number of

classifiers and how to combine them is another challenge of using data mining techniques in the field of power theft detection. Therefore, a power theft detection system that can detect theft attacks is essential.

Many studies have been carried out to detect power theft using data mining techniques through customer consumption patterns, however, each of them has some challenges, as mentioned before. In research [4], Feng *et al.* used an electrical theft detection method based on local matrix reconstruction (LMR) was presented and five daily load characteristics to replace the high load amplitude daily load curves. PCA has also been used to calculate the reconstruction of weight reduction errors in a local range. In order to calculate the number of local outliers, the reconstruction error of each sample was compared with neighboring samples, which indicates the degree of abnormality of each sample. In [5], Blazakis and Stavrakakis presented a computational method of analysis and identification of customer electricity consumption patterns based on data mining techniques in order to identify unauthorized residential customers. In the mentioned method, principal component analysis (PCA) is combined with mean shift algorithm for different power theft scenarios. In research [6], Zheng *et al.* presented a density-based power theft detection method in order to find abnormal consumption patterns. The degree of abnormalities in each customer's profile was calculated based on their distance matrix. K-means, gaussian mixture model (GMM) and density spatial clustering of applications with noise (DBSCAN) techniques were used to evaluate the presented method. In research [7], Park and Kim presented a method based on detecting an abnormal pattern in order to detect power theft from received data of smart meters. The presented method used only normal consumption data for training. Creating abnormal attacks on normal data has made it possible to use the presented method in real situations. The basis for constructing detection model of the outlier data in this research was the k-means clustering technique.

In [8], Singh *et al.* offered a detection method based on PCA to identify energy theft attacks in AMI. AMI data were reconstructed using principal components and used to construct relative entropy. In the mentioned method, relative entropy was used to measure the similarity between two possible distributions derived from the reconstructed consumption database. Also, by injecting theft attacks into AMI, the distribution of energy consumption probability was deviated from previous consumptions, leading to a larger relative entropy. In [9], Xiao and Ai suggested a data-driven power theft detector based on stochastic matrix theory. One of the main steps of the presented method was to use an auxiliary matrix as a data source that shows the relationship between the power consumer and the operating system modes under abnormal power consumption conditions. In the presented research, personal and regional theft detection algorithms and abnormal consumption pattern signal have been used, the aim of which was to quickly identify areas with high probability of theft and immediate selection of customers suspected of theft. In [10], Jokar *et al.* used single-class and multi-class support vector machines to discover fraudulent customers. Also, to create a balance in the dataset, non-normal samples are added to the dataset.

In [11], Hasan *et al.* presented an electrical theft detection system based on the combination of a convolutional neural network (CNN) and a long short-term memory (LSTM) architecture. The dataset of the actual electricity consumption of Chinese customers was used in this article which is available through the state grid corporation of China (SGCC). Synthetic minority over-sampling technique (SMOT) is used to generate synthetic data. In [12], Shuan *et al.* presented a new forest-based hybrid model and convolutional neural network to help companies to address the problems of inadequate inspections and unusual electricity consumption. The SMOT technique and the CNN-RF model had been used for the generation of synthetic data classification, respectively. They used two SEAI and low carbon London (LCL) datasets to perform the experiments. Because of being harmful electricity theft to electricity providers, the importance of the integrity of information flow and energy flow, as well as the usefulness of analysis of smart grid data in detecting theft, Zheng *et al.* [13] presented a new deep and wide convolutional neural network modeling method to solve the mentioned challenges. Experiments had been performed on real power consumption data published by SGCC. Due to the importance of damages caused by electricity theft, a new plan needed to be developed that could accurately detect these thefts in complex electricity networks. In [14], Yip *et al.* used two linear regression based algorithms to investigate the energy consumption behavior of consumers and to evaluate their anomaly coefficients in order to collate the power theft caused by the manipulation of the meter as well as the broken meter.

In research [15], Rezavi offered a feature engineering framework for detecting power theft in smart grids. In this framework, a combination of finite hybrid clustering was used to segment customers and a genetic algorithm was used to identify new features that are suitable for prediction. The gradient boosting algorithm has also been used for evaluation. The aspects of detection accuracy, detection delay, identification of irregular and hidden attacks and computational complexity of the implemented algorithms have been investigated. In research [16], Zhang *et al.* presented a method based on feature engineering for unsupervised detection of abnormal consumption behavior of customers. The original data set was created by brainstorming in the feature engineering phase. Then the optimal feature set was selected based on variance

and similarity between features and then in the abnormal detection stage, the density clustering algorithm was used. In research [17], Guerrero *et al.* suggested a framework and methodology that has been developed by two coordinated modules. The first module was based on a client filtering based on text mining and a complementary neural network. The second module includes classification, regression tree and a self-organizing map (SOM) neural network. This module has tripled the success of inspections. In research [18], Ghasemi and Gitizadeh used the methods of classifying customers' energy consumption patterns based on probabilistic neural networks and the mathematical model based on the Levenberg-Marquardt method to identify illegal customers. Also, the effect of distributed generation sources on illegal energy consumption has been evaluated and the proposed discovery algorithm has been modified. In research [19], Buzau *et al.* presented a methodology for detecting non-technical losses using supervised learning. The presented methodology was evaluated on the data of real smart meters of all endesa's customers. This methodology used all the information recorded by smart meters in order to deeply analyze the consumer behavior of customers. In presented research, auxiliary databases have been used in order to provide additional information about the geographical location and the characteristics of each meter. Several classifiers have been used, and the performance of the boosted gradient trees has been better than the others. In [20], Junior *et al.* used the optimum-path forest (OPF) clustering algorithm to identify regular and irregular profiles of customers of a Brazilian electricity company. Furthermore; a model for the NTL detection problem as an anomaly detection task in cases where there is insufficient information about unauthorized customers or this information is scarce, and two OPF-based approachs to detect unauthorized customers were presented.

