Journal of ELECTRICAL ENGINEERING, VOL 72(2021), NO5, 315–322

sciendo

PAPERS

Fault detection in power grids based on improved supervised machine learning binary classification

Mohammed Wadi[1]

With the increased complexity of power systems and the high integration of smart meters, advanced sensors, and high-level communication infrastructures within the modern power grids, the collected data becomes enormous and requires fast computation and outstanding analyzing methods under normal conditions. However, under abnormal conditions such as faults, the challenges dramatically increase. Such faults require timely and accurate fault detection, identification, and location approaches for guaranteeing their desired performance. This paper proposes two machine learning approaches based on the binary classification to improve the process of fault detection in smart grids. Besides, it presents four machine learning models trained and tested on real and modern fault detection data set designed by the Technical University of Ostrava. Many evaluation measures are applied to test and compare these approaches and models. Moreover, receiver operating characteristic curves are utilized to prove the applicability and validity of the proposed approaches. Finally, the proposed models are compared to previous studies to confirm their superiority.

K e y w o r d s: fault detection, smart grids, machine learning, binary classification

1 Introduction

Power grids are one of the most critical infrastructures in any country. Providing power electricity at acceptable levels of stability, reliability, and quality is one of the top tasks of power system operators. Despite all efforts undertaken to prevent fault or malfunction in the electricity networks, these faults are inevitable. To cope with these faults, it is essential to identify, classify, and locate the fault accurately. Voltage and current are the main signals in power systems to determine the faults [1]. However, there is no clear boundary between faulty and unfaulty signals. Therefore, it is highly complicated to reveal the deep faulty information embedded into these signals. Consequently, feature extraction methods are crucial to separate these signals and to accurately diagnosis fault signals [2]. Many approaches were proposed to handle feature extraction problem such as transformation [3, 4], dimensionality reduction [5], and modal transformation (Clarke transformation) [6, 7].

Yu and Gu [3], proposed an algorithm that combines analog low pass filter and modified full-cycle discrete Fourier transform (DFT) to eliminate the decaying direct current (DC) in a voltage or current signal and to improve the speed of decomposes the accurate fundamental frequency components. An approach based on artificial neural network (ANN) and support vector machine (SVM) relies on the gathered measurements at the substation, circuit breaker, and relay states for locating faults in radial distribution systems was presented in [5]. Many methods of fault detection and classification were utilized neural network [7, 8]. Zin *et al* [7] proposed an algorithm that combines back-propagation neural network (BPNN) based on modal Clarke transformation and discrete wavelet transform (DWT) for fault detection in transmission lines. A hybrid classification approach based on DWT and long short-term memory network (LSTMN) for detecting insulated overhead conductors fault according to partial discharge [9]. In addition, synthetic minority oversampling technique (SMOTE) is used to balance imbalanced data; however, hyperparameter tuning approaches were not used.

Many methods of fault detection and classification were appeared in the literature using fuzzy [10], neuro-fuzzy [11], adaptive neuro-fuzzy [12], and wavelet adaptive neuro-fuzzy [13]. In [12], an algorithm that integrates wavelet into adaptive neuro-fuzzy to identify the faults on series compensated transmission lines was introduced.

Feature extraction plays an essential role in fault detection. Fault detection is before fault classification and fault location stages. When the classifier is able to discriminate between faulty and unfaulty states, there is no need to implement additional fault detection techniques [2]. In this case, two methods can classify fault detection. One method uses classification-based technique to differentiate faulty and unfaulty states. The other method uses anomaly-based technique by adding the unfaulty states to the output category, and a fault is detected whenever the output is other than the unfaulty state [14]. With the advancement of complicated smart grids and due to the drawbacks of the classical fault detection methods [15]. Machine learning-based fault detection methods providing better performance and are less likely to be influenced by parameters of lines or fault parameters. The major

[1] Electrical-Electronics Engineering Depatrment, Istanbul Sabahattin Zaim University, Istanbul, Turkey, mohammed.wadi@izu.edu.tr

https://doi.org/10.2478/jee-2021-0044, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X

contributions and motivations of this paper can be summarized as follows:

- Fault occurrence in large-scale power grids causes large parts of a power outage that affects both utilities and customers; therefore, accurate and rapid fault detection algorithms are indispensable.
- Most of the proposed detection methods are based on simulation results that cannot reflect the actual situations in real systems. Therefore, this paper presents a modified supervised machine learning method based on real collected data that accurately shows the weakness and strengths of the used machine learning techniques.
- Complex modern and smart grids need very fast measuring meters and sensors to provide rapid detection and classification of different faults that can be efficiently achieved by machine learning methods, opposite to traditional detection methods that may fail.
- To improve the fault detection mechanism, two different models are proposed.
- To verify the proposed models, four binary classification models are trained and tested over a real data set collected from a real-time electrical grid.

2 Fault detection system description

A simplified model for the fault detection mechanism is depicted in Fig. 1. The mechanism can be summarized into the following main steps: firstly, the sampled current and voltage signals are passed to the feature extraction module. The main task of this module to extract features then passed them to the fault detector. Next, the fault detector identifies whether there is a fault or not. Then, the decision of the fault detector along with the extracted features is simultaneously passed to the fault classifier and the fault locator. Finally, the fault classifier identifies the fault type, whereas the fault location specifies the fault location.

