Electrical Power and Energy Systems 131 (2021) 107085

Contents lists available at ScienceDirect

International Journal of Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

Convolutional neural network applied to detect electricity theft: A comparative study on unbalanced data handling techniques

Jeanne Pereira *, Filipe Saraiva

Computer Science Postgraduate Program, Institute of Exact and Natural Sciences, Federal University of Pará, Belém, Pará, Brazil

ARTICLE INFO

Keywords:
Electricity theft
Convolutional neural network
Deep learning
Unbalanced data
2010 MSC:
00-01
99-00

ABSTRACT

Electricity theft is a problem that affects the efficiency and profitability of power companies. There are several studies and applications in order to detect electricity theft, including the use of artificial intelligence techniques and the most recent deep learning methods. For problems like it, the datasets utilized are completely unbalanced – consequently, the use of metrics as accuracy is not enough to properly evaluate the performance of the method for the application. In the present paper a Convolutional Neural Network (CNN) is applied to electricity theft detection problem using several techniques for balancing the classes of the dataset: Cost-Sensitive Learning, Random Oversampling, Random Undersampling, K-medoids based Undersampling, Synthetic Minority Oversampling Technique, and Cluster-based Oversampling. The objective is to compare and select the best unbalanced data-handling technique for CNN, utilizing a specific metric for problems with extremely unbalanced classes – the AUC (Area Under Receiver Operating Characteristic Curve). The results present that some techniques combined to CNN reach values of high quality, comparable to the obtained by other classifiers. Finally, the paper points studies related to electricity theft detection must deal with the unbalanced characteristic of the dataset in order to achieve better (or, in other words, correct) results.

1. Introduction

The generation, transmission, and distribution of electric energy are subjected to losses occurrences. Losses can be defined as the difference between the injected and the measured energy delivered to consumers [1].

The losses can be classified into technical and non-technical. First one is related to physical properties like Joule effect in electrical system components [2]. The second, also called commercial losses, is mainly due to electricity theft, which usually includes bypassing the electricity meter, tampering the meter reading, or hacking the meter [3].

According to a study from 2015 conducted by Northeast Group, world economy losses US$ 89.3 billion per year due to non-technical electricity losses [4]. Only India losses US$ 16.2 billion per year, while Brazil losses US$ 10.5 billion and Russia US$ 5.1 billion [5]. Some consequences of these losses are the increase of the electricity prices for paying costumers, the reduction of capital for future investments in power utilities, potential damages in the electricity transportation infrastructure or in the consumers side, the needs of investments from the government when it could invest in other areas, and more.

Non-technical loss (NTL) detection techniques performed by data-driven methods based in artificial intelligence have been very utilized [6], like Support Vector Machine (SVM) [7–9], Artificial Neural Network (ANN) [10,11], clustering techniques [12,13] and others. In addition, Deep Learning have been successful in applications such as computer vision [14], speech recognition [15] and natural language processing [16]. It represents a possibility to apply in NTL detection [17,18].

Energy theft detection problems use datasets that are extremely unbalanced, because the most part of the dataset is composed by non-theft consumers [19]. [20] explains that accuracy metric is not reliable when there is a highly unbalanced dataset. In an example utilized by [20], if a dataset contains 1% of a credit card fraudulent transactions and the model classifies all transactions as legitimate, then it achieved 99% of accuracy – but it failed to classify any fraudulent activity. There are specific and more suitable metrics to be applied in this cases, like AUC (Area Under ROC Curve) [20].

The problem of unbalanced datasets when utilized in machine learning and/or deep learning applications is known in the literature of the area. All the following examples are related to this characteristic,

* Corresponding author.
E-mail addresses: jeanneop22@gmail.com (J. Pereira), saraiva@ufpa.br (F. Saraiva).

https://doi.org/10.1016/j.ijepes.2021.107085
Received 31 March 2021; Accepted 2 April 2021
Available online 1 May 2021

and all of them must to deal with this: evaluation of sampling techniques in a wide variety of applications, like software engineering measurements, mammography and datasets from UCI Machine Learning Repository (glass, german-credit, solar-flare and others) [21]. [22–26] presents detection of NTLs using sampling techniques with machine learning methods like SVM, ANN, AdaBoost, LR, RF, KNN and XGBoost. To detect electricity theft, CNN with SMOTE (Synthetic Minority Oversampling Technique) is used in [27] and CNN without unbalanced data handling techniques is used in [1,17,18].

This paper proposes a comparative study between several approaches to handle unbalanced dataset in the context of electricity theft detection. The objective is to compare the techniques when applied to a Convolutional Neural Network (CNN), a deep learning method applied to classification problems. In order to perform the analysis, the authors implemented a CNN as presented in [17]. The comparisons are done using AUC, a metric more suitable for problems with the unbalanced classes characteristic.

According to the literature review, techniques to handle unbalanced data are used in a few applications in NTL detection and none of them applies CNN combined to the following unbalanced data handling techniques: Random Undersampling (RUS)[21], Random Oversampling (ROS)[21], Cluster-based Oversampling (CBOS)[28], Synthetic Minority Oversampling Technique (SMOTE) [29], class weighting [30] and k-medoids based undersampling [31].

This paper complements and extends the results presented in [6], where several machine learning techniques were combined with different strategies for datasets balancing and compared to each other. Differently from the previous paper, the current article uses CNN, a method from the deep learning network family methods aimed for classification problems.

