Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 54-7 (2021) 262–267

DEEP LEARNING FOR FAULT DETECTION IN TRANSFORMERS USING VIBRATION DATA

V. Rucconi*, L. De Maria**, S. Garatti*, D. Bartalesi**, B. Valecillos***, S. Bittanti*

** Dipartimento di Elettronica, Informazione e Bioingegneria - Politecnico di Milano, piazza L. da Vinci 32, 20133, Milan, Italy (valerio.rucconi@mail.polimi.it, {simone.garatti,sergio.bittanti}@polimi.it)*
***RSE S.p.A, via Rubattino, 20134, Milan, Italy (Letizia.demaria@rse-web.it, Daniele.Bartalesi@rse-web.it)*
****Trafoexpert GmbH, 8610 Uster, Switzerland (b.valecillos@trafoexperts.com)*

Abstract: The purpose of this paper is to evaluate the virtue of deep neural networks in detecting incipient failures of transformers, in particular windings looseness, via vibration data analysis. The transformer vibration technique is a non-invasive method to monitor winding looseness. It is based on the analysis of vibration spectra measured by sensors located on the transformer tank. In this paper, we rely on measurements that have been made in a dedicated lab under two different conditions: in presence or in absence of the clamping pressure on the windings. The analysis of data, oriented to fault detection, is performed by feedforward neural networks which, by experimental results, proved effective for a reliable prediction. Special emphasis is given to the robustness of the prediction to sensor misplacement and various techniques are carried out to evaluate and to enforce generalization to out-of-sample-data for the obtained classifier.

Keywords: power transformers, winding fault detection, machine learning, feedforward neural networks, regularization.

1. INTRODUCTION

The transformer windings are susceptible to various kinds of faults and defects such as deformation, shifts and loss of clamping pressure. Duan et al., (2018) observed that more than 50 % of the transformer's faults are caused by winding issues. These faults and deformations on windings can lead to an insulation breakdown and sudden transformer failure. Therefore, the diagnosis of winding looseness defects is of great significance for an early detection of transformer faults and ensuring grid safety. The vibration analysis method is an effective online and non-invasive method for monitoring the winding state based on the transformer vibration signal. This diagnostic approach is based on the acquisition of vibration spectra from the transformer utilizing sensors, like accelerometers, mounted on the tank of the transformer.

In this paper, the usage of the neural network classification technique will be investigated on the vibration spectra of the tank both for tight and loose clamping of its windings pack. Our study relies on experimental data obtained from laboratory experiments.

The paper unfolds as follows. The experimental setup is briefly recalled in Section 2, while Section 3 discusses the neural network implementation details for the problem at hand. The core of the paper is Section 4, which introduces the problem of robustness against sensor misplacements, discusses the procedure that have been adopted to enhance and test the robustness of the obtained classifiers and reports all the results in the various classification trials. Eventually, some conclusions are drawn in Section 5.

2. DATA ACQUISITION

The experimental setup is the same of Tavakoli et al., 2019. Here, the most salient facts are reported, while the reader is referred to Tavakoli et al., 2019 for more details.

A new oil-filled three-phase transformer (42kV/580V, 750kVA, ONAN cooling) was used as test transformer for collecting vibrations. It has been tested under tight (corresponding to no-fault) and loose (fault) clamping, which was replicated by loosening the three phases of the transformer simultaneously at the top of the structure. The available measurements considered in this paper have been performed in a *load* condition, where the transformer was energized from the High Voltage (HV) side to nominal current (10A) with shorted secondary winding. In this case, the winding vibrations are dominant due to the low flux in the transformer core. (Al-Abadi et al., 2017).

A grid of 24 measurement points was defined on the Low Voltage (LV) side of the tank. The measurements have been obtained from each sensor in the grid while the transformer was operated under both tight and loose clamping. Our goal is then to distinguish these conditions, and therefore detecting possible incipient faults, by analysing the vibration data.