Although many research have been done for theft detection using data mining, the challenges still are there. In this research, in order to strengthen and increase the efficiency of power theft detection, we have presented a framework based on optimal clustering and classification of data. The purpose of applying clustering in this framework is to augment and increase the classification detection rate based on the correlation and similarity that exists between the data of each cluster. We used cluster numbers to label the data. For optimal clustering, the improved crow search algorithm (ICSA) is used in combination with the k-means algorithm. In ICSA algorithm, unlike the basic crow search algorithm (CSA), the movement and new locations of each crow were done based on the weight feature or the attraction power of each crow. The weight feature of Crows increases the power of the CSA algorithm to find the optimal cluster centers. One of the important things to consider when using metaheuristic algorithms is to balance between diversification and intensification. Diversification focuses on producing a variety of solutions to explore the search space on a global scale, and intensification focuses on searching in a local area. Given the importance of this balance, correct adjustment of the parameter *AP* in the CSA algorithm is important. In the CSA algorithm, the parameter *AP* has a fixed value for all iterations, but in the ICSA algorithm, this parameter was calculated based on the proportion that exists dynamically between the fitness of each crow and the worst fitness of that iteration, which caused a balance between diversification and intensification. Therefore, the proposed algorithm enabled Crows to balance between global search and its temporary memory to achieve the optimal solution. Stacking technique is a high-level technique to achieve more accuracy that has also been used to model the data. In this technique, the results of a set of different classifiers at 0-level were combined by a Meta classifier at 1-level. Its purpose was to optimally integrate the decisions of the basic classifiers. The K-flod cross validation technique was used to evaluate the performance of the proposed framework and in order to ensure the optimal distribution of classes in random selected subsets of the cross validation technique, the stratified sampling technique was used to select the samples. At level-0 the basic support vector machine (SVM), logistic regression (LR) and k-neighbors classifier (KNN) algorithms were used and at level-1 the naive bayes algorithm due to the nature of flexible non-parametric, providing predictive distribution and simple and effective learning process was used as the meta algorithm.

This research was organized is being as: The PSO algorithm was discribed in section 2. The CSA algorithm was reviewed in section 3. Section 4 described the ICSA algorithm. The proposed framework for theft detection was described in section 5. Examinations and evaluations were discussed in section 6 and 7 was dedicated to the research conclusions.

2. PARTICLE SWARM OPTIMIZATION (PSO)

Particle swarm optimization algorithm [21] is one of the important algorithms of collective intelligence that is designed based on the social behavior of birds and has been used to discover patterns in the simultaneous flight of birds, their sudden change of direction and optimal transformation of flocks. In their flight, complex behaviors are visible when moving. Particles learn from each other and move towards their best neighbors based on the knowledge gained. Each particle adjusts its location in the search space according to the best location it has ever been in and the best location in its entire neighborhood. The position of the particles is updated by adding new velocity to the current position. These steps are repeated several

times until the desired answer is obtained. Particles move in n-dimensional space and change their path in search space based on their past experiences and those of other particles. In a group of *n* particles, the position of the i[th] particle is affected by a n-dimensional spatial vector according to (1).

$$X_i = (x_{i1}, x_{i2}, \dots, x_{in}) \quad (1)$$

In addition, the particle *i* has a velocity vector is being as (2):

$$V_i = (v_{i1}, v_{i2}, \dots, v_{in}) \quad (2)$$

The best position for particle *i* is obtained is being as (3):

$$P_i = (p_{i1}, p_{i2}, \dots, p_{in}) \quad (3)$$

Finally, the new position of the particles is obtained using (4) and (5):

$$V_i(t+1) = wV_i(t) + c_1 r_1\big(P_i(t) - X_i(t)\big) + c_2 r_2\left(P_g(t) - X_i(t)\right) \quad (4)$$

$$X_i(t+1) = X_i(t) + V_i(t) \quad (5)$$

where *g* is the index used for the particle that has the best position, *t* represents the number of repetitions, *w* the coefficient of inertia, c_1 and c_2 the learning coefficients, r_1 and r_2 the random numbers in the range 0 and 1, which cause a variety of answers. c_1 is the learning factor related to the personal experiences of each particle and c_2 is the learning factor related to the total experiences of the group.

3. CROW SEARCH ALGORITHM

Crows are today considered among the most intelligent animals in the world and, watch the other Crows to see where the other Crows are hiding their food. When a Crow leaves it's location, they steal it's food. If a Crow has committed theft, it will take extra precautions, such as moving food hideouts to prevent future victim. The Crow search algorithm is based on four principles, namely [22], [23]:

- Crows live in a group.
- Crows remember their secret location.
- Crows follow each other to steal.
- Crows protect their food hideouts from being stolen by a probability.

CSA algorithm is a metaheuristic optimization algorithm, which is based on intelligent behavior of Crow that was described by Askarzadeh [24]. Suppose, there is a *D* dimension environment containing some Crows. The number of Crows is *N* and the location of the Crow in the iteration it in the search space is specified by the following vector:

$$L^{i,it}\ i = 1, 2, \dots, N\ it = 1, 2, \dots, it_{max}$$

$$L^{i,it} = [L^{i,it}1, L^{i,it}2, \dots, L^{i,it}n]$$

where it_{max} is the maximum number of iterations. Each Crow has a memory in which its hideout location is stored. In each iteration, the location of Crow *j* is indicated by $Mem^{i,it}$. This is the best situation that Crow *i* has ever achieved. Also, in the memory of each Crow, its best experience is stored. Crows move around and look for better food sources. Suppose that in the iteration *it*, Crow *j* wants to visit his hideout ($Mem^{i,it}$), in this iteration, Crow *i* decides to follow Crow *j* in order to reach the hideout of Crow *j*. There may be two states:

State 1: Crow *j* does not know that Crow *i* is following it. As a result, Crow *i* reaches the hiding location of Crow *j*. In this state, the new location of is obtained is being as (6).

$$L^{i,it+1} = L^{i,it} + r_i \times fl^{i,it} \times (Mem^{i,it} - L^{i,it}) \quad (6)$$

Where r_i is a random number with uniform distribution between 0 and 1, and $fl^{i,it}$ denotes the flight length of Crow i in iteration it.

State 2: Crow j knows that Crow *i* is following it. As a result, in order to protect its food hideout from being stolen, Crow *j* confuses Crow *i* by going to another location of the search space. Taken together, the location $L^{i,it+1}$ can be formulated is being as:

$$L^{i,it+1} = \begin{cases} L^{i,it} + r_i \times fl^{i,it} \times (Mem^{i,it} - L^{i,it}) & r_j > AP^{j,it} \\ \text{a random location} & \text{otherwise} \end{cases} \quad (7)$$

Where r_j is a random number with uniform distribution between 0 and 1 and the parameter of awareness probability ($AP^{j,it}$) denotes the awareness probability of Crow j in iteration it. By reducing the probability of awareness, CSA tends to direct the search to a local area that indicates a good solution is found in the area. As a result, by increasing the values of AP, the local search probability of good solutions is reduced and CSA tends to follow the search space on a global scale.