The main contributions of this paper can be divided in three different fronts: (1) the characterization of NTL detection problem as a problem with extremely unbalanced datasets; (2) consequently the need to use more suitable metrics to evaluate classifiers, as AUC, and the need to apply unbalanced data handling techniques, as the 6 techniques implemented here; (3) finally, the comparison and analysis of these techniques applied to a CNN classifier method.

This paper is organized in the following way: Section 2 presents several related works, including a discussion about unbalanced class problems in electric energy area; Section 3 describes the unbalanced data handling techniques utilized in this study; Section 4 presents the proposed work with CNN and unbalanced data handling techniques; Section 5 presents and discusses the results; and finally the conclusions and future works are presented in Section 6.

2. Related work

The use of strategies to balance data is important for classification problems. Most of the classifiers without these strategies towards a bias to majority class elements and they can fail to classify minority class elements [20].

The survey [32] discusses several techniques to handle unbalanced data with machine learning and/or deep learning methods. Random Undersampling (RUS) [21], Random Oversampling (ROS) [21], Cluster-based Oversampling (CBOS) [28], SMOTE [29] and Cost-Sensitive Learning [30] are ways to handle. Other approach is k-medoids based undersampling [31]. These techniques above will be more detailed in the next section.

There are also some papers [1,17,18,22–27,33–35] about machine and/or deep learning methods, including CNN without and with unbalanced data handling techniques applied to electrical energy anomaly or electricity theft detection. [35] suggests a cost-sensitive approach using class weights with machine learning methods to detect NTLs.

The metrics AUC and F1-score are used in [1,18,21–27,34,35] because they are more suitable when classifiers deal with unbalanced datasets. [21–23,25–27,35] explains that only accuracy is not too useful.

[33] applies a probabilistic neural network (PNN) and a mathematical model based on Levenberg–Marquardt (LM) method to detect electricity theft. The methods are applied individually and combined. In the first experiment two classifications methods are compared using accuracy: PNN and SVM, PNN achieved a better value of accuracy than SVM. In the fourth experiment the combined method using PNN and LM achieves good results in a large dataset.

[34] applies a unsupervised technique called Local Matrix Reconstruction (LMR) to detect electricity theft and it compares to Local Outlier Factor (LOF) and Gaussian Kernel Local Outlier Factor (GKLOF). The algorithms are tested in a small and in a large dataset, they are compared using AUC. The LMR achieved better AUC values than LOF and GKLOF in both datasets.

In [22] SVM and ANN are used with a strategy that combines oversampling and undersampling to detect NTLs. It cites another work [36] that suggests the use of other metrics besides accuracy, then the metrics utilized in the comparison were accuracy, TPR, precision, F1-score, True Negative Rate (TNR), Fβ-score, AUC and Matthews Correlation Coefficient (MCC). The first analysis is conducted comparing classifiers with no oversampling and maximum oversampling. Different from other classifiers, Linear SVM is the only classifier that did not have an improvement in most of the metrics with maximum oversampling. The second analysis uses different oversampling proportions, it is shown the results for the MCC, AUC and Fβ-score, again only Linear SVM did not perform well.

[23] develops an algorithm to NTL detection that combines Random Undersampling with AdaBoost (Adaptive Boosting), called RUSBoost. In the preprocessing stage to extract features, it is used Maximal Overlap Discrete Wavelet-Packet Transform (MODWPT), applied before RUSBoost. The analysis used the same metrics as the discussed in [22], except the F1-score. The paper points out that accuracy is an inadequate performance measure for unbalanced datasets. RUSBoost algorithm surpasses other machine learning frameworks like Linear SVM, Non-Linear SVM and ANN. In addition, when they are applied with the preprocessing, RUSBoost is better than the machine learning frameworks except for the TNR metric.

In [24] the techniques LR, SVM, RF and KNN are applied to NTL detection. Different percentages of NTL were generated from undersampling. The models used just time series or also included neighborhood features and selected master data. AUC was used to evaluate the models. There are two analysis, in the first each one of the several NTL proportions was used to train and to test, the second analysis uses the best proportions found in the first analysis to train and all NTL proportions to test. For the first analysis, LR, SVM and KNN presented the best results for a balanced dataset of 50%, RF presented the best results for 60% considering only time series and 40% considering all features. For the second analysis, KNN did not perform well achieving the best result in one proportion, RF achieved the best AUC but it achieved the minimum AUC too.

[25] applies two undersampling approaches: first based in bad inspections and second randomly. XGBoost presented the best results without undersampling in comparison with SVM, LR and KNN, XGBoost was the only tested with undersampling variations. To evaluate are used the metrics AUC and precision-recall curve, recall is the same of TPR.

In [26] the purpose is to evaluate NTL detection using maximization of economic return. One dataset contains synthetic frauds, the other is a real dataset with 6% of fraud and other features. Oversampling was applied in the second dataset before RF, LR and ANN, the maximum economic return was achieved with RF for F1-score, precision and TPR.

[27] uses an architecture that combines CNN and Long-Short Term Memory (LSTM) to detect electricity theft in a labeled dataset. SMOTE is used to balance the data. To analyze different forms to split consumption (weekly, fortnightly or monthly) is used the metrics like precision, TPR, F1-score and accuracy. The results for different period splits did not present a significant difference, then consumption per day is adopted. The same metrics are used to compare CNN-LSTM before and after

applying SMOTE. CNN-LSTM with SMOTE is also compared to LR and SVM. After applying SMOTE, the metrics for theft user class are improved while for normal user had a small decrease, CNN-LSTM with SMOTE outperforms LR and SVM in all metrics. One thing to be mentioned is that not only train set but also test set is balanced, then this has a different result when just the train set is balanced.