The vibration sensing system used for the tests is based on optical MEMS accelerometers, an Electro-Optical Unit (EOU), and a conditioning and recording Unit. One 2-minutes timeseries was recorded for each position on the transformer tank both for tight and loose windings, sampled with a frequency of 44 kHz. Each recorded signal was then subdivided into 100 subseries, averaged, filtered and Fast

10.1016/j.ifacol.2021.08.369

Fourier Transformed into the frequency domain. Only harmonics up to 500Hz were considered as these carry the relevant information on windings vibration (Tavakoli et al., 2019). No data normalization was used in this work. The final dataset consists of pairs of the spectrum, represented as a vector of real numbers since only the magnitudes are considered, provided with the corresponding label which can be either “tight” or “loose”. Moreover, the dataset can be decomposed into 24 balanced sub-datasets, one for each position of the grid. These positions are indicated by progressive numbers 41, 42, …, 63, 64.

3. FAULT DETECTION VIA NEURAL NETWORKS

To the aim of classifying tight and loose transformer conditions, we resort to deep neural networks (networks with two or more layers), which are the perfect tools to identify faulty patterns and classify them, given their innate capability to approximate non-linear complex functions (Goebel et al., 2008). In this section, a brief description of the adopted setup is provided to highlight those aspects specific for the problem at hand. The reader is referred e.g. to Goodfellow et al. (2016) for a more comprehensive description.

3.1 Feedforward neural network

We resort to feedforward neural networks, which are simple but well-suited for the problem at hand. In order to make the network learn complex non-linear patterns present in the data, a key ingredient is the choice of the activation functions (Nwankpa et al., 2018). Two of the most commonly used activation functions are the so-called Tanh (Hyperbolic Tangent) and the ReLU=max(0,x) activation functions. The issue using Tanh is the vanishing gradient problem: the backpropagated gradient used to update the weights and biases during training will tend to vanish in each layer preventing some parameters to be trained. ReLU does not suffer from the vanishing gradient but it is more prone to overfit (Nwankpa et al., 2018). Since in this paper we consider neural nets with two layers only, this problem is not so significant; hence we resorted to the Tanh activation function. The suggested weight initialization using Tanh is the Xavier (or Glorot) initialization (Kumar, 2017).

3.2 Optimization algorithm

The training of the neural network is achieved by minimizing a suitable cost function, which is taken here as the Cross Entropy of the prediction error (cross entropy is best suited for classification problem). The minimization is performed by a numerical optimization algorithm among the class of second-order optimization algorithms, namely methods that, besides the gradient, use also the information conveyed by the Hessian matrix about the curvature of the objective function. Specifically, two options have been considered:

- Conjugate Gradient Descent. This is a family of second-order methods that, instead of calculating the inverse of the Hessian matrix, follows the conjugate direction, i.e. a specific direction that does not cancel out the progress made in the previous directions (Battiti, 1992). These methods exploit the fact that a sequence of conjugate search directions can be generated without a direct reference to the Hessian matrix.

- Quasi-Newton Methods. These methods utilize an approximation of the inverse Hessian matrix to retrieve the optimal training directions. The Broyden–Fletcher–Goldfarb–Shanno (BFGS) is the selected Quasi-Newton method, that handles only the first derivatives of the cost function.

3.3 Model generalization and overfitting

Overfitting in machine learning refers to a model that fits so well the training data that it lacks the ability to generalize to out-of-sample data. Neural networks trained on small datasets are prone to overfit. Since relatively small training sets (150-2000 data) are used in this paper, overfitting is a concrete problem. To address this issue, two methods are considered: a technique to reduce the chance and effect of overfitting (regularization) and a procedure to detect when overfitting is undergoing to prevent overtraining (early stopping).

3.3.1 Regularization

A neural network can overfit a training dataset because it has sufficient capacity to do so. A common approach, see e.g. Brownlee (2018), is to constrain the complexity of the model by ensuring that the weights of the neural network remain small. Supplementary techniques that aim to reduce overfitting by keeping network weights small are referred to as regularization methods. As Nielsen (2018) reported, the smallness of the weights makes it difficult for a regularized network to learn the effects of local noise in the data and it is therefore forced to build a relatively simple model based on patterns often seen across the training data.

The regularization technique implemented in this paper is the L2 regularization, implemented by adding to the cost function the sum of the squares of all the weights of the network scaled by a factor $\lambda/2n$, where $\lambda>0$ is the regularization parameter and n is the size of the training set. The effect of L2 regularization is well explained by Nielsen (2018): if λ is small, the minimization of the original cost function is prioritized, while if λ is large, the regularization effect makes the network prefer to learn small weights. The regularization parameter λ ranges from 0 to 1: value 0 corresponds to a model that is allowed to overfit the training data, while value 1 represents a model that underestimates the weights and so underfit the problem.

