Classifying Power Quality Disturbances in Noisy Conditions using Machine Learning

Bojana Velichkovska
Faculty of Electrical Engineering and Information Technologies
Ss. Cyril and Methodius University
1000 Skopje, N. Macedonia
bojanav@feit.ukim.edu.mk

Marija Markovska
Faculty of Electrical Engineering and Information Technologies
Ss. Cyril and Methodius University
1000 Skopje, N. Macedonia
marijam@feit.ukim.edu.mk

Hristijan Gjoreski
Faculty of Electrical Engineering and Information Technologies
Ss. Cyril and Methodius University
1000 Skopje, N. Macedonia
hristijang@feit.ukim.edu.mk

Dimitar Tashkovski
Faculty of Electrical Engineering and Information Technologies
Ss. Cyril and Methodius University
1000 Skopje, N. Macedonia
dtaskov@feit.ukim.edu.mk

ABSTRACT

When ensuring high quality power supply of the power grid it is of the upmost importance to correctly detect and classify any power quality (PQ) disturbance. The right features can aid in speeding the process of classifying an instance, as well as increase classification accuracy. Therefore, we analyze the signals and extract information from them, and then select the most significant features. Additionally, an effective classification model is required. Therefore, we use the obtained features in implementing a grid search algorithm, in order to determine the most effective method for classification of power quality disturbances.

KEYWORDS

Power quality, feature extraction, classification, machine learning

1 INTRODUCTION

As nowadays the power grid undergoes many disturbances, it has become more of a challenge to not only detect these disturbances, but also to correctly identify them (as harmonics, transients, voltage dips, etc.). Awareness regarding a disruption within a signal could be used to minimize the undesired effects which poor power quality could inflict. The information gathered during detection and classification of power grid malfunctions could be utilized as insight in solving them. Additionally, the reliability of a power distribution network is key in preventing complaints by customers, which is why there is an increase in the accent placed on detection of disturbances in the grid.

With regard to the feature extraction methods used to formulate the data frame before applying an algorithm, S transform is one of the more popular feature extraction approaches [1]. Other popular feature extraction tools are Wavelet Transform and Discrete Wavelet Transform [2] [3] [4] [5].

Proposed methods for classifications of PQ disturbances include rule-based approaches, as the one presented in [1], time series analysis [5], artificial neural networks (ANNs) [6] and machine learning techniques as Support Vector Machines, Decision Trees and Random Forrest [7] [8].

Upon combining different feature extraction methods with machine learning approaches unique investigations of the problem are provided. We utilize the results of a feature extraction and feature selection design to create descriptive feature groups and to use the obtained groups for training various models. The feature extraction process carries significant weight in addressing the type of disturbance inhibiting the power signal. A correctly determined set of features can be crucial for the accuracy of a model, as on a weak set of features even the best approach would provide extinguishable results. Moreover, a compressed feature set can influence the required processing time.

The processing of the signals and the feature extraction and selection method are addressed in section II, the approach methods are discussed in section III, the results are presented in section IV and we conclude this research in section V.

2 DATA AND FEATURE EXTRACTION

Our dataset is comprised of samples from 21 different type of PQ disturbances, as given in Table I, generated in accordance with the mathematical definitions given in [9]. We used those samples as PQ signals accompanied with 20 dB, 30 dB, 40 dB and 50 dB white Gaussian noise. Every signal we examined contained 10 cycles, for fundamental frequency of 50 Hz and sampling frequency of 3.2 kHz. Accordingly, a final instance in our dataset contained 640 samples linked in a time series. Upon these instances we preformed automatic extraction of relevant features with the help of tsfresh (Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests) [10]. The tsfresh library is used to accelerate the process of extracting features by combining 63 time series characterization methods, which resulted in 794 time series features,

Table 1 PQ Classes

PQ Disturbance	Classes
Pure	C1
Sag	C2
Swell	C3
Interruption	C4
Transient/Impulsive/Spike	C5
Oscillatory transient	C6
Harmonics	C7
Harmonics + Sag	C8
Harmonics + Swell	C9
Flicker	C10
Flicker + Sag	C11
Flicker + Swell	C12
Sag + Oscillatory transient	C13
Swell + Oscillatory transient	C14
Notch	C15
Harmonics + Sag + Flicker	C16
Harmonics + Swell + Flicker	C17
Harmonics + Sag + Oscillatory transient	C18
Harmonics + Swell + Oscillatory transient	C19
Harmonics + Sag + Flicker + Oscillatory transient	C20
Harmonics + Swell + Flicker + Oscillatory transient	C21

The obtained features can be applied for classification. We used the complete set of extracted features that tsfresh provides, to train our models. However, as the number of obtained features was significant, it was necessary to influence the computational efficiency of the models. To do so, we used a feature selection module designed to reduce the number of features by selecting the most relevant descriptors of the problem.