In [1] is utilized Decision Tree, RF, ANN, LSTM, Autoencoder and CNN to detect NTLs in a labeled dataset, that was generated by the authors. For each classifier the metrics per class used are precision, TPR and F1-score. To compare them are used accuracy, AUC and F1 average. CNN presented the best performance over the other techniques, ANN is better than the two other machine learning techniques, similar to CNN, and the deep learning methods LSTM and Autoenconder are better than ANN in terms of accuracy and F1 average, but considering AUC they are not.

In [17] several data-driven methods are compared to address energy theft detection problem. A combined Convolutional Neural Network (CNN) model is proposed there and it is compared to a Single CNN model, Simple Deep Neural Network (DNN) (or ANN because of it has only four layers), SVM, Random Forest (RF) and Logistic Regression (LR). Furthermore, the results of CNN in [17] are compared to another CNN in [18], that uses the same dataset. There is not any mention if a technique to handle unbalanced data was used. The accuracies had good values ($> 91\%$), but other metrics were not presented like AUC and TPR.

Table 1 shows a summary of the related work. The last row of the Table highlights the main contribution and the research gap covered by this paper in comparison to others papers from the literature.

This section exposed some works that apply unbalanced data handling techniques [21–27], and deal with electrical anomaly or electricity theft detection [1,17,18,22–27,37,38,35]. As far as we know, there is not a paper in literature that combines and compares CNN with the following unbalanced data handling techniques: RUS, ROS, CBOS, SMOTE, class weighting, and k-medoids based undersampling to detect electricity theft.

The next section talks about these techniques to handle unbalanced data.

3. Techniques to handle unbalanced data

This section explains briefly the techniques to handle unbalanced data used in this study: Cost-Sensitive Learning [30], Random Undersampling [21], Random Oversampling [21], k-medoids based undersampling [31], SMOTE (Synthetic Minority Oversampling Technique) [29], and Cluster-based Oversampling [28]. These techniques will be utilized in the study developed in this paper.

Table 1
Difference between this paper and other papers from the literature of NTL detection.

Paper	Classifier	Unbalanced data handling technique
[27]	CNN-LSTM	SMOTE
	SVM, LR, CNN-LSTM	–
[1]	Decision Tree, RF, ANN LSTM, Autoenconder, CNN	–
[18]	LR, RF, SVM, ANN, CNN	–
[17]	LR, RF, SVM, ANN, CNN	–
[6]	LR, RF, SVM, ANN	class weights, random undersampling, random oversampling, k-medoids based undersampling, SMOTE, cluster-based oversampling
This paper	CNN	class weights, random undersampling, random oversampling, k-medoids based undersampling, SMOTE, cluster-based oversampling

3.1. Cost-Sensitive Learning (Weighting)

Cost-sensitive approach is implemented as class weights. The weights are defined inversely proportional to their frequencies [30]. From this moment on the article 'class weights' will be referred as 'weighting'. Despite of the minority class has less number of elements, it has more weight to compensate this characteristic.

3.2. Random Undersampling (RUS)

In RUS [21] majority training samples are selected randomly to be removed. Then the quantity of the majority elements becomes equal or almost equal to minority elements.

3.3. Random Oversampling (ROS)

In ROS [21] minority training samples are selected randomly to be replicated. Then the quantity of the minority elements becomes equal or almost equal to majority elements.

3.4. K-medoids based undersampling

In work [31] the majority class elements in the training set are undersampled with k-medoids. From this moment on the article 'k-medoids based undersampling' will be referred just as 'k-medoids'. The number of clusters is equal to the number of minority training examples and then they and the medoids are used in the training. This results in a balanced training set because the number of class elements is equal.

3.5. Synthetic Minority Oversampling Technique (SMOTE)

SMOTE [29] interpolates existing minority training samples and their nearest minority neighbors to generate artificial minority samples. This procedure increases the number of minority samples to reach the number of majority samples and the classes have the same number of elements.

3.6. Cluster-based Oversampling (CBOS)

Proposed by [28], each class is clustered separately by k-means. After that, Random Oversampling starts. In the majority class, all clusters without the largest one are randomly oversampled to contain the same numbers of elements of the largest one.

Let *maxclasssize* be the total number of largest class elements. Each cluster in the minority class is randomly oversampled until the size of each cluster is equal to $maxclasssize/Nsmallclass$, where *Nsmallclass* is the number of subclusters in the small class [28]. Finally majority class and minority class have the number of elements equal to *maxclasssize*.

4. Proposed work

In this section the dataset and architecture of CNN is described. As mentioned in Section 1, only accuracy is not reliable when there is an unbalanced dataset, then the AUC metric is presented.

4.1. Dataset description

The dataset utilized in this paper was released by State Grid Corporation of China (SGCC) [18], the same used in [17]. It is a real world labeled dataset containing 42372 users within a time interval of 1035 days (01/01/2014–10/31/2016), with 38757 normal consumers and 3615 electricity thieves – approximately 91.47% are normal and 8.53%

are anomalous consumers. As in [17], one more day was added to complete 148 weeks (1036 = 148 × 7).

The instances, that are represented by the consumers, were selected randomly to compose training and testing sets, where 80% was utilized for training and 20% for testing. Each one of the 7 unbalanced data handling techniques with CNN was executed 10 times using a different selection of training and testing sets.

