Handling imbalance in an extended PLAID

Leen De Baets, Chris Develder, Tom Dhaene and Dirk Deschrijver
Department of Information Technology,
Ghent University - imec,
Technologiepark-Zwijnaarde 15,
9052 Ghent

Jingkun Gao and Mario Berges
Civil & Environmental Engineering
Carnegie Mellon University,
Pittsburgh, PA 15213-3890

***Abstract*—The ability to classify appliances, given the current and voltage consumption of a household is useful for a variety of applications, including demand response verification, and eco-feedback technologies. To support research efforts in this problem domain, this paper presents an extended version of the Plug-Level Appliance Identification Dataset (PLAID), which is called PLAID 2 and contains 30 kHz voltage and current measurements of different residential appliances as they are switched on. As an extension to PLAID, this dataset adds appliance instances as well as measurements for multiple operating modes (e.g., low or high fan settings for air conditioners). As with other datasets in this problem domain, the appliance classes are not equally represented in PLAID 2. Different techniques for handling this imbalance and avoiding biasing the classifiers during training are investigated. The results indicate that performance improvement depends on the classifier type, when binary VI images are used as input.**

I. Introduction

The ability to automatically identify electrical appliances from measurements of voltage and/or current could unlock and facilitate a growing number of smart grid applications including measurement verification for demand response programs, direct load control ensuring quality-of-service and load forecasting methods. Also the non-intrusive load monitoring (NILM) techniques, which extract the power consumption of electrical appliances out of aggregated power data, use methods to automatically identify electrical appliances. Public datasets, like PLAID [1], WHITED [2], and COOLL [3] that contain measurements of voltage and current for a variety of domestic appliances, are made available for the purpose of furthering research into algorithmic techniques for appliance classification. However, it is known that some imbalance is present in these datasets: some appliance types are represented by more measurements than others, as can be seen in Table I. This can influence the performance achieved by a particular classifier trained using this data.

The Plug-Level Appliance Identification Dataset (PLAID) was presented in [1] as a public and crowd-sourced dataset consisting of short voltage and current measurements of the activation for different household appliances. The goal of PLAID was to provide a public library for high-resolution appliance activation measurements that can be integrated into existing or novel appliance classification algorithms. The activation measurements contain a few seconds before and after the activation of the different household appliances. PLAID currently contains activation measurements for more than 200 different appliance instances, representing 11 appliance classes, and has more than a thousand records. This paper presents an extension of this dataset, PLAID 2, where the number of appliance instances is increased and the activation measurements include different modes of operation. For example, a fan spinning at high, medium or low speed is measured.

TABLE I
The number of instances for some selected appliances in the PLAID, WHITED, and COOLL dataset.

appliance type	PLAID [1]	WHITED [8]	COOLL [9]
AC	66	10	
CFL	175	20	
Fan	115	60	40
Hairdryer	156	60	80
ILB	114	60	80
Vacuum	38	40	140
Washing Machine	26	10	
Router			20

It is known that imbalances in the PLAID dataset [1] can influence the performance achieved by a particular classifier. The imbalance present in this dataset is caused by the large difference in the total number of measurements per appliance type. The types with the most and least samples are respectively called the majority and minority type. Similar imbalance is present in WHITED [2], and COOLL [3]. Table I lists the number of instances for some selected appliances for the three previously mentioned datasets, showing the present imbalance.

In the literature, no work is found concerning handling this appliance type imbalance in NILM. However, research exists that deals with the imbalance that is caused by the difference in the active and idle time of appliances. This is present in NILM datasets that contain consumption patterns over time. In [4], which determines how much energy a specific appliance consumes at any given moment using regression, the imbalance caused by the difference in activation and idle time of appliances is present. To handle it, they propose the usage of the target-weighted root mean squared error as an alternative error metric for optimizing the regression. In [5] where temporal sequence classification algorithms are researched, the same imbalance is counteracted with under-sampling: reducing the amount of majority samples (the idle samples) so that it equals the amount of minority samples (the active samples). It is qualitatively mentioned that this approach is preferred

978-3-901882-99-9 ©2017 IFIP

to over-sampling (increasing the amount of samples of the minority till it equals the amount of the majority) or leaving the data as-is, however no quantitative comparison of the results is shown. Increasing the dataset can be done by reusing the same measurements or by synthesizing new measurements. For the latter, one could use a smart home simulator like SmartSim [6] or AMBAL [7]. It must be mentioned that these are two frameworks for low frequency data concerning consumption patterns over time and not for high frequency data concerning the activation of appliances. Therefore, these simulators can not be used to generate PLAID-like data.