3.3.2 Early-stopping

An easy way to detect overfitting during training is to keep track of the accuracy of the validation dataset as the network trains. At the end of each epoch, the classification error on the validation data is computed using the weights resulting from the training. When the neural network begins to overfit the data, the validation error begins to rise and if it increases for a specified number of iterations (value used = 6), the training process ends so to avoid further overtraining of the data and the weights and biases of the neural network corresponding to the minimum validation error are returned. This procedure is known as early stopping and, according to Caurana at al. (2001), combined with backpropagation is so effective that large neural networks can be trained without significant overfitting since it occurs that at early stages they learn models similar to the ones learned by smaller networks.

3.4 Choice of the hyperparameters

The main hyperparameters that have to be selected in our problem through validation are:

- Number of neurons per layer. We considered only networks with an equal number of neurons in each layer so to have only one hyperparameter to tune without any huge drawback.

- Regularization coefficient.

The selection of best hyperparameter's configuration is obtained by means of a grid search, which exhaustively evaluates various configurations from a grid of values for the hyperparameters. The evaluation of the performance of a given configuration is obtained through K-fold cross-validation and more precisely to J-K-fold cross-validation (Moss et al., 2018) for the reasons given in the next section.

Besides the tuning of the mentioned hyperparameters, cross-validation is also exploited to determine all other possible choice that has to be made during the training phase, like which optimization algorithm and which activation function have to be used, whether or not to implement the ensemble of neural networks (see next Section 3.5), and so on.

Notice that, as for the number of layers, it emerged during some preliminary validation experiment that it was impossible to correctly classify the signals utilizing just one layer (shallow neural network). Instead, by using two layers, all the classification tasks were correctly carried out. For this reason, this hyperparameter has been fixed since the beginning to two hidden layers. Note also that the maximum number of training epochs is not an hyperparameter to be tuned in the present context since early stopping of the training phase is adopted.

3.5 The variance problem

During the tuning of the neural network, a high variance in the performance of the networks was observed. With a high variability, it is impossible to decide whether a model is truly better than another one or if it is just a matter of random fluctuations.

3.5.1 J-K-Fold cross-validation

The random splitting of the data done by the K-fold cross-validation leads to variation in the prediction estimates. Moss et al. (2018) stated that if one model outperforms another by a small amount on a particular data-split there is no guarantee that it is truly superior. J-K-fold cross-validation provides a way to reduce the variance of a neural network by repeating J independent K-fold cross-validations and use the mean result across all runs to assess performance. This is the motivation for using J-K-fold cross-validation and, specifically, in this paper, J and K are chosen equal to 10 so to substantially reduce the variance.

3.5.2 Ensemble of neural networks

Another established technique used to reduce the variance of a classifier is to train multiple different models and then combine their predictions. This method is called ensemble learning and not only it reduces the variance of predictions, it also improves its predictive performance because the combination of the predictions from different models is generally more accurate than any of the individual models (Kadkhodaie et al., 2009). This is because different models usually do not make the same errors on the test set and the combination of these mistakes can be leveraged through model ensembling. In this paper, the ensemble is implemented by repeating the training of the same network with various optimization algorithms, so as to eventually obtain a final ensemble of models that have learned a diversified assortment of mapping functions and thus with low correlation in their predictions.

4. RESULTS ON EXPERIMENTAL DATA

The aim of this work is to obtain a neural network able to predict whether the windings of a transformer are tight or loose when fed with the vibration signals returned by one given sensor. One requirement of utter importance is that the classifier must be robust against the position on the tank where the sensor is located by the final user because misplacements of the sensor are possible. To this purpose, the training of the neural network must rely on the whole experimental data set of the present experiment, which incorporates information about various sensor positions. On the other hand, it is also clear that reproducing the experimental setup (consisting of 24 sensors) of this paper may be not affordable in practice, and hence our analysis is also meant to determine which is the least number of sensors that are needed for the training of the neural network to enforce the required robustness property when the neural network will be used with the vibration signal recorded from one random position on the tank.