The steps of the program execution are showed in Fig. 1.

The software utilized were Python programming language, Keras[1] with Tensorflow[2] as backend, Scikit-learn[3], Numpy, Pandas, Pyclustering[4] and Imbalanced-learn[5].

4.2. Preprocessing

The missing values, when an instance does not present a value for an attribute [39], were filled according to the interpolation presented in [17] and they are showed in the Eq. (1), where x_i is the value of electricity consumption over a specific time period (e.g., a day).

$$f(x_i) = \begin{cases} 0, & x_i \in NaN, i = 1; \\ x_{i-1}, & x_i \in NaN, i > 1; \\ x_i, & x_i \notin NaN; \end{cases} \quad (1)$$

4.3. Convolutional Neural Network

Convolutional Neural Network (CNN) is a deep learning technique very used for image classification, but can be used for other types of classification like presented in [17,18]. It is composed by convolution layers and pooling layers intercalated, besides that contains one or more fully connected layers after the convolution and pooling layers [40]. The implemented CNN has three pooling layers and three convolution layers. This architecture was based in a CNN presented in [17].

CNN was used with class weights, ROS, RUS, k-medoids based undersampling, SMOTE and CBOS. It used the same parameters of the paper [17], presented in Table 2. Its loss function is categorical cross-entropy with *softmax* as activation function in the last layer. The others layers use Rectified linear unit (Relu) as activation function, where FC means Fully Connected. The optimizer was SGD, the number of epochs was 100, the batch size was 128. Table 2 presents other characteristics of the CNN implemented.

The SMOTE technique was tested with different values of k (number of neighbors), they are taken from the list {2, 3, 5, 7, 10, 20, 30, 40, 50, 60, 70, 100} and it was selected the k associated with the best value of AUC.

In CBOS, the number of clusters varied from 2 to 11. The elbow method and the silhouette coefficient were used to determine the best number of clusters [39].

4.4. Evaluation metrics

The results were evaluated using AUC (Area Under ROC Curve) and Accuracy.

Accuracy is the number of instances correctly classified divided by the total of instances and it is calculated as presented in Eq. (2), where, TP is True Positives, TN is True Negatives, FN is False Negatives, and FP is False Positives [39].

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP} \times 100\% \quad (2)$$

As discussed previously, accuracy is not suitable for classification problems where the dataset is unbalanced. This metric is utilized in the study just to show how it behaves in problems like this – and how a researcher can assumes wrongly the "good" (and in fact, incorrect) results presented by accuracy.

TPR is the percentage of real anomalies that were predicted by the classifier correctly and it is presented in (3). False Positive Rate (FPR) is the percentage of real normal consumers that were predicted like anomalous in (4) [20].

$$TPR = \frac{TP}{TP + FN} \times 100\% \quad (3)$$

$$FPR = \frac{FP}{TN + FP} \times 100\% \quad (4)$$

ROC (Receiver Operating Characteristic) curve shows the trade-off between TPR and FPR of a classifier in a bi-dimensional chart. The area under ROC curve is represented by AUC, where the values varies between 0.5 and 1. AUC equals 1 indicates that the model is perfect while AUC equals 0.5 indicates that model performs a random guessing [20].

The next section shows the results using the metrics cited to evaluate CNN without and with unbalanced data handling techniques and discusses the results obtained.

5. Results and discussion

The simulations were performed 10 times and the AUC and accuracy metrics were calculated. These metrics are presented in Table 3 and in Table 4, respectively. Both tables show the mean, standard deviation (Std) and best values for each metric. In these tables the better values for each metric are highlighted in a box.

In Table 3, considering only the mean of AUC, all the applied unbalanced data handling techniques reached values greater than 0.6. When the CNN is applied without any unbalanced data handling technique, the mean of AUC is 0.5162 - a value worse than all the techniques utilized in this study, as expected.

ROS achieved the best AUC mean (0.6714) and the second best AUC result (0.6813). CBOS achieved the best AUC (0.6833), but this value is not too different than the reached by ROS - the difference between them is just 0.002. In addition, as the standard deviation of ROS is very low when compared to the obtained by CBOS (0.0062 and 0.0299, respectively), it is possible to conclude ROS obtained a better performance than CBOS.

Table 4 presents results related to accuracy metric. This table has only the purpose to show how this metric can be misleading for problems like the faced here - a classification problem whose the classes are extremely unbalanced.

For Table 4, the No balance "method" achieved the best results for mean accuracy (91.59%), standard deviation (0.05%), and best accuracy obtained (91.65%). In fact, analyzing the mean accuracy, the better unbalanced data handling techniques achieved just 72% - CBOS (72.06%) and RUS (72.01%).

However, it is important to point despite the high accuracy, the CNN without unbalanced data handling techniques learned how to classify the conventional consumers, while the electricity thieves are not been classified correctly.

The Fig. 2 shows the boxplot for AUC considering the 10 executions for each unbalanced data handling technique. The small variation is found out in k-medoids, while CBOS presents a large one. In the figure, it is possible to see ROS reaches a good commitment between high quality values and low variation.