3 Power line fault detection data set

The utilized data set in this study was designed at Energy and Environmental Technology Center (ENET) at Technical University of Ostrava (VSB) [16]. The data were gathered with advanced meters designed at the ENET Center. Besides, this data set was firstly published on the Kaggle website in 2018 within the competition to identify the partial discharge patterns in signals collected from these power lines. The data set consists of 8712 signal samples, with each signal includes 800-thousand measurements of the line voltage. Each signal was registered everyone cycle (20 ms) for a 50 Hz power system. The measurements were recorded simultaneously at all phases and available as parquet Apache files. The raw data were rearranged into CSV Excel files; each signal has its target class 0 or 1 label. 1 indicates to faulty signal, while 0 indicates to unfaulty signal. In addition, the VSB or ENET data set contains 94% of unfaulty singles and 6% faulty singles. This big unbalancing leads to bad classification since the classifiers will bias to the class of the majority signals. Balancing unbalanced data is an inevitable task in order to improve the classification process. Many resampling techniques for balancing unbalanced data have appeared in the literature [17, 18, 19]. Balancing techniques divide into three main techniques over sampling, under sampling, and ensemble [20, 21]. Oversampling is used to generate extra data from the minority class to overcome its shortage of data [22]. There are many advantages of over sampling technique that does not lead to information loss and provides sufficient accuracy. The main drawback that it increases the data size, and it may lead to over fitting [23]. On the contrary, under sampling is used to reduce data from the majority class to match up the data of the minority class [17]. There are many advantages of the under sampling technique that reduces the storage problem and provides, in most cases, a balanced data set that carries the intelligent information. The main drawbacks originate from the loss of valuable information of the majority class that may lead to imperfect classifications [24]. However, ensemble sampling is a technique that generates many models then merges them into one model to provide refined results. One of the most common used over sampling techniques is SMOTE [25]. SMOTE consists of four steps: firstly, over sample the minority class based on synthetic-over sampled rather than by over sampling with replacement. Secondly, SMOTE detects the samples from the minority class which are the k nearest neighbours of the selected sample. Then, SMOTE randomly decides the sample that belongs to one of these k nearest neighbours. Finally, SMOTE generates a synthetic instance at a random point that combines the selected sample and its selected neighbour [25].

4 Proposed models of classification

Two machine learning classification models are proposed to improve the process of fault detection in power grids. The first proposed model is based on the concept of balancing imbalanced datasets into a balanced data set. The second proposed model is based on balancing imbalanced datasets into balanced datasets then applying the hyperparameter tuning approaches. Figure 2 and Fig. 3 illustrate the block diagrams of the proposed two machine learning models. The first machine learning classification model consists of four blocks: data preprocessing, pretraining, training, and test. The imbalanced data set is converted into a balanced data set using the SMOTE approach within the data preprocessing block, as illustrated in section 3. Then, the balanced data set is passed to the pre-training block. The processed data set is passed to the next block, whereas the machine learning training models such as two-class Bayes point machine (TCBPM), two-class averaged perceptron (TCAP), Two class boosted decision tree (TCBDT), and two-class decision Forest (TCDF) are initialized to be applied on the untrained

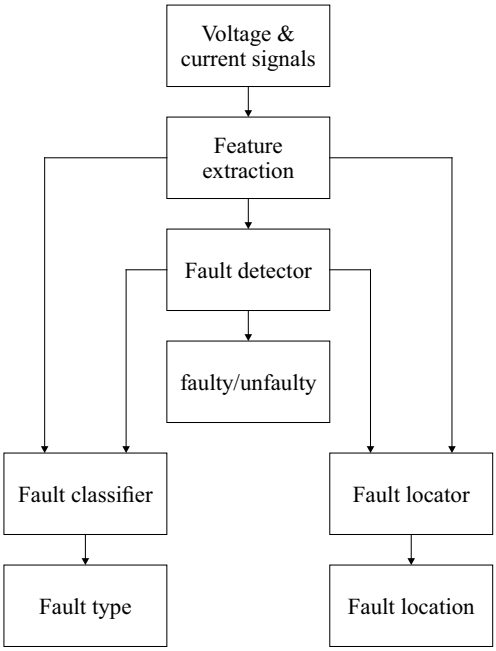

Fig. 1. A simplified model for fault detection mechanism

model. Within the training block, the processed data set is split into two subsets, training, and test; one of the four training mentioned above machine learning training models is applied to the training subset. Finally, the trained model is tested via test subset, and the evaluation scores are summarized as output.

Similarly, the second machine learning proposed classification model consists of four blocks; all blocks are like the first model blocks except the pertaining block. The pre-training block is modified by adding two sub-blocks, the optimal features the number of incorrect predictions of the positive class, and F_{N} is the number of incorrect predictions of the negative class.

5 Correctness of classification

Many statistical measures can be applied to compare and test the performance of the proposed machine learning calcification models [12]. In this study, seven measures based on True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) were utilized. TP is the number of correct predictions of the positive class (the unfaulty signal), TN is the number of correct predictions of the negative class (the faulty signal), FP is the number of incorrect predictions of the positive class, and FN is the number of incorrect predictions of the negative class. The formulas of measures are briefly defined as follows:

Here: T_{P} - are correct predictions, T_{N} - are the correct predictions of negative class ... and F_{P} - are the incorrect predictions of the positive class ...