Rather than removing the imbalance from PLAID, methods for dealing with the imbalance can be used. Although no literature can be found where these methods are applied on NILM data, these methods are well investigated for classical machine learning methods. These methods can change the distribution in the dataset by resampling the classes. Some methods oversample the dataset, like in [8] where they want to predict the age and gender from images. Simple oversampling (increasing the amount of samples in the minority classes by duplicating samples) is effective, but one should be aware of overfitting [9]. To avoid overfitting, more advanced oversampling can be used, like SMOTE [9]. Another possibility to change the distribution in the dataset is to undersample the dataset like in [10] where samples of the majority class are randomly deleted. In the case of [11] where a decision tree learner is used, undersampling is preferred over oversampling. But in [12] where convolutional neural networks are used for classifying imbalanced classes, it is found that oversampling performs better than undersampling. In [13], the undersampling is done multiple times and an ensemble of classifiers is trained upon them. Another way to handle imbalance is to change the classifier so that different misclassification errors incur different penalties [14]. In this paper methods including over- and under-sampling, synthesizing samples, balanced bagging and altering the weight function will be used.

Section II introduces the novel dataset and looks at the different operating modes of an appliance. Section III examines two types of methods to deal with an imbalanced dataset: resampling the dataset and reweighing the error function. Section IV concludes the paper.

II. PLAID 2

The PLAID 2 dataset contains current and voltage measurements, sampled at 30 kHz from 9 households in Pittsburgh, Pennsylvania, USA. It includes 82 different appliances representing 11 appliance types, with a total of 719 instances. The appliance types are the same as in the original PLAID. The main difference with PLAID 1 is that different operating modes for many appliance types are included. PLAID 2 is also made available on http://plaidplug.com.

In Table II, the monitored appliance types are listed together with the corresponding number of appliances, number of instances and different operating modes. For example, there are 7 air conditioners (AC), each AC can operate in 4 possible different modes (high cool, high fan, low cool or low fan), and the total number of instances with the label AC is 142. Note that different instances of the same appliance can be located in the same house, e.g., one house can have multiple fans.

TABLE II
SUMMARY OF THE DIFFERENT APPLIANCES IN THE PLAID 2 DATASET. AC = AIRCONDITIONER, CFL = COMPACT FLUORESCENT LIGHT, ILB = INCANDESCENT LIGHT BULB

appliance type	# appliances	# instances	modes
AC	7	142	[highcool, highfan, lowcool, lowfan]
CFL	9	45	[off-on]
Fan	7	95	[high, medium, low]
Fridge	9	52	[off-on]
Hairdryer	5	92	[highwarm, lowwarm, highhot, lowhot]
Heater	6	50	[high, low]
ILB	7	34	[off-on]
Laptop	7	35	[off-on]
Microwave	9	90	[high, medium]
Vacuum	7	35	[off-on]
Washing Machine	9	49	[off-on]
Total		719	

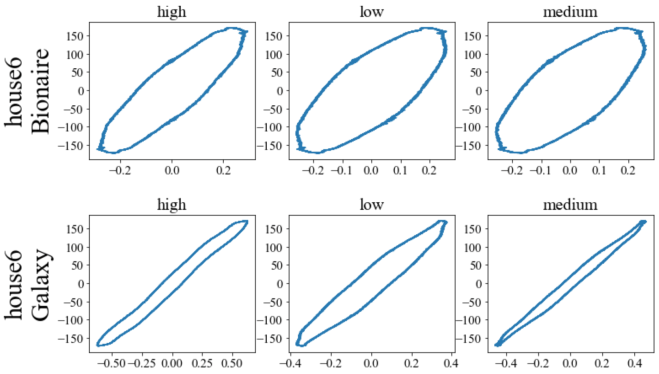


Fig. 1. The three different operating modes (high, low, medium) for two fans.

In Figure 1, the appliance signatures for two fans operating in three different modes are shown. The appliance signature is the voltage-current (VI) trajectory of an instance in steady state [15], [16]. For each fan separately, the appliance signature as well as the magnitude of the current are similar in the different modes. In fact, the operating modes of one fan are more alike than the same mode from different fans. It has been verified that this is the case for all appliance types having modes in this dataset. Similar observations can be made for other appliance types, see the notebooks on http://plaidplug.com. This observation only holds for the steady-state behavior of the appliances, as other effects (like e.g. the operating time) may have different characteristics. Further investigation is necessary to indicate what the real influence of the appliance modes is and how other (not yet measured) appliance types behave when operating in different modes.