The Fig. 3 shows the boxplot for accuracy considering the 10 executions for each technique. In the figure, it is possible to see the "no balance" method reaching values higher than 90%, while the other methods have low ones. In addition, "no balance" has also low variation

[1] https://keras.io/
[2] https://www.tensorflow.org/
[3] https://scikit-learn.org/
[4] https://pypi.org/project/pyclustering/
[5] https://imbalanced-learn.readthedocs.io/en/stable/index.html

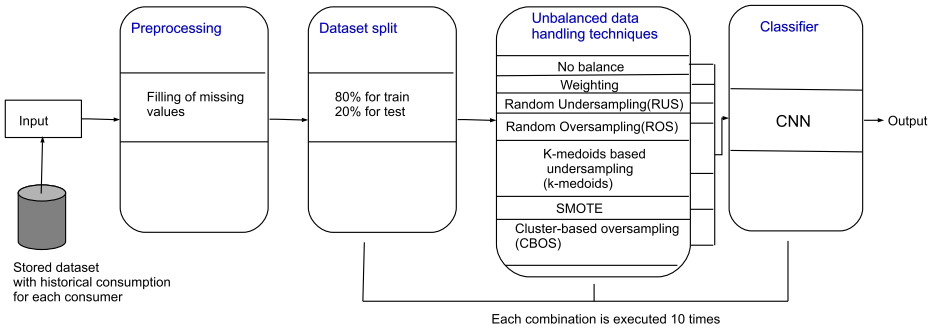

Fig. 1. Execution stages.

Table 2
Model architecture [17].

Single CNN
Input (148,7,1)
Conv2d layer1 (148,7,16)
Pooling layer1 (37,6,16)
Conv2d layer2 (37,6,32)
Pooling layer2 (9,5,32)
Conv2d layer3 (9,5,64)
Pooling layer3 (3,2,64)
FC layer1 (128)
FC layer2 (32)
FC layer3 (2)

Table 3
CNN AUC results.

Method	Mean AUC	Std of AUC	Best AUC
No balance	0.5162	0.0045	0.5235
Weighting	0.6580	0.0088	0.6680
RUS	0.6448	0.0051	0.6501
ROS	0.6714	0.0062	0.6813
K-medoids	0.6385	0.0029	0.6417
SMOTE	0.6500	0.0065	0.6571
CBOS	0.6310	0.0299	0.6833

Table 4
CNN Accuracy results.

Method	Mean Accuracy	Std of Accuracy	Best Accuracy
No balance	91.59%	0.05%	91.65%
Weighting	63.94%	4.67%	73.66%
RUS	72.01%	4.34%	78.61%
ROS	67.78%	4.02%	74.36%
K-medoids	69.87%	0.65%	70.88%
SMOTE	71.54%	5.54%	77.44%
CBOS	72.06%	15.28%	80.84%

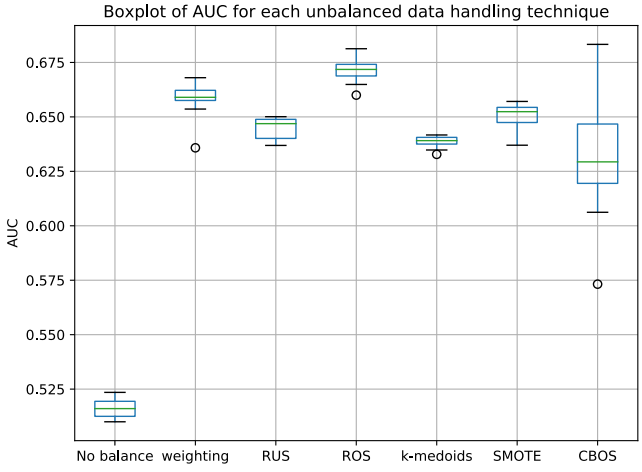

Fig. 2. Boxplot of AUC.

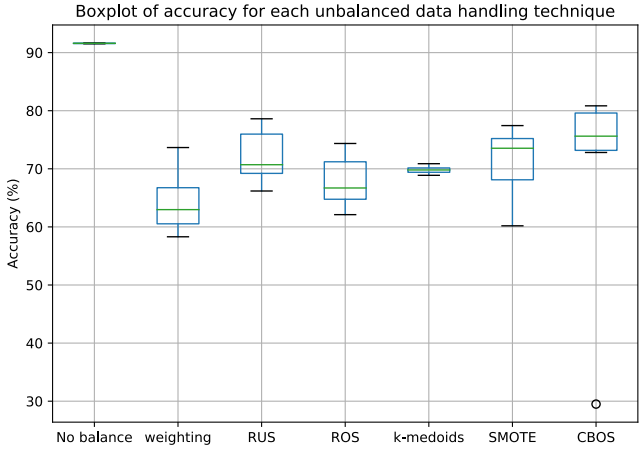

Fig. 3. Boxplot of accuracy.

when compared to the other methods.

Although "no balance" presented a high accuracy, it presented a low mean AUC, implying that is closer to a random guessing. All unbalanced data handling techniques presented a mean AUC greater than 0.6 and a mean accuracy greater than 60%. ROS presented the best mean AUC. "No balance" presented the best values for accuracy, as expected.

Despite accuracy is not a suitable metric for unbalanced datasets, it is used to illustrate the relation to other metrics. AUC combines TPR and FPR (1 - TNR) and their results are very interesting for unbalanced datasets. The ROC curves for each unbalanced data handling technique with CNN is showed in Fig. 4, the dashed diagonal line represents when the AUC is equal to 0.5 - or, in other words, a random guessing. The good ROC curves are above the dashed line. Like presented by the Table 3 for the mean AUC, the area for SMOTE is closer to the area for RUS.