- Accuracy is the ratio of the sum of correct predictions to the total samples of data set.

$$A = \frac{T_{\mathrm{P}} + T_{\mathrm{N}}}{T_{\mathrm{P}} + T_{\mathrm{N}} + F_{\mathrm{P}} + F_{\mathrm{N}}}. \tag{1}$$

- Precision is the ratio of correct predictions of the positive class to the total positive predictions.

$$P = \frac{T_{\mathrm{P}}}{T_{\mathrm{P}} + F_{\mathrm{P}}}. \tag{2}$$

- Recall (also called Sensitivity), is the ratio of correct predictions of the positive class to the sum of correct predictions of the positive class and incorrect predictions of the negative class.

$$R = \frac{T_{\mathrm{P}}}{T_{\mathrm{P}} + F_{\mathrm{N}}}. \tag{3}$$

- Score is a weighted average of Precision and Recall. Score is usually more valuable than accuracy, particularly if the underlying data set has an uneven class distribution.

$$S = \frac{2PR}{P + R}. \tag{4}$$

- Specificity, (also called true negative rate – R_{TN}), is the ratio of correct predictions of the negative class

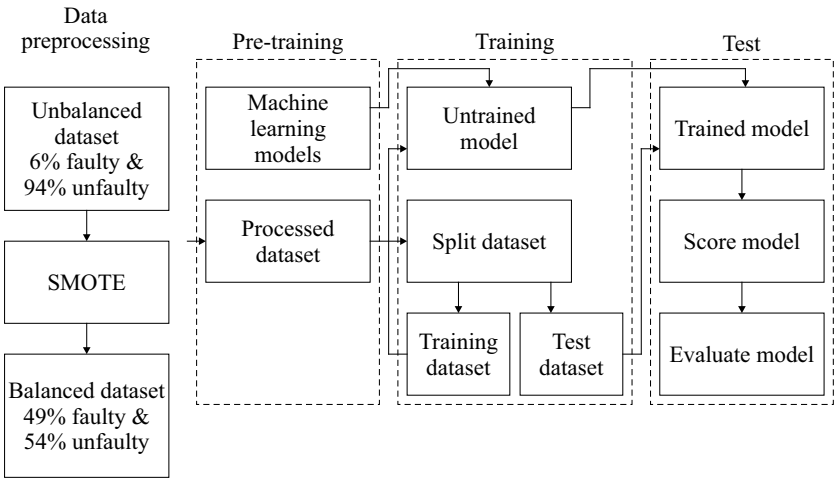

Fig. 2. The first machine learning proposed model

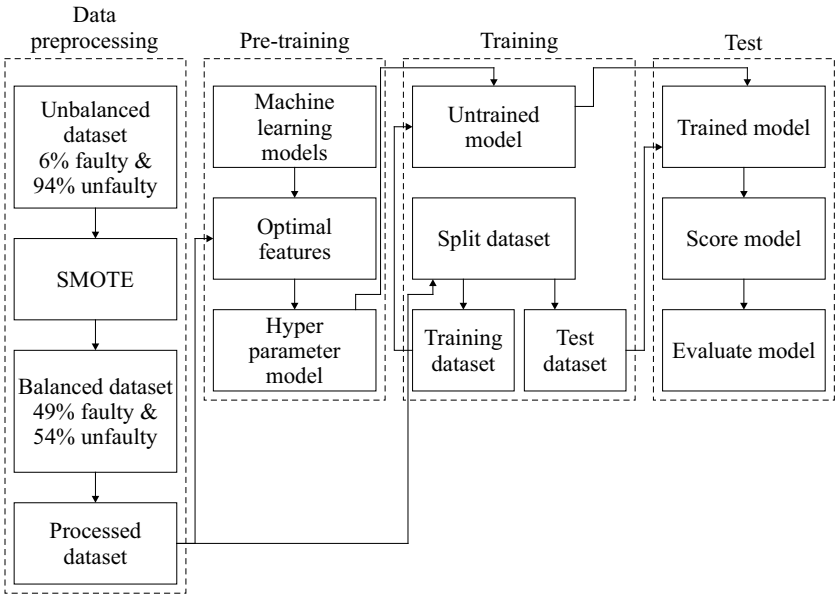

Fig. 3. The second machine learning proposed model

to the sum of correct predictions of the negative class and incorrect predictions of the positive class.

$$R_{\mathrm{TN}} = \frac{T_{\mathrm{N}}}{T_{\mathrm{N}} + F_{\mathrm{P}}}. \tag{5}$$

- False alarm rate (R_{FA}), also known as False Positive Rate (FPR), is the ratio of the incorrect predictions of the positive class to the sum of incorrect predictions of the positive class and correct predictions of the negative class.

$$R_{\mathrm{FA}} = \frac{F_{\mathrm{P}}}{F_{\mathrm{P}} + T_{\mathrm{N}}}. \tag{6}$$

- False negative rate (R_{FN}) is the ratio of the incorrect predictions of the negative class to the summation of incorrect predictions of the negative class and correct predictions of the positive class.

$$R_{\mathrm{FN}} = \frac{F_{\mathrm{N}}}{F_{\mathrm{N}} + T_{\mathrm{P}}}. \tag{7}$$

The confusion matrix summarizes the Accuracy, Specificity, Recall, and Precision and is used to measure the performance of a machine learning classification problem. Besides, it is an efficient tool for drawing receiver operating characteristic (ROC) curves and calculating area under the ROC Curve (AUC). The ROC is a graphical plot that demonstrates the characteristic capability of a binary classifier. While AUC shows how much the model can distinguish between classes. As AUC approaches 1, the classifier efficiently can distinguish unfaulty from faulty cases, whereas when it approaches 0.5, it works with 50% efficiency. Finally, approaches 0 completely can detect neither faulty nor unfaulty. AUC can be calculated using True Positive Rate (RTP) or Recall and RTN or Specificity as follows:

True poositive rate (R_{TP}) or Recall and TNR or Specificity as

$$AUC = \frac{R_{\mathrm{TP}} + R_{\mathrm{TN}}}{2}. \tag{8}$$

6 Results and discussion

The raw fault detection data were in parquet Apache files and rearranged into CSV Excel files using Python 3.9 and research co-laboratory platform developed by Google [28]. The proposed two machine learning models are developed and implemented by Azure Machine Learning Studio (AMLS) designer [29]. AMLS is a cloud-based platform that provides an efficient tool for developing, analyzing, and testing different machine learning models. This section can be divided into three scenarios as follows.

6.1 Scenario 1: Balanced data without hyper-parameter tuning at 70% training

Balancing data plays a crucial role in enhancing the performance of the classifier since imbalanced data cause classifier biasing to the majority class of the data. SMOTE is one of the most generally applied over sampling methods to balance the imbalanced data sets by randomly increasing minority samples by replicating them. Three essential parameters to set the SMOTE technique, its percentage, number of nearest neighbors, and random seeds. In this paper, SMOTE percentage value was 1400, the number of nearest neighbors was 10, whereas the number of random seeds was 1000. The SMOTE converts the imbalanced data set (6% faulty and 94% unfaulty) into a balanced data set (49% faulty and 51% unfaulty). The first proposed machine learning model given in Fig. 2 based on the four machine learning training models without hyper-parameter tuning illustrated in Section 4 was applied to classify the fault in the VSB data set. Table 1 depicts the experimental results under Scenario 1. In this study, three normalization methods were used Zscore, MinMax, and Loglogistic. It is essential to explain the input parameters of each training machine model. For each method, the first inputs are the parameters of the machine learning model, while the last input is to

Table 1. Results of the first proposed machine learning model under Scenario 1; $\alpha = 0.2$, $r = 1000$

Learning model	N		T_{P}	T_{N}	F_{P}	F_{N}	A (%)	P (%)	R (%)	S (%)	R_{TP} (%)	R_{FA} (%)	R_{TN} (%)	R_{FN} (%)
TCBPM	30	Zscore	1757	1960	496	605	77.1	78.0	74.4	7762	84.0	20.2	79.8	25.6
	50		1751	1953	503	611	76.9	77.7	74.1	77.3	84.0	20.5	79.5	25.9
	100		1749	1953	503	613	76.8	77.7	74.0	77.2	84.1	20.5	79.5	26.0
	30	MinMax	1645	1749	707	717	70.4	69.9	69.6	70.2	75.7	28.8	71.2	30.4
	50		1662	1766	690	700	71.1	70.7	70.4	7.09	76.7	28.1	71.5	29.6
	100		1687	1817	639	675	72.7	72.5	71.4	72.6	78.5	26.0	74.0	28.6
	30	Loglogistic	1454	1448	1008	908	60.2	59.1	61.6	59.6	62.6	41.0	59.0	38.4
	50		1390	1571	972	885	61.5	58.8	61.1	60.1	63.0	38.2	61.8	38.9
	100		1333	1624	832	1029	61.4	61.6	56.4	61.5	63.2	33.9	66.1	43.6
TCAP	10	Zscore	1371	1534	922	991	60.3	59.8	58.0	60.0	63.5	37.5	62.5	42.0
	100		1429	1441	1015	933	59.6	58.5	60.5	59.0	64.2	41.3	58.7	39.5
	10	MinMax	1749	1901	555	613	75.8	75.9	74.0	75.8	82.5	22.6	77.4	26.0
	100		1756	1987	469	606	77.7	78.9	74.3	78.3	84.7	19.1	80.9	25.7
	10	Loglogistic	1371	1534	922	991	60.3	59.8	58.0	60.0	63.5	37.5	62.5	42.0
	100		1429	1441	1015	933	59.6	58.5	60.5	59.0	64.2	41.3	58.7	39.5

Learning model	$N_1 \mid T$ ($N_2 = 10$)		T_{P}	T_{N}	F_{P}	F_{N}	A (%)	P (%)	R (%)	S (%)	R_{TP} (%)	R_{FA} (%)	R_{TN} (%)	R_{FN} (%)
TCBDT	20 | 100	Zscore	2279	2367	89	83	96.4	96.2	96.5	96.3	99.4	3.6	96.4	3.5
	50 | 200		2281	2389	67	81	96.9	97.1	96.6	97.0	99.3	2.7	97.3	3.4
	20 | 100	MinMax	2285	2359	97	77	96.4	95.9	96.7	96.2	99.4	3.9	96.1	3.3
	50 | 200		2275	2387	69	87	96.8	97.1	96.3	96.9	99.4	2.8	97.2	3.7
	20 | 100	Loglogistic	2270	2354	102	92	96.0	95.7	96.1	95.8	99.3	4.2	95.8	3.9
	50 | 200		2276	2371	85	86	96.5	96.4	96.4	96.4	99.1	3.5	96.5	3.6