As the operating modes do not seem to have significant influence on the appliance signature, the data in PLAID 1 and

TABLE III

SUMMARY OF THE DIFFERENT APPLIANCES IN THE PLAID 1 + 2 DATASET. AC = AIRCONDITIONER, CFL = COMPACT FLUORESCENT LIGHT, ILB = INCANDESCENT LIGHT BULB

appliance type	# appliances	# instances
AC	26	208
CFL	44	220
Fan	30	210
Fridge	27	90
Hairdryer	36	248
Heater	15	85
ILB	32	148
Laptop	45	207
Microwave	32	229
Vacuum	14	73
Washing Machine	16	75
Total		1793

2 datasets are combined, and used together for the remainder of this paper. A summary of this unified dataset is given in Table III, which confirms that the dataset is highly imbalanced. E.g., there are 248 instances for the hair dryer, but only 73 for vacuum and 75 for the washing machine. A similar imbalance is present in PLAID 1 and 2 separately. Section III offers methods handling the imbalance, so that the used classifier considers the minority types equally important as the majority types.

III. IMBALANCED DATASET

When the dataset is imbalanced, there are the majority and minority appliance types where the majority is represented by more samples than the minority. Due to the class imbalance, it is possible that the classifier focuses too strongly on the majority types during training, thereby ignoring the minority types. In this section of the paper, different methods to deal with the imbalance are evaluated so that the classifier considers the minority types equally important to the majority types. As presented in the introduction, there are two sorts of methods: modifying the dataset itself (see Subsection III-A) or adapting the classifier by adjusting the error function (see Subsection III-B). The results on PLAID 1 + 2 are presented in Subsection III-C when using the binary VI-image as input to multiple classifiers. It is important to note that these methods influence the train phase of the machine learning methods and not the test phase, i.e., the approaches respectively modify the train set and the error function used for training the classifier's parameters.

A. Modifying the dataset

Changing the dataset so that it becomes balanced can be done in several ways:

- **Over-sampling**: adjust the distribution of the dataset by replicating samples of the minority types till the amount of the majority types is reached [17]. For PLAID 1 + 2, this results in a train set of 2673 samples on average. This is a significant increase when compared to the size of the normal train set of 1769 samples.
- **Under-sampling**: adjust the distribution of the dataset by reducing the number of samples in the majority types to the amount of the minority types [17]. For PLAID 1 + 2, this results in a train set of 803 samples on average. This is a significant decrease when compared to the size of the normal train set of 1769 samples.
- **Synthesizing samples**: instead of replicating samples from the minority types, artificial samples are created. In this paper, the synthetic minority oversampling technique (SMOTE) [9] is used, where the artificial samples are formed by interpolating two neighbouring samples of a minority type. For example, having samples A and B with n features, then the new interpolated sample C is constructed by:

  $$\begin{aligned} &C[i] = A[i] + \text{gap} \times \text{diff}, \forall i \in [1, n] \\ &\text{diff} = B[i] - A[i] \\ &\text{gap} = \text{random number between 0 and 1} \end{aligned}$$

  The formula can be easily extended for multi-dimensional features by applying the formula in each dimension. For PLAID 1 + 2, this procedure results in a train set of 2673 samples on average, just like when over-sampling is performed.
- **Balanced bootstrapping** (BB): This method is proposed in [13], based on a probabilistic approach allowing to identify dataset characteristics (such as dimensionality, sparsity, etc.) that exacerbate the problem. It goes as follows:
  1) randomly select instances from the train set with replacement (bootstrap the dataset). Do this multiple times, each resulting dataset is called a bootstrap.
  2) under-sample each bootstrap, like explained above.
  3) train a classifier on each bootstrap.
  4) when classifying the test instances, the majority vote of the classification of all separate classifiers is taken as outcome.

  For this dataset, 10 bootstraps are created, each with size equal to the original train set, but now with the difference that some samples can be present more than once.

B. Adaptation of the classifier

The error function of the classifier can be changed such that misclassification of the minority types is penalized more strongly by the classifier than misclassification of the majority types. One way to achieve this, is by assigning a weight to each instance. In this paper, weights w are defined per appliance type t corresponding to the imbalance (minority types will get a higher weight) using the following equation:

$$w_t = \frac{\#\text{ instances}}{\#\text{ types} \times \#\text{ instances of type } t}$$

This definition is standard in the Python's sklearn library and is based on [18]. It must be noted that this approach is only valid for the classifiers whose error function is dependent on hyperparameters that can be tuned in order to minimize the error. For example, if the k-nearest-neighbors classifier is used with $k = 5$, giving weights to the instances does not impact the outcome as the error function is not dependent on any hyperparameter.