4. IMPROVED CROW SEARCH ALGORITHM (ICSA)

Population-based algorithms increase the probability of finding a good solution and escaping local optimum. Unlike the algorithms PSO, ant colony optimization (ACO), harmony search (HS) and genetic algorithm (GA), which require setting of 4, 4, 3, 6 parameters, respectively, the ICSA algorithm only needs to set 2 parameters. Due to the low diversity of PSO and ACO algorithms and falling into the trap of local optimization as well as the challenge of constantly updating pheromone in the ACO algorithm in order to balance diversification and intensification, we have improved the basis CSA algorithm. In CSA algorithm, each member of population was considered to be a optimal solution, and the food location was the optimal location that a Crow had obtained in its searches. Suppose, the CSA algorithm has an initial population of N members that each member has a fitness value in terms of the objective function of the problem. Admittedly, any member of the population who has been able to obtain more optimal locations in the search space is more qualified. In other words, any Crow that optimizes the objective function has a greater ability to collect food and hide it. The more optimal Crows or solutions access to more food sources and are closer to the optimal solution. In fact, each Crow or the optimal solution has a greater ability to attract other Crows. Hypotheses of proposed method for improving the Crow search algorithm using weighting mechanism:

- Each Crow is assumed to be a solution to the problem and its fitness value is expressed in terms of the objective function of the problem.
- Each Crow has a fitness weight to be considered by other Crows, where weight is determined by the fitness value of the best and worst population Crows in each iteration.
- Each Crow is considered by the other Crows according to their fitness and weight values.

In each iteration of ICSA algorithm, each Crow that is better in weight than the others is noticed by more Crows in the flock, and the rest of Crows fly to it. This means that in each iteration of the ICSA algorithm, the best Crows (locations) found so far are used to arrive at better solutions. For weight-based searches require the weight of each Crow be determined according to its fitness and here, the weight indicates the importance of a Crow to perform local search around it. To determine the weight of each Crow in iteration it, it is necessary to determine *Best* and *Worst* is being as (8, 9).

$$Best_{it} = {}_{i=1}^{n}minFit(Crow_{i.it}) \quad (8)$$

$$Worst_{it} = {}_{i=1}^{n}maxFit(Crow_{i.it}) \quad (9)$$

where, fit ($Crow_{i,\ it}$) is fitness value of Crow i in iteration it. *Best* and *Worst* variables are the best and worst member of Crows respectively. This weight is calculated in iteration it according to (10).

$$w_{i,it} = \frac{Worst_{it} - Fit(Crow_{i,it})}{Worst_{it} - Best_{it}} \quad (10)$$

In proposed method, N_{max} is the number of searches around Crow i and according to the weight of Crow i, the number of search operations around it is performed according to (11).

$$S_{Crow_{i,it}} = N_{max} \times w_{i,it} \quad (11)$$

where, $S_{Crowi,it}$ specifies Crow i will attract multiple members of the population to search around it.

In this research to increase the detection rate, the parameter AP is calculated based on fitness value of the candidate Crow. To improve the parameter AP, the idea of Karaboga and Basturk is used [25]. The parameter AP in iteration it is calculated is being as (12).

$$AP^{i.it} = \frac{Fit(Crow_{i,it})}{Worst_{it}} \quad (12)$$

where, $Worst_{it}$ illustrates the worst fitness value that is seen so-far in iteration *it*. (12) adjusts the *AP* parameter dynamically.

In fact, it causes the value of *AP* in the ICSA algorithm not to be unilaterally large or small and always stays in balance. The result of this balance is a balance between intensification and diversity. In fact, the parameter *AP* causes Crows to seek solutions sometimes in the local area and sometimes in the wider area.

5. THE PROPOSED FRAMEWORK

In this section, a four-phases framework was presented to detect power theft. Figure 1 shows the phases of proposed framework.

– phase 1: Preprocessing

At certain hours of the day, data recording by a smart meter may be disrupted and empty values were recorded. If during a day, the number of empty values was high, the record for that day is removed from the dataset, and in cases where the number of empty values is low, by averaging the values of local neighbors during a day, the new value is replaced. Also, in this phase, data normalization is performed to increase performance. Normalization before clustering is required due to the use of distance metrics such as Euclidean distance that is sensitive to the difference of the attribute scale. In this phase, the min-max normalization [26] method was used to normalize the data.

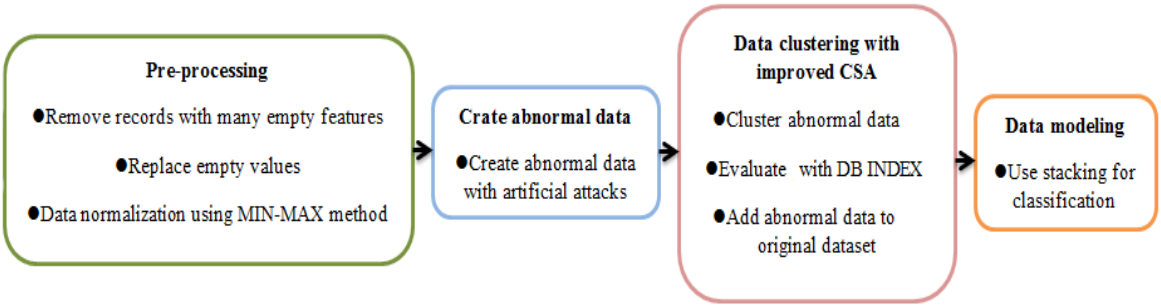


Figure 1. The phases of proposed framework

– phase 2: Creating abnormal samples

Since datasets often contain the consumption pattern of normal users, artificial attacks were added in order to balance the datasets and increase the detection accuracy of classifiers. In this phase, artificial attacks were used and abnormal samples of each customer created. Artificial attacks were made using the formulas that are presented in Table 1. For example, the sampling rate in SEAI dataset was computed per hour. Each sample was represented as $x = x_1, \ldots, x_{24}$.

Table 1. Artificial attacks to generate customers' abnormal consumption samples in SEAI dataset

Attack	Formula	Description
$f_1(x_t)$	$a \times x_t$ a=random(0.1,0.8)[10]	A random constant α is multiplied to all attributes
$f_2(x_t)$	$a_t \times x_t a_t$=random(0.1,0.8)[10]	A random amount is Multiplied to each attribute
$f_3(x_t)$	$a_t \times$ mean(x)a_t=random(0.1,0.8) [10]	For each hour, a random amount is multiplied to the average readings of a day
$f_4(x_t)$	mean(x)[10]	Average readings a day is sent each hour
$f_5(x_t)$	x_{24-t} [10]	Changing the readings of a day
$f_6(x_t)$	min(x)	For each hour, the lowest daily consumption is sent
$f_7(x_t)$	a ×mean(x) a=random(0.1,0.8)	For each hour, a random constant α is multiplied to mean(x)
$f_8(x_t)$	x_t-min(x)	Minimum consumption for whole a day is reduced from each hour

According to Table 1, for attacks $f_1(.)$ and $f_2(.)$, the customer focuses on deducting the amount of consumption per hour during the day. For attacks $f_3(.)$, f4(.) and $f_7(.)$, the customer focuses on sending the average consumption during the day. In attack $f_5(.)$, the customer reverses the consumption values in the hours of a day. The basis of attacks $f_6(.)$ and $f_8(.)$ is sending the minimum consumption of a day.