The best combination obtained was CNN + ROS. This combination reaches good quality values including when compared to other classification methods applied to NTL detection. The mean AUC obtained by CNN + ROS (0.6714) is very close to the mean AUC obtained by an Artificial Neural Network (ANN) combined to SMOTE technique (0.6792) - the best result obtained by a machine learning method combined to a data handling technique to NTL detection [6].

After the performance evaluation based in AUC metric, we will compare the methods by the memory consumption and the computational time processing.

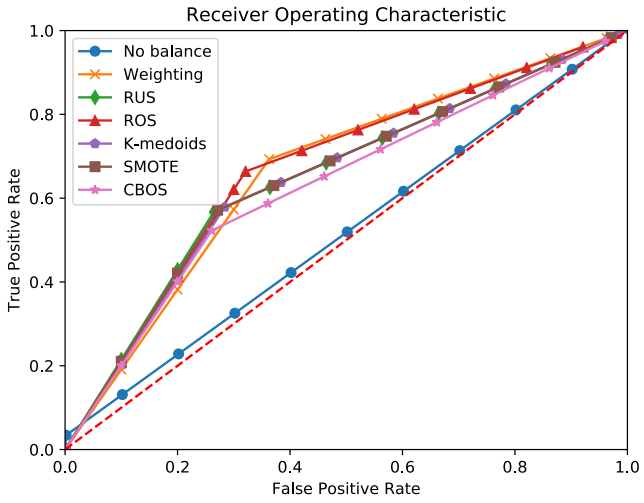

Fig. 4. CNN ROC curves.

In order to compare the memory consumption of the methods, we will use the number of samples generated by them during the training phase.

In the weighting, the number of samples is the same of the original dataset as the weights are defined inversely proportional to the class frequencies. The class of anomalous consumers has the weight equals to 5.86 while the class of normal consumers has the weight 0.55.

In RUS, the number of normal samples is smaller than the number in the original dataset because the method performs a "undersampling" of the data. For the utilized dataset, both the anomalous and normal classes have 2892 elements - therefore, the total elements is 5784.

In the other hands, for ROS the method performs a "oversampling" od the data. For this case, both normal and anomalous classes have 31005 elements, therefore the total of elements is 62010.

In the k-medoids, the quantities are similar to RUS - the anomalous class has 2892 elements and the normal class has also 2892 elements.

In SMOTE the total elements is 62010 for all the neighbors, where 31005 are normal elements and 31005 are anomalous - similar to ROS. In SMOTE, 28113 elements of anomalous class were generated.

In CBOS the number of elements varies for each execution. For the 10 executions performed, the number of normal class elements were 155005, 124008, 93009, 155005, 155005, 124008, 155005, 93009, 93009, 124008, while the number of anomalous class elements were 155007, 124008, 930012, 155007, 155008, 124008, 155008, 93009, 930012, 124008. The total of elements for each of the 10 executions were 310012, 248016, 186021, 310012, 310013, 248016, 310013, 186018, 186021, 248016.

Table 5 shows the sizes of the training dataset before and after the application of unbalanced data handling techniques.

The SMOTE, ROS and CBOS have the higher memory consumption because the samples are increased by these methods. CBOS, in special, is a case where the number of samples are enormously increased - the small number of samples in CBOS has 186018 elements.

In the other hands, RUS and k-medoids decreased the number of samples - for both it is 5784 elements, consuming less memory than the other methods.

Table 6 shows the computational times for training and inference (test) phases for the CNN. The numbers presented for training column are the average time for the 10 executions, while for inference collumn are the average time for the whole test phase.

In SMOTE, the time to find the neighbor with best AUC for each execution is taking into account in inference time, while in CBOS the training time includes the time to select the best number of clusters.

Both SMOTE and CBOS have the highest training time and inference time. RUS and k-medoids have the fastest time because they reduce the number of instances. ROS, the method that achieved the best AUC, had a good processing time for training when compared to the others techniques analyzed.

Analysing together the AUC performance, memory consumption, and computational processing time, ROS has a good commitment between these metrics - the AUC is the best and, despite the memory consumption be high, it is not too high than other methods (CBOS, specifically). Meanwhile, the computational processing time for both training and testing phases are acceptable.

Table 5
Dataset original and new sample sizes.

Method	Original size	Size after unbalanced data handling technique
No balance	33897	33897
Weighting	33897	33897
RUS	33897	5784
ROS	33897	62010
K-medoids	33897	5784
SMOTE	33897	62010
CBOS	33897	{310012, 248016, 186021, 310012, 310013, 248016, 310013, 186018, 186021, 248016}

Table 6
CNN training time and inference time.

Method	CNN training time	CNN inference time
No balance	44 min and 7.98s	12.58s
Weighting	46 min and 34.74s	15.85s
RUS	12 min and 47.82s	9.13s
ROS	1 h and 16.79s	13.52s
K-medoids	26 min and 51.67s	11.66s
SMOTE	8 h and 12.56 min	4 min and 12.71s
CBOS	8 h and 1.65 min	30.59s

6. Conclusions

In this paper, unbalanced data handling techniques were used to improve the NTL detection in a dataset that contains electricity theft. The utilization of accuracy is not appropriate when there is an unbalanced dataset, like presented in several studies of the area. For this case, other metrics like AUC are most recommended and suitable.

In this work, unbalanced data handling techniques were utilized in combination to CNN. They were evaluated using AUC and accuracy. The ROS technique presented the best results in AUC, reaching the value 0.6714. Other techniques obtained better standard deviation (k-medoids: 0.0029) and best AUC (CBOS: 0.6833), but ROS has a good commitment combining low results variation and better results quality. That values are also as good as other techniques found in literature, as the combination between ANN and SMOTE.