C. Results

For the discussion concerning methods handling imbalanced datasets, the binary VI-image is used as a feature because this was the most promising feature extracted from the study performed on PLAID 1 [15]. This feature is created by overlaying the VI-trajectory (plotting the voltage against the current) of an instance with a $n \times n$ grid and assigning a binary value (0 or 1) denoting whether it contains a sample of the trajectory or not. In this paper, $n = 16$. Multiple classifiers are used: k-nearest-neighbors (kNN), Gaussian naive Bayes (GNB), logistic regression classifier (LGC), support vector machines (SVM), linear discriminant analysis (LDA), decision tree (dTree), random forest (rForest), adaptive boosting (adaBoost). These classifiers were also applied to PLAID 1 in the previous study on PLAID [15].

As PLAID 1 + 2 is imbalanced, it is important to choose the correct evaluation metric. In [19], it is suggested to use the $F1$-measure when the classification performance is wanted. Additionally, it is noted that the accuracy should be used with caution when some appliances are rarely used. In this paper, the performance of the methods is expressed using the macro-F1 measure, which is calculated as follows:

$$F1_{\text{macro}} = \frac{1}{A} \sum_{a} F1_a \tag{1}$$

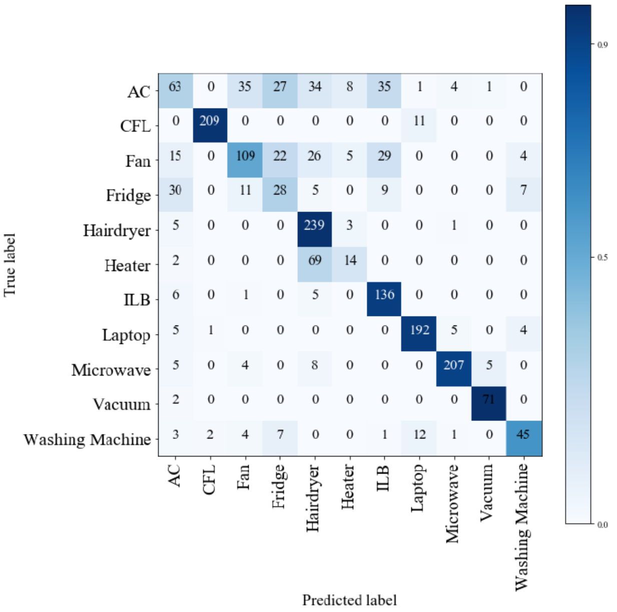
$$F1_a = \frac{\text{TP}_a}{2\ \text{TP}_a + \text{FP}_a + \text{FN}_a} \tag{2}$$

$$\tag{3}$$

where $F1_a$, TP_a, FP_a, and FN_a are the $F1$-measure, true positives, false positives and false negatives for the results of the classifier classifying appliance type a. The $F1_a$-measure for a perfect classifier is 1, whereas a random classifier yields an $F1_a$-measure of 0.25. This measure provides information about the confusion between instances. The magnitude is mainly determined by the number of instances that are correctly labeled as appliance type a, and says nothing about the instances that are correctly labelled as not being appliance type a (the true negatives).

TABLE IV
THE GAIN/LOSS IN THE $F1_{\text{MACRO}}$-MEASURE (IN %) WHEN HANDLING DATA IMBALANCE COMPARED TO THE STANDARD CLASSIFIER.

method	standard	over	under	smote	BB	weighted
kNN	80.60	+0.04	−2.07	+0.15	+10.35	+0.00
GNB	78.47	+0.09	−0.79	+0.06	+4.45	+0.00
LGC	89.47	−0.07	−0.46	−0.66	+1.47	−0.06
LDA	56.36	−3.1	+2.12	+5.77	+34.00	+0.00
dTree	78.86	+0.22	−1.25	+0.35	+11.60	+0.47
rForest	91.72	+0.14	−1.18	+0.00	+1.02	+0.09
SVM	94.55	−0.62	−0.31	−0.41	−1.03	−0.01
adaBoost	73.39	+0.56	−1.13	+0.12	+15.63	+0.00