– phase 3: Clustering with ICSA

What is the purpose of clustering? The purpose of applying clustering in this framework is to strengthen and increase the classification detection rate based on the correlation and similarity that exists between the data of each cluster. Efficient clustering increases the sensitivity of the classifier for better prediction. In this phase, for optimal clustering, the improved Crow metaheuristic algorithm is used in combination with the K-means algorithm. Unlike CSA algorithm, in ICSA algorithm, the movement and new locations of each Crow are done based on the weight feature or the attraction power of each Crow. Using of the weight feature increases the power of the CSA algorithm to find the optimal cluster centers. the algorithm ICSA with was used to cluster normal and abnormal samples. In this phase, at first, the parameters of the ICSA algorithm were adjustmented and the initial location of each Crow was randomly determined and their fitness value calculated. Personal memory of Crows and best global solution were updated. In each iteration, according (11), the number of Crows that can search for food around a Crow was determined. The locations of these Crows were updated according (7) and cost functions calculated. According cost functions, the values of personal memories and best global solutions were updated again. In fact, Crows with more weight have the ability to attract more Crows , and more searches are done around them for a better solution, and the algorithm converges more quickly. Due to the importance of the balance between variation and intensification, in this phase, in each iteration, the parameter *AP* was adjusted dynamically. Also, Davies & Bouldin (DB) index [27] was used for fitness function that was explains is being as (13):

$$DB = \frac{1}{n}\sum_{i=1,i\neq j}^{n} max\left\{\frac{S_n(Q_i)+S_n(Q_j)}{S(Q_i,Q_j)}\right\} \quad (13)$$

Where n is the number of clusters, S_n is the mean Euclidean distance of the cluster data from the cluster center and $S(Q_i,Q_j)$ is the distance between the centers of the cluster. Therefore, when the inside of the clusters isclose together and the clusters are far from each other, this ratio decreases. The lowest value of the *DB* index indicates optimal clustering.

– phase 4: Data modelling

In this phase, data modeling was performed based on a combination of the results of several basic classifiers. In our research, stacking technique [28] was used to theft detection that is a two-levels technique. In the level-0, classifiers LR, SVM and KNN were used and in the level-1, the Naive Bayes classifier due to the nature of flexible non-parametric, providing predictive distribution and simple and effective learning process was used.

6. EXPERIMENTS AND ANALYSIS

In this section, according to the proposed framework, extensive experiments were conducted in each phase, and the results analyzed. Furthermore, the results of the proposed framework were compared with the literature research.

6.1. Dataset

In this research, clustering and classification of data has been performed based on the data of references datasets sustainable energy authority of Ireland (SEAI) [29], RAW electricity consumption [30] and low carbon London (LCL) [31]). In fact, in both clustering and classification phases, the consumption data of customers that stored in these datasets have been used. These features are based on kWH. The SEAI dataset contains power consumption data for 5,000 meters of Irish customers. Data were stored every half hour during the day. The RAW dataset contains the electricity consumption data of 42,372 electricity customers within 1,035 days (from Jan. 1, 2014 to Oct. 31, 2016) that 42,372 customers with daily power consumption, of which 38,757 customers were labeled as normal customers and 3,615 customers labeled as abnormal customers. The LCL dataset contains 525 days of power consumption information for 5,500 customers, of which 1,200 have been labeled abnormal. For example, Table 2 shows features of a consumption record for a meter, including the meter ID, Consumption for each hour (in kWh), and consumption date.

Table 2. Features of a consumption record for a meter

Date	Clock1 (kWh)	Clock2 (kWh)	Clock3 (kWh)	…	Clock22 (kWh)	Clock23 (kWh)	Clock24 (kWh)
19/June/2009	0.052	0.172	0.061	…	0.087	0.164	1184

6.2. Evaluation measurement

Samples are divided in two Positive and Negative classes in binary classification. The results of a binary classification are classified into four categories. Table 3 represents the confusion matrix of binary classification.

Table 3. The confusion matrix

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

In this Table, true positive (TP) is positive data that was correctly categorized. False negative (FN) is positive data that incorrectly assigned to the negative class, true negative (TN) is negative data that correctly categorized and false positive (FP) is negative data that incorrectly attributed to the positive class. In order to evaluate the performance of classification algorithms, performance criteria including $Accuracy = (TP + TN)/(TP + TN + FP + FN)$, $Precision = TP/(TP + FP)$, $Recall = TP/(TP + FN)$ and $F - score = 2 * Recall * Precision/(Recall + Precision)$ were used.

Another important criterion for evaluating the performance of a classifier is the ROC curve, which is the balance between sensitivity and specificity. The ROC curve is a two-dimensional curve that X-axis is false positive rate ($FPR = 1 - specificity = FP/(FP + TN)$) and Y-axis is true positive rate ($TPR = TP/(TP + FN)$). The area under the curve is called AUC. The numerical value of AUC is a number between zero and one and indicates the detection power of classifier. If this number is close to one, it means that the TPR value is high.

6.3. Experiment

In this section, the experiments related to the proposed framework were explained. After data normalization with min-max method, the abnormal data were created. For example, in SEAI dataset, there were 535 normal samples for each customer. Using attacks of Table 1, 4280 abnormal samples were created. The normal and abnormal data were clustered with ICSA-CL algorithm. Evaluation was performed by DB index. Crows with more weight cause more Crows to search for food around them and costs of achieving the final solution are drastically reduced. In fact, (10) and (12) were caused faster convergence and drastic reduction of costs. As shown in Figure 2, by increasing the number of iterations, the ICSA algorithm has the lowest DB compared to K-means, PSO and CSA algorithms for normal and abnormal data. According to Figure 2(a), the values of DB index of K-means, PSO, CSA and ICSA-CL algorithms for abnormal data are 0.621, 0.6010, 0.523 and 0.354 respectively. Also, according Figure 2(b), the values of DB index of K-means, PSO, CSA and ICSA-CL algorithms for normal data are 0.56, 0.6388, 0.3717 and 0.215 respectively. With the values of obtained DB index, the normal and abnormal data were divided into 2 and 3 clusters respectively and the cluster number indicates the class label of each data.