Firstly, we determined the mutual information between each extracted feature and the corresponding class. Considering the fact that the higher the value of the mutual information the more relevant the feature is to the class, we sorted the features in descending order based on this value. We selected the upmost 300 features. Next, we divided the features in groups of 50. For the first group, we calculated the Pearson correlation coefficient [11] for every pair of features. Upon encountering a pair with a correlation higher than 0.8, the feature with the lower mutual information to the class was removed. Once the whole group was iterated, the next group was appended to the remainder of the features and the process was repeated for all groups, until a final set of features was obtained.

These features were a mix of both time domain features and frequency domain features. We inputted the time domain and frequency domain features to our models separately, as well as a union between the sets upon training our models.

3 MACHINE LEARNING APPROACH

With the four feature-wise possible data frames (all initial features, remainder of features after the feature selection process only containing time domain features, only containing frequency domain features and a combination of the two) we trained our models with 1000 signals per class. The signals used for training and testing within one class had different phase shifts. Our test set also contained 1000 signals per class.

We trained our models with different signal sets, where each set was synthetized with 20 dB, 30 dB, 40 dB or 50 dB of white Gaussian noise. Also, a set of signals synthesized with 20~50 dB of noise was used. The latter is due to the noise variations occurring within a real signal.

The following five algorithms we used for the experimental analysis.

- *K Nearest Neighbors (KNN)* is algorithm that analyzes a test sample in comparison to the whole of the train data. Through comparing the sample with the whole of the data frame, the closest neighbors which have the highest similarity with the sample are found. They determine the class in which the sample will be placed.
- *Decision Tree* is an algorithm which preforms data division by splitting the frame into several branches recursively, in a way that each split is determined by a value of one feature from the data frame. The branching ends when a class for the analyzed sample is reached.
- *Random Forest* is an ensemble algorithm of decision trees. Each tree performs on the values of a random feature vector sampled independently and with the same distribution for all trees in the forest.
- *Gradient Boosting* produces a prediction model in a stage-wise fashion as an ensemble of weak prediction models, typically decision trees.
- *XGBoost* is an implementation of gradient boosted decision trees which are designed for speed and performance.

4 RESULTS

Through the results given in Tables 2-6, we can observe the behavior of our algorithms depending on the noise and the feature set selected.

We concluded that using all the features tsfresh extracts to train our models provided the highest accuracy. After we extracted the features with the method described in Section 2 and trained our models with the extracted features, a significant decrease in the accuracy of the models occurs. Those remaining features we divided into two categories, time domain and frequency domain features, before we retrained our algorithms. The time domain features in the worst-case scenario provided insignificant decrease in the accuracy of the classifiers, whereas the accuracy obtained through the frequency domain features caused a severe drop in accuracy.

When analyzing the algorithms themselves it shows that the XGBoost algorithm performed best on all counts, with Gradient Boosting a close second.

Table 2. Comparison of algorithms performance with 20 dB noise.

20 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.99	0.43	0.46	0.31
Decision Tree	0.29	0.6	0.54	0.37
Random Forrest	0.43	0.42	0.53	0.26
Gradient Boosting	0.99	0.75	0.75	0.53
XGBoost	0.98	0.78	0.77	0.54

Table 3. Comparison of algorithms performance with 30 dB noise.

30 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.99	0.46	0.52	0.34
Decision Tree	0.17	0.68	0.63	0.43
Random Forrest	0.48	0.52	0.6	0.25
Gradient Boosting	0.98	0.85	0.85	0.55
XGBoost	0.98	0.88	0.87	0.57

Table 4. Comparison of algorithms performance with 40 dB noise.