Given these premises, the investigation unfolds in 4 conceptually consecutive sections. In Section 4.1, we first consider one sensor position at a time, and we train various classifiers, one for each sensor position, and analyse the reliability of each trained neural networks. As expected, the obtained classifiers do not show the sought robustness property being trained based on data corresponding to one single sensor position only. In Section 4.2, the potential of neural networks for a more comprehensive classifier will be assessed by incorporating in the training dataset multiple sub-datasets corresponding to all the sensor positions in the lateral parts of the transformer tank in a first experiment and all the positions in the central part in a second experiment. The so obtained classifiers show improved robustness against the position of the sensor from which test data are collected and in Section 4.3 we also inspect the robustness against unseen sensor positions. This is obtained by removing from the training dataset all data points corresponding to a given position and testing the obtained classifier on these data corresponding to the drawn-out position. These analyses reveal which sensors are more relevant to the purpose of obtaining a robust classifier and, eventually by leveraging these results, the least number of sensors needed to train a neural network that is capable to carry out a successful fault detection are discussed in Section 4.4.

The reliability of the neural networks obtained at each stage of our investigation is expressed by accuracy, sensitivity, and specificity, which are defined as follows. The accuracy represents the proportion of correct predictions (both true positive, i.e. loose condition correctly predicted, and true negative, i.e. tight condition correctly predicted) among all tests. Sensitivity evaluates how well the classifier is able to correctly classify a loose condition (true positive) among all positive ("loose" labelled) data. The specificity instead describes the ability to correctly predict tight conditions (true negative) above all negative ("tight" labelled) data.

4.1 *Training and testing classifier for each sensor position*

In this first step, we consider the data corresponding to one single sensor position at a time and train various neural networks, one for each sensor position. Specifically, the data collected by the considered sensor is divided randomly into a design dataset (90% of all data) and a test dataset (10% of the data). The design dataset is further split into a train dataset (90% of the design dataset) and a validation dataset (10% of the design dataset), which is used for weight estimation and for hyperparameter tuning and early stopping, respectively.

Irrespective of the chosen sensor position, after the validation phase, the selected configuration is a [10,10] feedforward neural network trained with the Conjugated Gradient Descent optimization algorithm CGB and 0.5 as regularization coefficient; in all cases, we obtain a 100% validation accuracy. See Figure 1 for a prototypical result of grid search validation for one given sensor position.

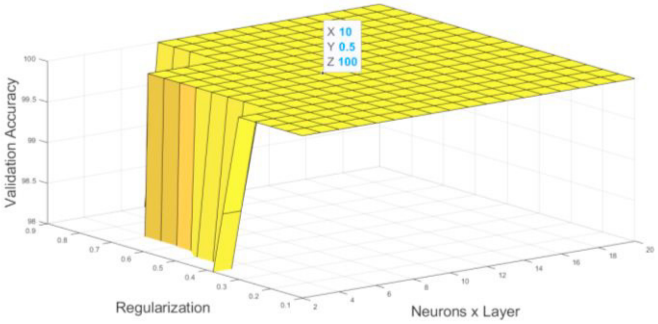

Fig. 1. Grid search validation accuracy for position 41.

For each sensor's location, the performance of the trained neural network is first assessed by classifying the unseen test data, referring to the same position of the design dataset. The accuracy of the neural network classifier for the corresponding holdout test data is always 100%, irrespective of the considered position. However, when the various classifiers obtained from one single sensor are tested against data corresponding to other sensor positions, the prediction performances drastically drop (this is true irrespective of the training data sensor position). For example, a network trained on position 61 labels correctly only 1 % of the signals measured on adjacent position 63, indirectly showing the changeability of the vibration modes with the position on the transformer tank wall. This also reveals, as expected, that classifiers constructed from data referring to one single sensor position are not robust against possible misplacements of the sensor in the test phase and motivates the subsequent analysis.

4.2 *Robustification of the classifier by extending the training dataset*

To address the problem highlighted in the previous Section 4.1, the learning process is enhanced by merging data referring to a wide range of sensor positions. The idea behind this extension is to figure out if a neural network classifier is able to correctly classify the different vibrational patterns that are coming from the various vibration signals obtainable from the various positions on the tank of the transformer. Therefore, in a first experiment, all the data coming from each sensor position of the lateral (left and right) parts of the transformer tank are gathered in a unique dataset. Likewise before, 90% of randomly chosen, data points, are then used as design dataset while the remaining 10% are used as test data to reveal the reliability of the trained network. The design dataset is in turn split into training data (90%) and validation data (10%).