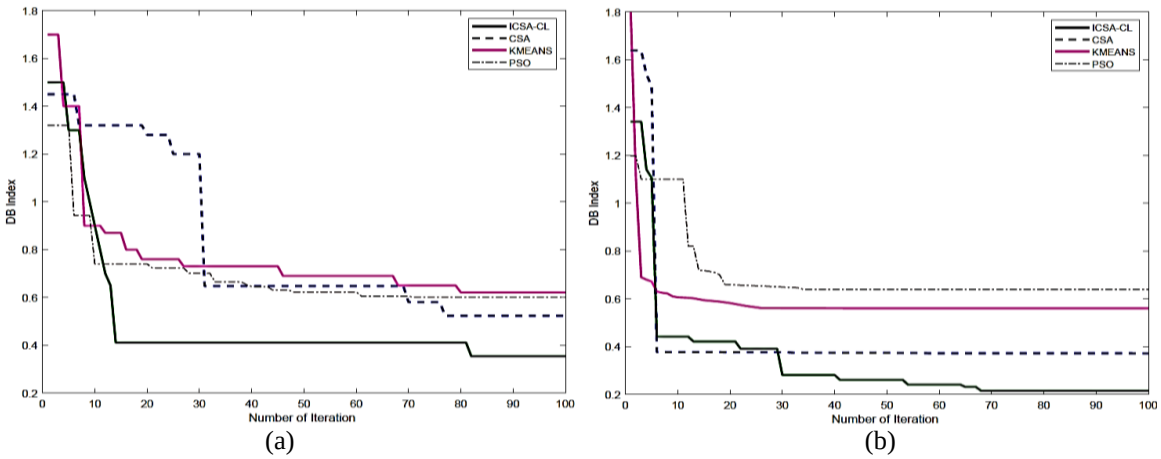


Figure 2. The values of DB index for K-means, PSO, CSA and ICSA-CL algorithms, (a) DB index for abnormal data, (b) DB index for normal data

Table 4 shows the optimal clusters for K-means, PSO, CSA and ICSA-CL algorithms that were obtained for normal and abnormal data. After data clustering, it is necessary to build a model for classification of new data. The stacking ensemble technique was used in the proposed framework. Before using this technique, Accuracy of the basic Naıve Bayes, SVM, linear regression (LR) and KNN classifiers was calculated on SEAI, Raw and LCL datasets separately. Table 5 elucidates the values of this criterion for Naive Bayes, SVM, LR and KNN classifiers. The accuracy of the Naive Bayes classifier in all three datasets was higher than the other classifiers. Therfore, this classifier is used as a Meta classifier of the stacking technique.

To evaluate the performance of the proposed framework, various experiments were performed separately on SEAI, RAW and LCL datasets. Two plans with two scenarios were executed on these datasets. In the first plan, each attack was evaluated separately and in the second plan, all of attacks were evaluated together. Also, in the first scenario, 80% of the data was for training and 20% for testing, and in the second scenario, 70% of the data was for training and 30% for testing.

Table 4. The optimal clusters that obtained with DB index

Algorithm / Dataset	K-means	PSO	CSA	ICSA-CL
Normal data	2	3	3	2
Abnormal data	2	4	3	3

Table 5. The Accuracy (%) of Naive Bayes, SVM, LR and KNN classifiers

Classifier / Dataset	Naive Bayes	SVM	LR	KNN
SEAI	89.67	73	61	78.7
RAW	82.33	78.36	78.76	79.13
LCL	90.17	83.51	83.35	78.08

6.3.1. Experiment first plan on SEAI dataset

There are 535 normal consumption samples for each customer in the SEAI dataset. In this plan, the accuracy, recall, precision and F-score criteria of proposed framework were calculated separately for each Attack (i) that i is number of attack (i=1,...,8). In scenario 1 of this plan, for each attack, there were 535 normal samples and 535 abnormal samples, of witch 20% (107 samples) of both normal and abnormal samples were considered as testing data and 80% (428 samples) as training data. Also, in scenario 2 of this plan, 30% (161 samples) of both normal and abnormal samples were considered as testing data and 70% (374) as training data. After identifying training and testing samples, the proposed framework will be experimented on them.

Table 6 shows the accuracy, recall, precision, F-score and appropriate use criteria (AUC) criteria of proposed framework for each attack separately using two mentioned scenarios. According to Table 6, the proposed framework, with high efficiency, detects power thefts in any attack. Based on the AUC values, it is observed that the efficiency of the proposed method is acceptable. For example, the AUC values of attack 1 in scenarios 1 and 2 are equal to 99.03 and 97.86, respectively, or attack 6 in scenarios 1 and 2 are equal to 99.63 and 97.15, respectively which indicates the high performance of the proposed framework.

Table 6. The accuracy, recall, precision, F-score and AUC criteria of each attack

Attack (i) / Criterion (%)	Attack1		Attack2		Attack3		Attack4	
Accuracy	97.19	96.58	97.66	97.20	98.59	95.96	98.13	96.58
Recall	98.10	96.39	97.56	96.14	98.62	95.77	97.96	96.52
Precision	98.05	96.67	97.59	96.01	98.52	95.92	98.48	96.43
F-score	98.07	96.52	97.57	96.07	98.56	95.84	98.21	96.47
AUC	99.03	97.86	98.46	98.15	99.24	97.81	99.34	97.15
Scenario	Scn1	Scn2	Scn1	Scn2	Scn1	Scn2	Scn1	Scn2
Attack (i) / Criterion (%)	Attack5		Attack6		Attack7		Attack8	
Accuracy	99.06	95.34	99.53	95.34	100	95.56	99.53	94.72
Recall	99.24	95.02	99.42	95.14	100	95.39	99.42	94.15
Precision	98.88	95.15	99.47	95.07	100	95.49	99.65	94.16
F-score	99.05	95.08	99.44	99.10	100	95.43	99.53	94.15
AUC	99.48	97.56	99.63	97.15	100	97.62	99.36	96.25
Scenario	Scn1	Scn2	Scn1	Scn2	Scn1	Scn2	Scn1	Scn2

Figure 3 shows the ROC curves of two scenarios in this plan. In all ROC curves, X-axis and Y-axis are (FPR) and (TPR) respectively. According to these curves, scenario 1 has a better performance than scenario 2 due to the increase in the number of training samples.