	AC	CFL	Fan	Fridge	Hairdryer	Heater	ILB	Laptop	Microwave	Vacuum	Washing Machine
AC	63	0	35	27	34	8	35	1	4	1	0
CFL	0	209	0	0	0	0	0	11	0	0	0
Fan	15	0	109	22	26	5	29	0	0	0	4
Fridge	30	0	11	28	5	0	9	0	0	0	7
Hairdryer	5	0	0	0	239	3	0	0	1	0	0
Heater	2	0	0	0	69	14	0	0	0	0	0
ILB	6	0	1	0	5	0	136	0	0	0	0
Laptop	5	1	0	0	0	0	0	192	5	0	4
Microwave	5	0	4	0	8	0	0	0	207	5	0
Vacuum	2	0	0	0	0	0	0	0	0	71	0
Washing Machine	3	2	4	7	0	0	1	12	1	0	45

Fig. 2. The confusion matrix constructed when SVM is applied on PLAID 1 + 2. The numbers in the matrix are the absolute values, the colors represent the value relative to the total number of appliance instances per appliance type.

Table IV shows the relative gain/loss for the different methods when comparing them to the result when the imbalance is not counteracted (the standard approach). Three things can be noted from the results. First, it can be noted that applying over-, and under-sampling, smote or adapting the error function, does not lead to an improved $F1_{\text{macro}}$-measure when applied on PLAID 1 + 2 if the binary VI-image is used as input for the previous mentioned classifiers. Second, the improvement is significant for balanced bootstrapping when used for kNN, LDA, dTree and adaBoost. However, in these cases, the standard performance is well below the $F1_{\text{macro}}$ result of a SVM ($F1_{\text{macro}}$ = 94.55%). Even when handling data imbalance, none of these classifiers outperform the SVM method when the binary VI-image is used as input. Third, when considering the best classifiers rForest and SVM, none of these methods lead to a significant improvement when the binary VI-image is used as input. This shows that rForest and SVM are quite robust in learning the appliances types, even when the data is imbalanced. The confusion matrix constructed from the results when SVM is applied on the binary VI-images of PLAID 1 + 2 is shown in Figure 2. The values in the matrix represent the absolute number of appliance instances detected. The color represents per appliance type the relative amount with respect to the total number of that appliance type. The air conditioner (AC), fan and fridge get confused with each other, as well as the heater and the hairdryer. The cause is that the confused appliances contain similar electrical components: the AC and fridge are mostly compressors, and both the heater and hairdryer consist of a heating element.

From the results in Table IV, one can also conclude that the dataset contains redundancy and more measurements were

TABLE V
THE GAIN/LOSS IN THE $F1_{\text{MACRO}}$-MEASURE (IN %) WHEN HANDLING DATA IMBALANCE COMPARED TO THE STANDARD CLASSIFIER USING LESS TRAINING DATA.

method	standard	over	under	smote	BB	weighted
kNN	81.47	$+1.29$	-4.36	$+1.60$	$+10.85$	$+0.00$
GNB	84.31	$+0.03$	-6.94	$+0.03$	$+4.85$	$+0.00$
LGC	88.40	$+0.02$	-2.33	-0.16	$+4.17$	$+0.09$
LDA	54.24	-0.69	$+0.48$	-0.65	$+38.81$	$+0.00$
dTree	80.36	$+0.94$	-5.35	$+2.03$	$+11.48$	$+0.06$
rForest	93.99	$+0.04$	-3.66	-0.29	-0.68	-0.15
SVM	95.57	-0.75	-2.69	-1.09	-1.74	-0.18
adaBoost	78.85	$+0.88$	-4.09	-1.09	$+13.87$	$+0.00$

performed than necessary as synthesizing and over-sampling do not offer a performance increase when the VI-binary image is used as input. To reinforce this statement, the experiments are repeated in the same manner but each training set is reduced such that in each house, each appliance is only measured once. It is important to note that the test set remained the same (so some data remained unused). Based on the results in Table V, similar conclusions can be made as above, and the standard $F1_{\text{macro}}$-measure for each classifier is about the same as when trained with more data. This confirms the statement that for appliance identification, more measurements than necessary are present in the PLAID 1 + 2 dataset when the VI-binary image is used as input.

IV. Conclusion

In this paper, an extension of the PLAID dataset was presented (PLAID 2), where the number of appliance instances is increased with 719 to a new total of 1793. Where possible, the appliances were measured while operating in different modes. An exploration of the data reveals that the different operating modes of an appliance do not change the appliance signature in steady-state significantly. This will be a topic of further investigation.