$N_3 = 8$, $N_4 = 128$, $N_5 = 1$, $D = 32$

Learning model		T_{P}	T_{N}	F_{P}	F_{N}	A (%)	P (%)	R (%)	S (%)	R_{TP} (%)	R_{FA} (%)	R_{TN} (%)	R_{FN} (%)
TCDF (bagging)	Zscore & MinMax	2214	2366	69	87	96.8	97.1	96.3	96.9	99.4	2.8	97.2	3.7
	Loglogistic	2220	2365	91	142	95.2	96.1	94.0	95.6	98.5	3.7	96.3	6.0

illustrate the utilized normalization method, for example, TCBPM (N, Zscore) means that the first input of TCBPM is the number of iterations, and the second input is the normalization method. The input parameters of the used machine learning training models are explained as follows: N – is the maximum number of iteration, α – is the learning rate, r – is the number of seeds, N_1 is the maximum number of leaves/tree, N_2 is the minimum number of training instances to form a leaf, T – is the number of constructed trees, N_3 – is the number of decision trees, D – is the maximum depth of the decision trees, N_4 - is the number of random splits per node, and N_5 – is the minimum number of samples per leaf node.

The model with the best performance accomplishes high $A, P, R, S, AUC, R_{\mathrm{TP}}, R_{\mathrm{TN}}$, and low R_{FA} and R_{FN}. The boxed values demonstrate the training models that achieved the best performance. Examining Tab. 1, it can be noticed that TCBPM based on Zscore normalization outperformed TCBPM based on MinMax and Loglogistic. For instance, comparing TCBPM ($N = 30$, Zscore) and TCBPM ($N = 30$, MinMax), it can be observed that all evaluation measures are improved by about 10% only by changing the normalization method. Also, TCBDT achieved the best performance overall in other machine models. Notably, at TCBDT ($N_1 = 50, N = 10, \alpha = 0.2, T = 200, r = 1000$), the increasing number of leaves per tree and the number of constructed trees significantly improve the classifier performance [30]. TCDF came in the second rank after TCBDT in terms of performance. The balancing data set offers significant improvements in all evaluation measures. For example, AUC, R_{FA}, and R_{FN} were improved from 0.553 to 0.994, 0.90 to 0.028, and 0.88 to 0.033, respectively. Accuracy, Precision, Recall, and Score were also considerably enhanced.

6.2 Scenario 2: Balanced data with hyperparameter tuning at 70% training

As mentioned in Section 4, optimal features selection and hyperparameter tuning subblocks play a critical role in improving the performance of the classifier. There are three built-in models in AMLS for hyperparameter tuning

True class	The worst model	0	1333	832	61.5%	38.5%
		1	1029	1624	38.8%	61.2%
	The best model	0	2281	67	97.2%	2.8%
		1	81	2389	96.7%	3.3%
			0	1	TPR	FNR
			Predicted class			

Fig. 4. Confusion matrix of scenarios 1 and 2

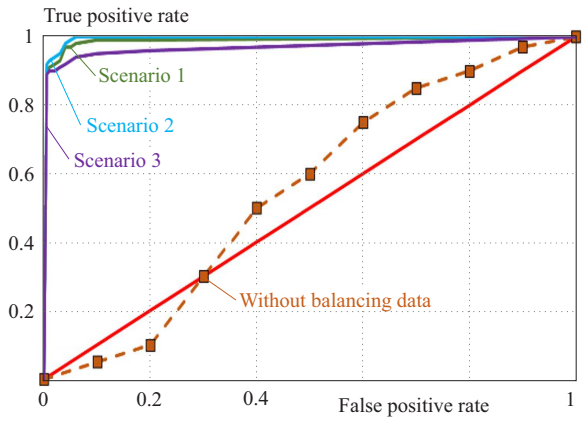

Fig. 5. ROC curves of the three scenarios

based on sweeping parameter mode: entire grid, random sweep, and random grid [29].

In this Scenario, the processed data are passed to optimal features selection subblock to select the most impactful ten features from 19 whole features. Afterward, the parameters of the machine learning model based on these chosen features are tuned. The random sweep mode with five runs on a random sweep was applied to the four machine learning training models to adjust their parameters. In this Scenario, the hyperparameter tuning slightly improved the performance of the classifier, and the results, to a large extent, are similar to the results in Tab. 1. Figure 4 presents the confusion matrix of two scenarios.

6.3 Scenario 3: Balanced data with hyperparameter tuning at 60% training

The percentages of data splitting into training and test are still a controversial problem. According to the Pareto principle, it is advisable to follow the ratio of 80:20 [21]. Other studies prefer the ratio of 70:30 for small data and 90:10 for massive data [22]. In this scenario, the balanced dataset was split into 60% training and 40% test to verify the robustness of the classifier. The experimental results are given in Tab. 2. Comparing Tab. 1 and Tab. 2, it can be noticed that are slight differences between the results. Based on the 60:40 ratio, the evaluation measures slightly deteriorate compared to the 70:30 ratio, and these results confirm that a high training ratio leads to better performance. The obtained results verify the robustness of the proposed models. In addition to the utilized evaluation measures described in Section 5, the ROC curve can be used to rank the best classifier [3]. ROC curve is a plot that explains the performance of a classifier by comparing it with the diagonal line of the ROC curve that divides the plot into two identical triangles, up and down. As the classifier curve approaches the upper left corner of the ROC curve as the performance becomes enhanced. Fig. 5 depicts the ROC curves of the proposed machine learning models compared to processing data without balancing. The AUC equals 0.695 without balancing data, whereas it equals 0.994 when balancing data based on TCBDT, which means nearly 30% of improvements. Table 2 summarizes the results of Scenario 3. Again, TCBDT outperformed all other training models. It is evident from Fig. 5 that Scenario 2 achieved the best performance, while scenarios 1 and 3 occupied the second and third rank, respectively.