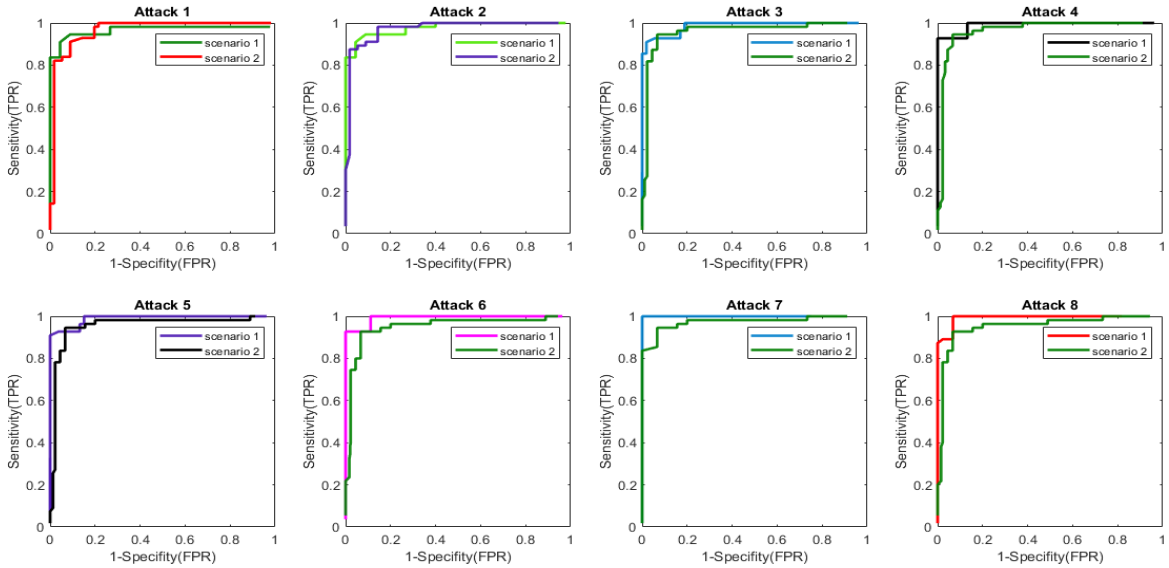


Figure 3. The ROC curves for each attack in two scenarios

6.3.2. Second plan on SEAI dataset

In this plan, all of attacks in Table 1 were applied simultaneously to 535 normal samples, which, the number of each normal and abnormal samples was equal to 3280 samples. In scenario 1 of this plan, 20% (856 samples) of both normal and abnormal samples were considered as testing data and 80% (3224 samples) as training data. Also, in scenario 2 of this plan, 30% (1288 samples) of both normal and abnormal samples were considered as testing data and 70% (2992 samples) as training data. Table 7 shows the accuracy, recall, precision and F-score criteria of the proposed framework for this plan. According Table 7, the values of accuracy, recall, precision, F-score and AUC criteria demonstrate that proposed framework is efficient when all attacks were sent to the system together. Also, the values of AUC in scenarios 1 and 2 were equal to 99.71 and 97.92, respectively which show the high efficiency of the proposed framework to detect power theft.

Table 7. The accuracy, recall, precision and F-score and AUC criteria of all 8 attacks together

Criterion (%) / Scenario	Accuracy	Recall	Precision	F-score	AUC
Scn1	99.41	99.41	99.39	99.39	99.71
Scn2	97.08	96.68	97.01	96.84	97.92

6.3.3. Second plan on RAW dataset with scenario 1

In this experiment, the performance of the proposed framework was evaluated using all attacks on the RAW dataset with scenario 1. After applying all 8 attacks, 20% of both normal (62001) and abnormal (62734) samples as testing data and 80% of both normal (248045) and abnormal (250937) samples as training data were selected. Table 8 represents the accuracy, recall, precision, F-score and AUC criteria of this plan. The results of this Table also show the high performance of the proposed framework. According to this Table, the AUC of the proposed framework in this plan is 99.72, which indicates the high ability to detect theft.

Table 8. The accuracy, recall, precision and F-score and AUC criteria of all 8 attacks together

Criterion (%) / Scenario	Accuracy	Recall	Precision	F-score	AUC
Scn1	99.63	99.59	99.61	99.59	99.72

6.3.4. Second plan on LCL dataset with scenario 1

The LCL dataset contains 4300 normal and 1200 abnormal customers. After applying all 8 attacks on normal samples, the total number of normal and abnormal samples was 34400 and 35600, respectively. 20% of both normal (62001) and abnormal (62734) samples as testing data and 80% of both normal (248045) and abnormal (250937) samples as training data were selected. In this experiment, the performance of the roposed framework was evaluated using all attacks on the LCL dataset with scenario 1. Table 9 displays the accuracy, recall, precision, F-score and AUC criteria of this plan.

According to Table 9, the performance evaluation criteria for this plan have high values, which indicates the high efficiency of the proposed framework to detect power theft. In another experiment, the results of the proposed framework were compared with Jokar *et al.* [10]. These results were shown in Table 10. In one of the experiments, only normal samples and in another experiment only the f2 (xt) attack see Table 1 were used.

Table 9. The accuracy, recall, precision and F-score and AUC criteria of all 8 attacks together

Criterion (%) / Scenario	Accuracy	Recall	Precision	F-score	AUC
Scn1	99.35	99.34	99.35	99.34	99.51

Table 10. The results of proposed framework and Jokar *et al.* [10]

Experiment	DR (%)	FPR (%)
Using normal samples for training [10]	76	29
Using normal samples for training in proposed algorithm	95	8
Only $h_3(x_t)$ attack was used for training [10][a]	86	16
Only $f_2(x_t)$ attack was used for training in proposed algorithm	98	1

[a] $h_3(x_t))=y_tx_t$, =random (0.1, 0.8), $x=\{x_1,...,x_{24}\}$, t=1,...,24

According to the results of Table 10, in the case of normal samples for training, the detection rate (DR) and false positive rate FPR values in the proposed framework are 95% and 8%, respectively, however, those were reported by Jokar *et al.* were 76% and 29%, respectively. Moreover, it can be evident that the DR and FPR values are 86% and 16%, respectively, for the case that only h3 (xt) was used for training by Jokar *et al.*, while those in the proposed framework were 98% and 1%, respectively. The results of Table 10 show that the proposed framework has high DR and low FPR. Another experiment is the comparison of the results of the proposed framework with Zheng [6] research. In this experiment, attacks 2 to 5 of Table 11 were performed on the SEAI dataset and the accuracy criterion was compared.

Table 11 shows comparison of the results of this experiment, which proves that the accuracy of the proposed framework for attacks 2, 3 and 5 is higher. In the proposed framework, the sampling rate has been reduced, which has increased customer privacy. For this purpose, in another experiment, the accuracy criterion of the proposed framework was compared with the results of literature research [32] and [33] on the SEAI dataset. In this experiment, Jokar's research attacks [10] (6 attacks) were used. According to Table 12, it can be seen the Accuracy of the proposed method in both scenarios with customer privacy, is better than other methods.