The best configuration of hyperparameters obtained from validation is 100 neurons for each layer and 0.5 as the regularization coefficient. The accuracy, specificity, and sensitivity of this neural network over the holdout test dataset are respectively 99.83 %, 99.75 % and 99.92 %. The same approach has been carried out also for data referring to the central sensor positions. In this case, the neural network configuration obtained from the validation is 90 neurons per layer and regularization coefficient 0.5. In this instance, the accuracy, specificity, and sensitivity of this neural network over the holdout test dataset are respectively 99.90 %, 99.80 % and 100.00 %. Since these two neural networks exploit data coming from all the sampled positions of the transformer, their robustness is expected to be superior with respect to neural networks trained only on a single sensor measurement.

4.3 *Testing classifier over data from unseen positions*

We first consider the data referring to the lateral parts of the transformer tank. To test the robustness of the classifier obtained from measurements in various positions against data referring to a sensor position which is not available in the training phase, we proceed as follows. For each position in the lateral parts of the transformer, all data referring to that specific sensor are kept as holdout test data, while the data referring to all the other sensors of the lateral parts form the train and validation dataset. In this way, the unused position can be interpreted as totally unobserved vibration data in the design of the classifier and the reliability of the trained classifier indicates its robustness against this unobserved position. The procedure is repeated for every sensor position.

As for the training, there is a crucial observation to be made about the selection of one hyperparameter. As reported in Section 3.4.1, the regularization hyperparameter controls the overfitting of the model to the train dataset. In this stage, the model is trained on a training dataset that substantially differs from the test dataset, and therefore the goal is to enhance generalization as much as possible to correctly classify data coming from unexplored positions. For this reason, the tuning of the regularization coefficient is obtained by selecting the largest possible value provided that the validation accuracy is not impoverished too much. This latter requirement is fundamental since low validation accuracy associated to a high regularization coefficient is a sign of underfitting.

The configuration that has the most satisfying performance for all positions is a [60,60] net with 0.9 as regularization coefficient. The reliability of the trained neural network against the test dataset of the different excluded positions during the design phase is reported in Table 1.

Position	Accuracy	Sensitivity	Specificity
41	91.25 %	82.55 %	99.95 %
42	97.94 %	99.70 %	96.16 %
43	98.64 %	97.30 %	100.00 %
44	70.15 %	71.80 %	68.48 %
45	97.35 %	98.00 %	96.70 %
46	90.22 %	82.97 %	97.47 %
59	52.80 %	5.60 %	100.00 %
60	56.74 %	13.48 %	100.00 %
61	100.00 %	100.00 %	100.00 %
62	98.85 %	99.90 %	97.80 %
63	57.58 %	99.80 %	15.35 %
64	97.25 %	99.40 %	95.10 %

Table 1. Reliability over the lateral drawn-out positions

As it appears, the reliability of the classifiers for unseen sensor positions is high for most positions, showing that gathering together data from various sensor positions is indeed useful to enhance the robustness of the obtained classifier against data from unexplored positions. On the other hand, there are a few positions, i.e. positions 59, 60, and 63, which are completely unpredictable by a network obtained from data related to the other positions.

A similar experiment is also performed relatively to the dataset referring to the positions in the central part of the transformer. The results are reported in Table 2.

Position	Accuracy	Sensitivity	Specificity
48	91.63 %	88.30 %	94.95 %
49	80.27 %	97.23 %	63.13 %
50	47.26 %	93.28 %	1.70 %
51	49.90 %	2.10 %	98.18 %
52	32.26 %	34.75 %	29.80 %
53	62.50 %	53.10 %	71.90 %
55	50.00 %	99.90 %	0.10 %
56	64.70 %	94.00 %	35.10 %
57	70.60 %	43.33 %	97.60 %
58	50.75 %	1.50 %	100.00 %

Table 2. Reliability over the central drawn-out positions

As for the central part, there is just one position, position 48, that can be correctly predicted by a classifier trained from data taken from other positions. This fact indicates that classifiers obtained from data collected from the central part of the transformer are highly sensitive to possible sensor misplacements in the test phase and hence these data are not suitable to achieve the sought robustness property.