To fairly judge the proposed machine learning models in this paper, they compared with the study [31, 9, 8] since they utilized the same dataset. Table 3 illustrates these comparisons. The proposed two models based on three different scenarios outperformed the results of these studies in terms of all evaluation metrics. For instance, the precision, Recall, and Score are improved by 26%, 25%, and 25.5%, respectively, when compared to [31]. These comparisons validate the effectiveness of the suggested machine learning models in the electrical fault detection problem.

7 Conclusions

With the increased complexity of power systems, fault detection in such systems becomes a challenging mission. Therefore, this field of research is still a rich and active area. This paper presents an automated fault detection approach to cope with deficiencies of the classical fault detection approaches. It proposes two machine learning models based on the binary classification to improve fault detection in smart grids. The improvements were made over two stages, dataset balancing and hyperparameter tuning. A real-time fault detection dataset designed by the Technical University of Ostrava was utilized to conduct this study. The raw data were 94% unfaulty to 6% faulty. These data were balanced using SMOTE technique to be 51% unfaulty to 49% faulty. The results of the balanced dataset were compared to the imbalanced dataset, and there were significant enhancements in all evaluation metrics; for instance, AUC, R_{FA}, and R_{FN}, were improved by 44%, 96%, and 95%, respectively. Hyper parameter tuning is the process of tuning the machine learning model parameters during the training process to improve testing results for generalizing them to be used in real-time applications. In this paper, the effect of hyper parameter tuning has slightly improved the performance of the classifier. The proposed two models were compared to previous research, and their superiority was verified. Finally, this study can be extended to many studies in this active research area, such as applying different machine learning models, balancing techniques, and classification approaches.

Table 2. Results of the second proposed machine learning model under Scenario 3; $\alpha = 0.2$, $r = 1000$

Learning model	N		T_P	T_N	F_P	F_N	A (%)	P (%)	R (%)	S (%)	R_{TP} (%)	R_{FA} (%)	R_{TN} (%)	R_{FN} (%)
TCBPM	30	Zscore	2341	2613	662	809	77.1	78.0	74.3	77.5	84.4	20.2	79.8	25.7
	100		2343	2612	663	807	77.1	77.9	74.4	77.5	84.5	20.2	79.8	25.6
	30	MinMax	2212	2351	924	938	71.0	70.5	70.2	70.8	75.7	28.2	71.8	29.8
	50		2222	2372	903	928	71.5	71.1	70.5	71.3	76.8	27.6	72.4	29.5
	100		2238	2436	839	912	72.7	72.7	71.0	72.7	78.6	25.6	74.4	29.0
	30	Loglogistic	1650	2300	975	1500	61.5	62.9	52.4	62.2	63.4	29.8	70.2	47.6
	50		1735	2232	1043	1415	61.7	62.5	55.1	62.1	63.5	31.8	68.2	44.9
	100		1770	2192	1083	1380	61.7	62.0	56.2	61.9	63.4	33.1	66.9	43.8
TCAP	10	Zscore	2330	2529	746	820	75.6	75.7	74.0	75.7	82.5	22.8	77.2	26.0
	100		2333	2708	567	817	78.5	80.4	74.1	79.4	85.0	17.3	82.7	29.5
	10	MinMax	2330	2529	746	820	75.6	75.7	74.0	75.7	82.5	22.8	77.2	26.0
	100		2333	2706	569	817	78.4	80.4	74.1	79.4	85.0	17.4	82.6	25.9
	10	Loglogistic	1601	2320	955	1549	61.0	62.6	50.8	61.8	63.4	29.2	70.8	49.2
	100		1581	2322	953	1569	60.7	62.4	50.2	61.6	63.5	29.1	70.9	49.8

Learning model	$N_1 \mid T$ ($N_2 = 10$)		T_P	T_N	F_P	F_N	A (%)	P (%)	R (%)	S (%)	R_{TP} (%)	R_{FA} (%)	R_{TN} (%)	R_{FN} (%)
TCBDT	20 | 100	Zscore	3030	3159	116	120	96.3	96.3	96.2	96.3	99.3	3.5	96.5	3.8
	50 | 200		3032	3181	94	118	96.7	97.0	96.3	96.8	99.4	2.9	97.1	3.7
	20 | 100	MinMax	3031	3156	119	119	96.3	96.2	96.2	96.3	99.3	3.6	96.4	3.8
	50 | 200		3031	3173	102	119	96.6	96.7	96.2	96.7	99.3	3.1	96.9	3.8
	20 | 100	Loglogistic	3027	3145	130	123	96.1	95.9	96.1	96.0	99.2	4.0	96.0	3.9
	50 | 200		3022	3183	92	128	96.6	97.0	95.9	96.8	99.2	2.8	97.2	4.1

$N_3 = 8$, $N_4 = 128$, $N_5 = 1$, $D = 32$

Learning model		T_P	T_N	F_P	F_N	A (%)	P (%)	R (%)	S (%)	R_{TP} (%)	R_{FA} (%)	R_{TN} (%)	R_{FN} (%)
TCDF (bagging)	Zscore & MinMax	2940	3157	118	210	94.9	96.1	93.3	95.5	98.5	3.6	96.4	6.7
	Loglogistic	2941	3179	96	209	95.3	96.8	93.4	96.0	98.6	2.9	97.1	6.6