Table 11. Accuracy comparison of proposed framework with literature research

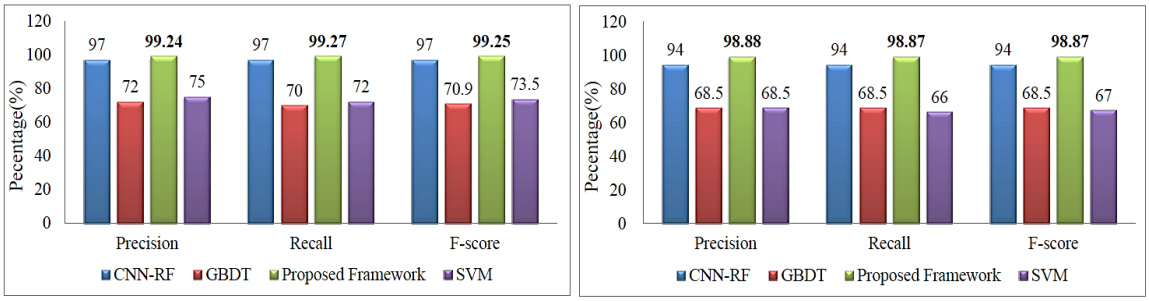
Attack / Research	Attack2	Attack3	Attack4	Attack5
Zheng [6]	92.70	93.30	99.50	90.30
Proposed framework	97.66	98.59	98.21	99.06

Table 12. Accuracy comparison of the proposed framework with literature research

Research / Criterion	ETDFE [33]	Model1 [32]	Model2 [32]	Model3 [32]	Proposed framework
Accuracy (%)	99.36	91.80	90.20	90.30	Scn1=98.35 Scn2=97.06

In the last 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

algorithms were proposed by Shuan *et al.* [12]. The experiment has been performed on SEAI and LCL datasets. Figure 4 shows the comparison of results on SEAI and LCL datasets. With comparison, we found that the performance of the proposed framework was higher in both datasets. According to Figure 4(a), the F-score criterion for SEAI dataset by proposed framework is 0.99.25, while, for CNN-RF, GBDT and SVM classifiers are 97,70.98 and 73.46 respectively. Also, for the LCL dataset in Figure 4(b), the F-score criterion for algorithms CNN-RF, GBDT, SVM classifiers and proposed framework are 94, 68.5, 67.22 and 98.87 respectively. In fact, the clustering phase greatly reduces the time of training the stacking technique by optimally clustering the data and eliminating inefficient clusters. Therefore, according to the proposed framework, the percentage of accurate prediction of customers who manipulate their metrics and do not send their actual consumption data to the center has improved.



(a) (b)

Figure 4. Comparison of the proposed framework with others algorithms in, (a) SEAI dataset, (b) LCL dataset

7. CONCLUSION

In this research, a four-phases framework was proposed to theft detection. In the first phase, data normalization was performed. Since, there is no reference dataset covering both normal and abnormal consumptions of customers, in second phase, abnormal samples were created using 8 artificial attacks. In the third phase, the algorithm CSA was improved and the weight feature of Crows was used for optimal clustering. In order to balance between diversification and intensification, the parameter *AP* was dynamically adjusted and normal and abnormal samples were clustered into two and three clusters respectively. Moreover, the cluster number was labeled as the data class. In the last phase, data modeling was performed using the Stacking hybrid technique, which is a two-levels technique. In the level-0, simple classifiers such as LR, SVM and KNN classifiers were used and in the level-1, the classifier Meta Naive Bayes was used. The SEAI, LCL and Raw datasets were used to experiments. The results of each phase revealed that the proposed framework had high performance. To evaluate the efficiency of the proposed framework, the Accuracy, Precision, Recall, F-measure, DR, FPR and ROC criteria were calculated. The results of the proposed framework were compared with the literature research and it was found that the results of the proposed framework were better. Therefore, with the evaluation results, we showed that in actual power grids, if the data is collected correctly and sufficiently, this framework can effectively detect abnormal attacks. Also, for new attacks, if their patterns can be detected from the data, it is easily possible to implement these types of attacks. Due to the fact that real time detection of electricity theft requires special data and relevant datasets, and the scenario of its implementation has its own literature, the scenario of real time theft detection is proposed as the future work.

REFERENCES

[1] S. Salkuti, "Challenges, issues and opportunities for the development of smart grid," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, pp. 1179-1186, 2020, doi: 10.11591/ijece.v10i2.pp1179-1186.

[2] G. Dranka and P. Ferreira, "Towards a smart grid power system in Brazil: Challenges and opportunities," *Energy Policy*, vol. 136, 2019, Art. no. 111033, doi: https://doi.org/10.1016/j.enpol.2019.111033

[3] R. Kappagantu and S. A. Daniel, "Challenges and issues of smart grid implementation: A case of Indian scenario," *Journal of Electrical Systems and Information Technology*, vol. 5, no. 3, pp. 453-467, 2018, doi: 10.1016/j.jesit.2018.01.002.