40 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.99	0.54	0.59	0.43
Decision Tree	0.29	0.68	0.65	0.5
Random Forrest	0.43	0.57	0.6	0.39
Gradient Boosting	0.99	0.87	0.87	0.63
XGBoost	0.99	0.9	0.89	0.65

Table 5. Comparison of algorithms performance with 50 dB noise.

50 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.99	0.57	0.63	0.45
Decision Tree	0.29	0.63	0.65	0.52
Random Forrest	0.51	0.58	0.69	0.44
Gradient Boosting	0.98	0.87	0.87	0.64
XGBoost	0.99	0.9	0.89	0.7

Table 6. Comparison of algorithms performance with 20-50 dB noise.

20~50 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.98	0.51	0.57	0.4
Decision Tree	0.42	0.64	0.6	0.43
Random Forrest	0.45	0.54	0.59	0.3

Gradient Boosting	0.99	0.85	0.84	0.6
XGBoost	0.99	0.88	0.87	0.6

5 CONCLUSION

With this paper we addressed the effectiveness of observing the PQ signals as time series and using features obtained with tsfresh in classification of PQ disturbances. The testing was conducted with numerous classification models, each time on 21 classes and accompanied with different noise levels.

The results show that features extracted with tsfresh can be used for correctly classifying PQ disturbances. However, a deeper understanding in choosing subsections of those features is needed, and so even additional testing.

Our future endeavors will be creating a more optimal feature set, which might consist of a combination of our current features and features obtained from Wavelet Transform. Additionally, we will investigate the possibility of improvements which deep learning methods would bring to our results.

ACKNOWLEDGMENTS

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

[1] Rodríguez, A & Aguado, Jose & Martín, F & Lopez, J.J. & Muñoz, F & Ruiz, J.E.. (2012). Rule-based classification of power quality disturbances using S-transform. Electric Power Systems Research. 86. 113–121. 10.1016/j.epsr.2011.12.009.

[2] I. Daubechies, Ten lectures on wavelets, Philadelphia, PA:Society for Industrial and Applied Mathematics, (1992).

[3] C.S. Burrus, R.A. gopinath, and H.Guo, Introduction to Wavelets and Wavelet Transforms: A Primer (Englewood Cliffs, NJ: Prentice-Hall, 1998).

[4] Subhamita Roy, and Sudipta Nath, Classification of power quality disturbances using features of signals, International Journal of Scientific and Research Publications, Vol. 2, No. 11, November 2012,01-09.

[5] Lalit Kumar Behera, Maya Nayak, and Sareeta Mohanty, Discrete wavelet transform and S-transform based time series data mining using multilayer perception neural network, International Journal of Engineering Science and Technology,Vol. 3 No. 11, November 2011, 8039-8046.

[6] Monedero, Iigo & León, Carlos & Ropero, Jorge & García, Antonio & Manuel Elena, Jos & Montaño, Juan-Carlos. (2007). Classification of Electrical Disturbances in Real Time Using Neural Networks. Power Delivery, IEEE Transactions on. 22. 1288 - 1296. 10.1109/TPWRD.2007.899522.

[7] Milchevski, A.; Taskovski, D. Classification of power quality disturbances using wavelet transform and SVM decision tree. In proceedings of the 11th IEEE Electrical Power Quality and Utilization Conference, Lisbon, Portugal, 17–19 October 2011.

[8] Markovska, Marija & Taskovski, Dimitar. (2018). The effectiveness of wavelet based features on power quality disturbances classification in noisy environment. 10.1109/ICHQP.2018.8378873.

[9] Markovska, M.; Taskovski, D. On the choice of wavelet based features in power quality disturbances classification. In Proceedings of the 2017 IEEE 17th International Conference on Environment and Electrical Engineering (EEEIC), Milan, Italy, 6–9 June 2017.

[10] M. Christ, A.W. Kempa-Liehr, M. Feindt, Distributed and parallel time series feature extraction for industrial big data applications. Asian Machine Learning Conference (ACML) 2016, Workshop on Learning on Big Data (WLBD), Hamilton (New Zealand), ArXiv preprint arXiv: 1610.07717v1.

[11] F. Galton, Co-relations and their measurement, chiefly from anthropometric data. Proceedings of the Royal Society of London 45 (1888):135–145.Conference Name:ACM Woodstock conference