Table 3. Comparison of previous research studies with this study

Study	Proposed technique		A	P	R	S	AUC
(31)	STL+SVM (Linear kernel)		-	0.72	0.70	0.70	-
	STL+SVM (6-Degree polynomial kernel)		-	0.67	0.55	0.45	-
	STL+SVM (RBF kernel)		-	0.75	0.73	0.72	-
	STL+SVM (Sigmoid kernel)		-	0.53	0.52	0.50	-
(9)	Non-PD Signals		-	0.81	0.81	0.81	-
	PD Signals		-	0.81	0.81	0.83	-
(8)	1-Phase optimized threshold		0.953	0.67	0.84	-	-
	3-Phase global threshold		0.967	0.73	0.96	-	-
This study		Scenario	A	P	R	S	AUC
TCBDT Zscore	$N_1 = 50$, $N_2 = 10$,	1	0.969	0.97	0.97	0.97	0.99
	$\alpha = 0.2$, $T = 200$,	2	0.969	0.97	0.97	0.97	0.99
	$r = 1000$,	3	0.967	0.97	0.96	0.97	0.99

References

[1] G. Fandi, F. O. Igbinovia, J. Tlusty, and R. Mahmoud, "Voltage regulation and power losses reduction in a wind farm integrated MV distribution network", *Journal of Electrical Engineering*, vol. 69, no. 1. 2018, doi: 10.1515/jee-2018-0012.

[2] A. A. Abo El-Ela, S. M. Allam, A. M. Shaheen, and N. A. Nagem, "Optimal allocation of biomass distributed generation in distribution systems using equilibrium algorithm", *Int. Trans. Electr. Energy Syst*, 2020, doi: 10.1002/2050-7038.12727.

[3] M. A. Abdelkader, Z. H. Osman, and M. A. Elshahed, "New analytical approach for simultaneous feeder reconfiguration and DG

hosting allocation in radial distribution networks", *Ain Shams Eng. J*, vol. 12, no. 2, 2021, doi: 10.1016/j.asej.2020.09.024.

[4] A. A. A. El-Ela, R. A. El-Sehiemy, A. M. Shaheen, and N. K. El-Ayaa, "Optimal allocation of DGs with network reconfiguration using improved spotted hyena algorithm", *WSEAS Trans. Power Syst*, vol. 15, pp. 60-67, Apr. 2020, doi: 10.37394/232016 .2020.15.7.

[5] A. Nuhanovi, J. Hivziefendi, and A. Hadimehmedovi, "Distribution network reconfiguration considering power losses and outages costs using genetic algorithm", *J. Electr. Eng*, vol. 64, no. 5, 2013, doi: 10.2478/jee-2013-0039.

[6] D. Šošić and P. Stefanov, "Multi-objective optimal reconfiguration of distribution network", *J. Electr. Eng*, vol. 69, no. 2, 2018, doi: 10.2478/jee-2018-0016.

[7] D. Šošić and P. Stefanov, "Reconfiguration of distribution system with distributed generation using an adaptive loop approach", *J. Electr. Eng*, vol. 70, no. 5, 2019, doi: 10.2478/jee-2019 -0066.

[8] A. M. Shaheen, A. M. Elsayed, and M. A. El Aziz, "Capacitor Switching with Distribution System Reconfiguration and Load Variations: Practical Case Study using ETAP and Network Analyzer," in" *2019 21st International Middle East Power Systems Conference, MEPCON 2019 - Proceedings*, 2019, doi: 10.1109/MEPCON47431.2019.9008159.

[9] A. Selim, S. Kamel, and F. Jurado, "Capacitors Allocation in Distribution Systems Using a Hybrid Formulation Based on Analytical and Two Metaheuristic Optimization Techniques", *Comput. Electr. Eng*, vol. 85, p. 106675, Jul. 2020, doi: 10.1016/ j.compeleceng.2020.106675.

[10] S. Sharma, S. Bhattacharjee, and A. Bhattacharya, "Quasi-Oppositional Swine Influenza Model Based Optimization with Quarantine for optimal allocation of DG in radial distribution network", *Int. J. Electr. Power Energy Syst*, vol. 74, pp. 348-373, Jan. 2016, doi: 10.1016/J.IJEPES.2015.07.034.

[11] X. Fu, H. Chen, R. Cai, and P. Yang, "Optimal allocation and adaptive VAR control of PV-DG in distribution networks", *Appl. Energy*, vol. 137, pp. 173-182, Jan. 2015, doi: 10.1016/J.APENERGY.2014.10.012.

[12] A. M. Shaheen, A. M. Elsayed, R. A. El-Sehiemy, and A. Y. Abdelaziz, "Equilibrium optimization algorithm for network reconfiguration and distributed generation allocation in power systems", *Appl. Soft Comput*, vol. 98, 2021, doi: 10.1016/j.asoc. 2020.106867.

[13] Abdullah M. Shaheen, R. A. El-Sehiemy, S. Kamel, E. E. Elattar, and A. M. Elsayed, "Improving Distribution Networks Consistency by Optimal Distribution System Reconfiguration and Distributed Generations", *IEEE Access*, vol. 9, pp. 67186-67200, 2021.

[14] A. M. Shaheen, A. M. Elsayed, R. A. El-Sehiemy, S. Kamel, and S. S. M. Ghoneim, "A modified marine predators optimization algorithm for simultaneous network reconfiguration and distributed generator allocation in distribution systems under different loading conditions", *Eng. Optim*, pp. 1-22, Apr. 2021, doi: 10.1080/0305215X.2021.1897799.