[4] Z. Feng, J. Huang, W. H. Tang, and M. Shahidehpour, "Data mining for abnormal power consumption pattern detection based on local matrix reconstruction," *International Journal of Electrical Power and Energy Systems*, vol. 123, 2020, Art. no. 106315, doi: https://doi.org/10.1016/j.ijepes.2020.106315.
[5] K. Blazakis and G. Stavrakakis, "Efficient Power Theft Detection for Residential Consumers Using Mean Shift Data Mining Knowledge Discovery Process," *International Journal of Artificial Intelligence and Applications*, vol. 10, 2019, doi: 10.5121/ijaia.2019.10106.
[6] K. Zheng, Y. Wang, Q. Chen and Y. Li, "Electricity theft detecting based on density-clustering method," *2017 IEEE Innovative Smart Grid Technologies-Asia (ISGT-Asia)*, Auckland, New Zealand, 2017, pp. 1-6, doi: 10.1109/ISGT-Asia.2017.8378347.
[7] C. Park and T. Kim, "Energy Theft Detection in Advanced Metering Infrastructure Based on Anomaly Pattern Detection," *Energies*, vol. 13, 2020, Art. no. 3832, doi: 10.3390/en13153832.
[8] S. Singh, R. Bose, and A. Joshi, "Energy Theft Detection for AMI usingPrincipal Component Analysis basedReconstructed Data," *IET Cyber-Physical Systems: Theory and Applications*, vol. 4, no. 2, pp. 1-9, 2018, doi: 10.1049/iet-cps.2018.5050
[9] F. Xiao and Q. Ai, "Electricity Theft Detection in Smart Grid Using Random Matrix Theory," *IET Generation, Transmission and Distribution*, vol. 12, no. 2, pp. 371-378, 2017, doi: 10.1049/iet-gtd.2017.0898.
[10] P. Jokar, N. Arianpoo, and V. C. M. Leung, "Electricity Theft Detection in AMI Using Customers' Consumption Patterns," in *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 216-226, Jan. 2016, doi: 10.1109/TSG.2015.2425222.
[11] Hasan, M., Toma, R. N., Nahid, A. A., Islam, M. M., and Kim, J. M., "Electricity Theft Detection in Smart Grid Systems: A CNN-LSTM Based Approach," *Energies*, vol. 12, no. 17, 2019, Art. no. 3310, doi: 10.3390/en12173310.
[12] S. Li, Y. Han, X. Yao, S. Yingchen, J. Wang, and Q. Zhao, "Electricity Theft Detection in Power Grids with Deep Learning and Random Forests," *Journal of Electrical and Computer Engineering*, vol. 2019, pp. 1-12, 2019, doi: doi: 10.1155/2019/4136874.
[13] Z. Zheng, Y. Yang, X. Niu, H. Dai, and Y. Zhou, "Wide and Deep Convolutional Neural Networks for Electricity-Theft Detection to Secure Smart Grids," in *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1606-1615, April 2018, doi: 10.1109/TII.2017.2785963.
[14] S. C. Yip, K. Wong, W. P. Hew, M. T. Gan, R. C. W. Phan, and S. W. Tan, "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, doi: 10.1016/j.ijepes.2017.04.005
[15] R. Razavi, A. Gharipour, M. Fleury, and I. J. Akpan, "A practical feature-engineering framework for electricity theft detection in smart grids," *Applied Energy*, vol. 238, pp. 481-494, 2019, doi: 10.1016/j.apenergy.2019.01.076.
[16] W. Zhang, X. Dong, H. Li, J. Xu, and D. Wang, "Unsupervised Detection of Abnormal Electricity Consumption Behavior Based on Feature Engineering," in *IEEE Access*, vol. 8, pp. 55483-55500, 2020, doi: 10.1109/ACCESS.2020.2980079.
[17] J. I. Guerrero, I. Monedero, F. Biscarri, J. Biscarri, R. Millán, and C. León, "Non-Technical Losses Reduction by Improving the Inspections Accuracy in a Power Utility," in *IEEE Transactions on Power Systems*, vol. 33, no. 2, pp. 1209-1218, March 2018, doi: 10.1109/TPWRS.2017.2721435.
[18] A. A. Ghasemi and M. Gitizadeh, "Detection of illegal consumers using pattern classification approach combined with Levenberg-Marquardt method in smart grid," *International Journal of Electrical Power and Energy Systems*, vol. 99, pp. 363-375, 2018, doi: 10.1016/j.ijepes.2018.01.036.
[19] M. M. Buzau, J. Tejedor-Aguilera, P. Cruz-Romero and A. Gómez-Expósito, "Detection of Non-Technical Losses Using Smart Meter Data and Supervised Learning," in *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 2661-2670, May 2019, doi: 10.1109/TSG.2018.2807925.
[20] L. A. P. Júnior *et al*., "Unsupervised non-technical losses identification through optimum-path forest," *Electric Power Systems Research*, vol. 140, pp. 413-423, 2016, doi: 10.1016/j.epsr.2016.05.036.
[21] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 1995, doi: 10.1109/MHS.1995.494215
[22] N. Clayton and N. Emery, "Corvid cognition," *Current biology*, vol. 15, no. 3, pp. R80-81, Feb. 2005, doi: 10.1016/j.cub.2005.01.020.
[23] H. Prior, A. Schwarz, and O. G¨unt¨urk¨un, "Mirror-induced behavior in the magpie (Pica pica): evidence of self-recognition," *PLoS Biology*, vol. 6, Aug. 2008, Art. no. e202, doi: 10.1371/journal.pbio.0060202.
[24] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm," *Computers and Structures*, vol. 169, pp. 1-12, 2016, doi: 10.1016/j.compstruc.2016.03.001.
[25] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, pp. 459-471, 2007, doi: 10.1007/s10898-007-9149-x.
[26] J. Wang, "Max-Min Distance Nonnegative Matrix Factorization," *Neural Networks*, vol. 61, 2013.
[27] D. Davies and D. Bouldin, "A Cluster Separation Measure," Pattern Analysis and Machine Intelligence, *IEEE Transactions on, vol. PAMI-1*, pp. 224-227, 1979, doi: 10.1016/j.neunet.2014.10.006.
[28] D. H. Wolpert, "Stacked Generalization," *textslNeural Networks*, vol. 5, no. 2, pp. 241-259, 1992, doi: 10.1016/S0893-6080(05)80023-1.

[29] Commission for Energy Regulation (CER), "CER Smart Metering Project-Electricity Customer Behaviour Trial, 2009-2010 [dataset]," *1st Edition. Irish Social Science*, Data Archive. [Online]. Available: https://www.ucd.ie/issda/data/commissionforenergyregulationcer/, [Accessed 18 May 2020].
[30] Raw user dataset-state grid corporation of china (sgcc):Sep 22, 2018. [Online]. Available: https://github.com/henryRDlab/ElectricityTheftDetection, [Accessed 18 June 2020].
[31] Low-Carbon-London, [Online]. Available: https://data.london.gov.uk/dataset/smartmeter-energy-use-data-in-london-households?, [Accessed 4 June 2020].
[32] M. Nabil, M. Ismail, M. M. E. A. Mahmoud, W. Alasmary, and E. Serpedin, "PPETD: Privacy-Preserving Electricity Theft Detection Scheme With Load Monitoring and Billing for AMI Networks," in *IEEE Access*, vol. 7, pp. 96334-96348, 2019, doi: 10.1109/ACCESS.2019.2925322.
[33] M. I. Ibrahem, M. Nabil, M. M. Fouda, M. M. E. A. Mahmoud, W. Alasmary, and F. Alsolami, "Efficient Privacy-Preserving Electricity Theft Detection With Dynamic Billing and Load Monitoring for AMI Networks," in *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 1243-1258, 15 Jan, 2021, doi: 10.1109/JIOT.2020.3026692.

BIOGRAPHIES OF AUTHORS

Hassan Ghaedi received the M.S. degrees in software engineering from Islamic Azad University, Arak branch, Iran. He is currently pursuing the Ph.D. degree in the Department of Computer Engineering, islamic azad University, Neyshabur branch, iran. His research interests include data mining (classification and clustering). Email: hassan.ghaedi60@gmail.com

Seyed Reza Kamel Tabbakh is with the Department of Software Engineering, Faculty of Engineering, Islamic Azad University - Mashhad branch, Mashhad, Iran. He received his PhD in communication and network engineering from University Putra Malaysia (UPM) in 2011. He received his BSc and MSc in software engineering from Islamic Azad University, Mashhad branch and Islamic Azad University, South Tehran branch, Iran respectively. His research interests include IPv6 networks, routing and security.

Reza Ghaemi received his Ph.D. degree in Computer Engineering and Artificial Intelligence from National University Putra Malaysia in 2011. He is currently an Associate Professor at Islamic Azad University, Quchan branch. His research interests include machine learning, data mining and soft computing. Email: rezaghaemi73@gmail.com