Furthermore, it was shown that applying methods handling imbalance on PLAID 1 + 2 like over-, and under-sampling, synthesizing samples, balanced bootstrapping and adjusting the error function do not lead to any improvements in terms of the $F1_{\text{macro}}$-measure if the right classifier is used and when the binary VI-image is used as input. If a sub-optimal choice of classifier is made, balanced bootstrapping can increase the performance. The results also indicate that for appliance identification purposes, more measurements than necessary are present in the PLAID 1 + 2 dataset. This was confirmed by the fact that the results when training with less data (only one measurement per appliance in a house), are comparable when the binary VI-image is used as input.

References

[1] J. Gao, S. Giri, E. C. Kara, and M. Bergés, “PLAID: a public dataset of high-resoultion electrical appliance measurements for load identification research: demo abstract,” in *Proc. ACM Conference on Embedded Systems for Energy-Efficient Buildings*. ACM, 2014, pp. 198–199.

[2] M. Kahl, A. U. Haq, T. Kriechbaumer, and H.-A. Jacobsen, “Whited-a worldwide household and industry transient energy data set,” in *Proc. 3rd International Workshop on Non-Intrusive Load Monitoring*, 2016.

[3] T. Picon, M. N. Meziane, P. Ravier, G. Lamarque, C. Novello, J.-C. L. Bunetel, and Y. Raingeaud, “COOLL: Controlled On/Off Loads Library, a Public Dataset of High-Sampled Electrical Signals for Appliance Identification,” *arXiv preprint arXiv:1611.05803*, 2016.

[4] M. Mayo and S. Omranian, “Towards a new evolutionary subsampling technique for heuristic optimisation of load disaggregators,” in *Proc. Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2016, pp. 3–14.

[5] K. Basu, V. Debusschere, A. Douzal-Chouakria, and S. Bacha, “Time series distance-based methods for non-intrusive load monitoring in residential buildings,” *Energy and Buildings*, vol. 96, pp. 109–117, 2015.

[6] D. Chen, D. Irwin, and P. Shenoy, “Smartsim: A device-accurate smart home simulator for energy analytics,” in *Smart Grid Communications (SmartGridComm), 2016 IEEE International Conference on*. IEEE, 2016, pp. 686–692.

[7] N. Buneeva and A. Reinhardt, “Ambal: Realistic load signature generation for load disaggregation performance evaluation,” in *Smart Grid Communications (SmartGridComm), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1–9.

[8] G. Levi and T. Hassner, “Age and gender classification using convolutional neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 34–42.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: synthetic minority over-sampling technique,” *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[10] M. Tahir, J. Kittler, K. Mikolajczyk, and F. Yan, “A multiple expert approach to the class imbalance problem using inverse random under sampling,” *Multiple Classifier Systems*, pp. 82–91, 2009.

[11] C. Drummond, R. C. Holte *et al.*, “C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling,” in *Workshop on learning from imbalanced datasets II*, vol. 11. Citeseer Washington DC, 2003.

[12] M. Buda, A. Maki, and M. A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” *arXiv preprint arXiv:1710.05381*, 2017.

[13] B. C. Wallace, K. Small, C. E. Brodley, and T. A. Trikalinos, “Class imbalance, redux,” in *Proc. 11th International Conference on Data Mining (ICDM)*. IEEE, 2011, pp. 754–763.

[14] C. Elkan, “The foundations of cost-sensitive learning,” in *International joint conference on artificial intelligence*, vol. 17, no. 1. Lawrence Erlbaum Associates Ltd, 2001, pp. 973–978.

[15] J. Gao, E. C. Kara, S. Giri, and M. Bergés, “A feasibility study of automated plug-load identification from high-frequency measurements,” in *Proc. Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2015, pp. 220–224.

[16] N. Sadeghianpourhamami, J. Ruyssinck, D. Deschrijver, T. Dhaene, and C. Develder, “Comprehensive feature selection for appliance classification in NILM,” *Energy and Buildings*, 2017.

[17] N. Chawla, N. Japkowicz, and A. Kotcz, “Editorial: special issue on learning from imbalanced data sets, SIGKDD explor. newsl. 6 (1)(2004) 1–6.”

[18] G. King and L. Zeng, “Logistic regression in rare events data,” *Political analysis*, vol. 9, no. 2, pp. 137–163, 2001.

[19] S. Makonin and F. Popowich, “Nonintrusive load monitoring (NILM) performance evaluation,” *Energy Efficiency*, vol. 8, no. 4, pp. 809–814, 2015.