In other aspects, as memory consumption and computational time processing, ROS also had a good commitment. For both metrics, ROS is not a method which uses the less memory and less computational time consumption, however there are methods such uses much more computational resources than ROS - for example, CBOS and SMOTE.

For future works the values of AUC can be improved through implementation of different CNN architectures. We would like to investigate different parameters for CNN architecture in order to improve those metrics. In addition, the application of this study for others machine learning methods is also in our plans for future works.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research was supported by CNPQ (Brazilian National Council for Scientific and Technological Development), Grant No. 132015/2018–8. The authors would like to thank the agency for funding this work.

References

[1] Bhat R, Trevizan R, Sengputa R, Li X, Bretas A. Identifying Nontechnical Power Loss via Spatial and Temporal Deep Learning. In: in: 15th IEEE International Conference on Machine Learning and Applications ICMLA; 2016. p. 272–9. https://doi.org/10.1109/ICMLA.2016.0052.

[2] Sankari E, Rajesh R. Detection of Non-Technical Loss in Power Utilities using Data Mining Techniques. International Journal for Innovative Research in Science & Technology 2015;1(9):97–101.

[3] McLaughlin S, Holbert B, Fawaz A, Berthier R, Zonouz S. A Multi-Sensor Energy Theft Detection Framework for Advanced Metering Infrastructures. IEEE J. Sel. Areas Commun. 2013;31(7):1319–30. https://doi.org/10.1109/JSAC.2013.130714.

[4] Global Markets, Solutions, and Vendors 2017.

[5] Northeast Group - LLC, World Loses 89.3*BilliontoElectricityTheftAnnually*, 58.7 Billion in Emerging Markets, URL:https://www.prnewswire.com/news-releases/world-loses-893-billion-to-electricity-theft-annually-587-billion-in-emerging-markets-300006515.html, Access in 11/02/2019.

[6] Pereira J, Saraiva F. A comparative analysis of unbalanced data handling techniques for machine learning algorithms to electricity theft detection, in. IEEE Congress on Evolutionary Computation (CEC) 2020;2020:1–8. https://doi.org/10.1109/CEC48606.2020.9185822.

[7] Pal K, Chauhan B. Support Vector Machine Approach for Non-Technical Losses Identification in Power Distribution Systems. International Journal on Recent and Innovation Trends in Computing and Communication 2018;6(1):158–62.

[8] S. Depuru, L. Wang, V. Devabhaktuni, Support vector machine based data classification for detection of electricity theft, in: 2011 IEEE/PES Power Systems Conference and Exposition, IEEE, 2011, pp. 1–8. doi:10.1109/PSCE.2011.5772466.

[9] Nagi J, Yap K, Tiong S, Ahmed S, Mohamad M. Nontechnical Loss Detection for Metered Customers in Power Utility Using Support Vector Machines. IEEE Trans. Power Delivery 2010;25(2):1162–71. https://doi.org/10.1109/TPWRD.2009.2030890.

[10] H. Huang, S. Liu, K. Davis, Energy Theft Detection Via Artificial Neural Networks, in: 2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe), 2018, pp. 1–6. doi:10.1109/ISGTEurope.2018.8571877.

[11] Costa B, Alberto B, Portela A, Maduro W, Eler E. Fraud Detection in Electric Power Distribution Networks using an Ann-Based Knowledge-Discovery Process. International Journal of Artificial Intelligence & Applications 2013;4:17–23. https://doi.org/10.5121/ijaia.2013.4602.

[12] L.A.P. Júnior, C.C.O. Ramos, D. Rodrigues, D.R. Pereira, A.N. de Souza, K.A.P. da Costa, J. ao Paulo Papa, Unsupervised non-technical losses identification through optimum-path forest, Electric Power Systems Research 140 (2016) 413 – 423. doi:10.1016/j.epsr.2016.05.036.

[13] Viegas JL, Esteves PR, Vieira SM. Clustering-based novelty detection for identification of non-technical losses. International Journal of Electrical Power & Energy Systems 2018;101:301–10. https://doi.org/10.1016/j.ijepes.2018.03.031.

[14] A. Krizhevsky, I. Sutskever, G. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, in: 25th International Conference on Neural Information Processing Systems, Vol. 1, 2012, pp. 1097–1105.

[15] G. Hinton, L. Deng, D. Yu, G. Dahl, A. r Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, B. Kingsbury, Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups, Signal Processing Magazine, IEEE 29 (2012) 82–97. doi:10.1109/MSP.2012.2205597.

[16] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, S. Khudanpur, Recurrent neural network based language model, in: Proceedings of the 11th Annual Conference of the International Speech Communication Association, INTERSPEECH 2010, Vol. 2, 2010, pp. 1045–1048.

[17] Yao D, Wen M, Liang X, Fu Z, Zhang K, Yang B. Energy Theft Detection with Energy Privacy Preservation in the Smart Grid. IEEE Internet of Things Journal (Early Access) 2019:1. https://doi.org/10.1109/JIOT.2019.2903312.

[18] Zheng Z, Yang Y, Niu X, Dai H, Zhou Y. Wide and Deep Convolutional Neural Networks for Electricity-Theft Detection to Secure Smart Grids. IEEE Trans. Industr. Inf. 2018;14(4):1606–15. https://doi.org/10.1109/TII.2017.2785963.