[15] M. Esmaeili, M. Sedighizadeh, and M. Esmaili, "Multi-objective optimal reconfiguration and DG (Distributed Generation) power allocation in distribution networks using Big Bang-Big Crunch algorithm considering load uncertainty", *Energy*, vol. 103, pp. 86-99, May 2016, doi: 10.1016/J.ENERGY.2016.02.152.

[16] H. B. Tolabi, A. L. Ara, and R. Hosseini, "A new thief and police algorithm and its application in simultaneous reconfiguration with optimal allocation of capacitor and distributed generation units", *Energy*, vol. 203, p. 117911, Jul. 2020, doi: 10.1016/J.ENERGY.2020.117911.

[17] T. Yuvaraj and K. Ravi, "Multi-objective simultaneous DG and DSTATCOM allocation in radial distribution networks using cuckoo searching algorithm", *Alexandria Eng. J*, vol. 57, no. 4, pp. 2729-2742, Dec. 2018, doi: 10.1016/J.AEJ.2018.01.001.

[18] M. G. Hemeida, A. A. Ibrahim, A. A. A. Mohamed, S. Alkhalaf, and A. M. B. El-Dine, "Optimal allocation of distributed generators DG based Manta Ray Foraging Optimization algorithm (MRFO)", *Ain Shams Eng. J*, vol. 12, no. 1, pp. 609-619, Mar. 2021, doi: 10.1016/J.ASEJ.2020.07.009.

[19] A. M. Shaheen, E. E. Elattar, R. A. El, S. Senior, and A. M. Elsayed, "An Improved Sunflower Optimization Algorithm Based-Monte Carlo Simulation for Efficiency Improvement of Radial Distribution Systems Considering Wind Power Uncertainty", *IEEE Access*, vol. 9, pp. 2332-2344, 2020, doi: 10.1109/ACCESS.2020.3047671.

[20] A. M. Shaheen, A. M. Elsayed, A. R. Ginidi, E. E. Elattar, and R. A. El-Sehiemy, "Effective Automation of Distribution Systems With Joint Integration of DGs/ SVCs Considering Reconfiguration Capability by Jellyfish Search Algorithm", *IEEE Access*, 2021, doi: 10.1109/ACCESS.2021.3092337.

[21] E. E. Elattar, A. M. Shaheen, A. M. El-Sayed, R. A. El-Sehiemy, and A. R. Ginidi, "Optimal Operation of Automated Distribution Networks Based-MRFO Algorithm", *IEEE Access*, 2021, doi: 10.1109/ACCESS.2021.3053479.

[22] W. Zhao, L. Wang, and Z. Zhang," *Artificial ecosystem-based optimization: a novel nature-inspired meta-heuristic algorithm*, vol. 32, no. 13. Springer London, 2020.

[23] R. S. Rao, K. Ravindra, K. Satish, and S. V. L. Narasimham, "Power loss minimization in distribution system using network reconfiguration in the presence of distributed generation", *IEEE Trans. Power Syst*, vol. 28, no. 1, pp. 317-325, 2013, doi: 10.1109/TPWRS.2012.2197227.

[24] A. M. Shaheen and R. A. El-Sehiemy, "Enhanced feeder reconfiguration in primary distribution networks using backtracking search technique", *Aust. J. Electr. Electron. Eng*, pp. 1-7, Sep. 2020, doi: 10.1080/1448837X.2020.1817231.

[25] O. A. Saleh, M. Elshahed, and M. Elsayed, "Enhancement of radial distribution network with distributed generation and system reconfiguration", *J. Electr. Syst*, vol. 14, no. 3, 2018.

[26] U. Raut and S. Mishra, "An improved sine-cosine algorithm for simultaneous network reconfiguration and DG allocation in power distribution systems", *Appl. Soft Comput. J*, vol. 92, p. 106293, Jul. 2020, doi: 10.1016/j.asoc.2020.106293.

[27] A. M. Shaheen, A. M. Elsayed, and R. A. El-sehiemy, "Optimal Economic-Environmental Operation for AC-MTDC Grids by Improved Crow Search Algorithm", *IEEE Syst. J*, 2021.

[28] W. Zhao, L. Wang, and Z. Zhang, "Supply-Demand-Based Optimization: A Novel Economics-Inspired Algorithm for Global Optimization", *IEEE Access*, vol. 7, pp. 73182-73206, 2019, doi: 10.1109/ACCESS.2019.2918753.

[29] A. M. Shaheen and R. A. El-Sehiemy, "Optimal Co-ordinated Allocation of Distributed Generation Units/ Capacitor Banks/ Voltage Regulators by EGWA", *IEEE Syst. J*, 2020, doi: 10.1109 /jsyst.2020.2986647.

Received 27 July 2021

Mohammed Wadi received the BSc and MSc in Electrical Engineering degrees from The Islamic University of Gaza, Palestine and the PhD degree from Yildiz Technical University, Istanbul, Turkey, in 2017. His main field of interest is in the reliability assessment of power systems. He worked as a lecturer at the University College of Science and Technology, Khanyounis, Palestine, as an electrical engineer at the Ministry of Higher Education, Khanyounis, Palestine and as a Director of Control & Operation Departement at Gaza 220/20 kV Power Plant, Palestine. He is now working as an assistant professor at the Department of Electrical & Electronics Engineering, Istanbul Sabahattin Zaim University, Istanbul in Turkey.