[19] Messinis GM, Hatziargyriou ND. Review of non-technical loss detection methods. Electric Power Systems Research 2018;158:250–66. https://doi.org/10.1016/j.epsr.2018.01.005.

[20] Tan P-N, Steinbach M, Kumar V. Introduction to Data Mining. 1st Edition,. Addison-Wesley Longman Publishing Co., Inc; 2005.

[21] Van Hulse J, Khoshgoftaar TM, Napolitano A. Experimental Perspectives on Learning from Imbalanced Data, in. In: Proceedings of the 24th International Conference on Machine Learning, ICML '07 ACM; 2007. p. 935–42. https://doi.org/10.1145/1273496.1273614.

[22] Figueroa G, Chen Y, Avila N, Chu C. Improved practices in machine learning algorithms for NTL detection with imbalanced data, in. IEEE Power Energy Society General Meeting 2017;2017:1–5. https://doi.org/10.1109/PESGM.2017.8273852.

[23] Avila NF, Figueroa G, Chu C. NTL Detection in Electric Distribution Systems Using the Maximal Overlap Discrete Wavelet-Packet Transform and Random Undersampling Boosting. IEEE Trans. Power Syst. 2018;33(6):7171–80. https://doi.org/10.1109/TPWRS.2018.2853162.

[24] Glauner P, Meira JA, Dolberg L, State R, Bettinger F, Rangoni Y. In: Neighborhood Features Help Detecting Non-Technical Losses in Big Data Sets, in Proceedings of the 3rd IEEE/ACM International Conference on Big Data Computing, Applications and Technologies, BDCAT '16. Association for Computing Machinery; 2016. p. 253–61. https://doi.org/10.1145/3006299.3006310.

[25] Buzau MM, Tejedor-Aguilera J, Cruz-Romero P, Gómez-Expósito A. Detection of Non-Technical Losses Using Smart Meter Data and Supervised Learning. IEEE Transactions on Smart Grid 2019;10(3):2661–70. https://doi.org/10.1109/TSG.2018.2807925.

[26] Massaferro P, Martino JMD, Fernández A. Fraud Detection in Electric Power Distribution: An Approach That Maximizes the Economic Return. IEEE Trans. Power Syst. 2020;35(1):703–10. https://doi.org/10.1109/TPWRS.2019.2928276.

[27] M.N. Hasan, R.N. Toma, A.-A. Nahid, M.M.M. Islam, J.-M. Kim, Electricity Theft Detection in Smart Grid Systems: A CNN-LSTM Based Approach, Energies 12 (17). doi:10.3390/en12173310.

[28] Jo DT, Japkowicz N. Class imbalances versus small disjuncts. SIGKDD Explorations 2004;6:40–9. https://doi.org/10.1145/1007730.1007737.

[29] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research 2002;16:321–57.

[30] C.X Ling, V. Sheng, Cost-Sensitive Learning and the Class Imbalance Problem, Encyclopedia of Machine Learning.

[31] Dubey R, Zhou J, Wang Y, Thompson P, Ye J. Analysis of sampling techniques for imbalanced data: An n = 648 ADNI study. NeuroImage 2014;87:220–41. https://doi.org/10.1016/j.neuroimage.2013.10.005.

[32] Johnson JM, Khoshgoftaar TM. Survey on deep learning with class imbalance. Journal of Big Data 2019;6(1):27. https://doi.org/10.1186/s40537-019-0192-5.

[33] Ghasemi AA, Gitizadeh M. Detection of illegal consumers using pattern classification approach combined with Levenberg-Marquardt method in smart grid. International Journal of Electrical Power & Energy Systems 2018;99:363–75. https://doi.org/10.1016/j.ijepes.2018.01.036.

[34] Feng Z, Huang J, Tang W, Shahidehpour M. Data mining for abnormal power consumption pattern detection based on local matrix reconstruction. International Journal of Electrical Power & Energy Systems 2020;123:106315. https://doi.org/10.1016/j.ijepes.2020.106315.

[35] Ghori KM, Abbasi RA, Awais M, Ullah A, Szathmary L. Performance Analysis of Different Types of Machine Learning Classifiers for Non-Technical Loss Detection. IEEE Access 2019:1. https://doi.org/10.1109/ACCESS.2019.2962510.

[36] Glauner P, Boechat A, Dolberg L, State R, Bettinger F, Rangoni Y, Duarte D. Large-scale detection of non-technical losses in imbalanced data sets, in. IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT) 2016;2016:1–5. https://doi.org/10.1109/ISGT.2016.7781159.

[37] Ramos CC, Souza AN, Chiachia G, Falcão AX, Papa JP. A novel algorithm for feature selection using harmony search and its application for non-technical losses detection. Computers & Electrical Engineering 2011;37(6):886–94. https://doi.org/10.1016/j.compeleceng.2011.09.013.

[38] Pereira DR, Pazoti MA, Pereira LA, Rodrigues D, Ramos CO, Souza AN, Papa JP. Social-spider optimization-based support vector machines applied for energy theft detection. Computers & Electrical Engineering 2016;49:25–38. https://doi.org/10.1016/j.compeleceng.2015.11.001.

[39] Han J, Pei J, Kamber M. Data Mining: Concepts and Techniques. 3rd Edition,. Morgan Kaufmann Publishers Inc.; 2012.

[40] Patterson J, Gibson A. Deep Learning: A Practitioner's Approach. 1st Edition,. O'Reilly Media Inc; 2017.