4.4 Classifier trained with the least number of sensors

In view of the conclusion of the previous section, we here focus on the data corresponding to the lateral parts of the transformer tank. The results in Table 1 reveal that some sensor measurements can be ignored during the training phase without harming the robustness of the resulting classifier. This suggests that, by using a reduced number of sensors, it would be possible to obtain a neural network able to correctly classify all the vibration data, even of unexplored positions. In particular, Table 1 suggests that positions 59, 60, and 63, must be included in the training dataset and the analysis that follows reveals that the least number of sensors that suffice to obtain a classifier that has high enough accuracy with test data taken in any other position from the lateral parts of the transformer is 6 and it consists of sensor position 42, 44, 46, 59, 60, and 63.

Specifically, the design (training and validation) dataset in this phase consists of the measurements from the 6 sensors mentioned above, while, to assess its robustness, the obtained network is then tested on all other data points corresponding to the other positions. Likewise, in the previous Section 4.3, given that in this phase the model is trained on a training dataset that substantially differs from the test dataset, the goal is to prioritize generalization. In this phase too we tune the regularization coefficient by taking the largest possible value that does not impoverish the validation accuracy too much.

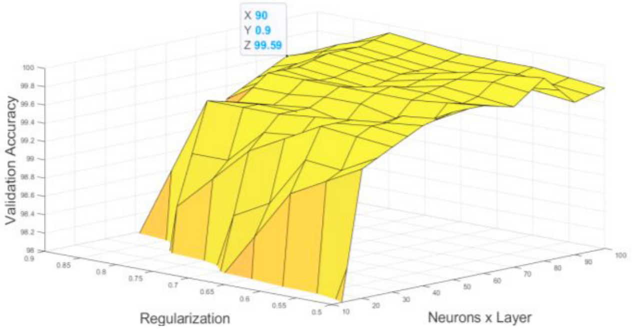

Fig. 2. Validation accuracy for the least number of sensors.

The grid search validation results are shown in Figure 2 and it turns out that the best configuration is a [90,90] net with regularization 0.9, trained based on the 'CGB' optimization algorithm. This configuration presents a large variance during the cross-validation phase though. This suggests to adopt an ensemble of neural networks to reduce the variance and also to improve the predictive capability.

The final ensemble model, as retrieved by validation, is given by the composition of the following five networks: [90,90], 0.9, Conjugate Gradient with Powell-Beale Restarts (CGB); [80,80], 0.9, Scaled Conjugate Gradient (SCG); [80,80], 0.9, Polak-Ribiére Conjugate Gradient (CGP); [60,60], 0.9, Fletcher-Reeves Conjugate Gradient (CGF) x2 (repeated since it is the best performing configuration).

After the tuning of the ensemble classifier, the data corresponding to positions other than 42, 44, 46, 59, 60, and 63 (and hence not used in training and validation) are used as test data to check the classifier robustness against sensor misplacement, i.e., to check whether the classifier is capable to predict the transformer winding condition from measurements taken in unexplored sensor positions. The results are reported in Table 3.

Position	Accuracy	Sensitivity	Specificity
41	94.01 %	88.03 %	99.99 %
43	95.46 %	90.97 %	100.00 %
45	97.15 %	94.83 %	99.47 %
61	99.96 %	99.92 %	100.00 %
62	99.38 %	100.00 %	98.76 %
64	94.35 %	99.99 %	88.71 %

Table 3. Reliability of the classifier trained with the least number of sensors

As it appears, the neural network obtained based on measurements from 6 sensor positions only is able to predict the transformer state with a quite high accuracy, robustly with respect to any possible misplacement in the positioning of the sensor at the lateral part of the transformer. Note that, since data are in short supply, no data from positions 42, 44, 46, 59, 60, and 63 have been left for testing. Still, from the results in Sections 4.1 and 4.2, it is possible to presume that if tested with vibration data taken from that positions, the network would have correctly predicted the corresponding transformer state with 100% accuracy.

5. CONCLUSIONS

In this paper we have studied neural nets classifiers as a tool for the detection of loss of clamping pressure in transformer windings. During each step, neural networks proved to be an excellent means to analyse the vibration spectra obtained by sensors placed in various parts of the transformer. Altogether, if applied to a consistent dataset (data for the train and test belongs to the same dataset), they proved to be very reliable in detecting faults. Moreover, they proved to be robust when applied to signals coming from positions unexplored during the training and validation process, provided that data from the lateral part of the transformer are used. Thus, except for some peculiar positions that have been detected by our analysis, the measured vibrations on the tank surface have a common pattern that allows for generalization to unexplored positions during training and validation. This eventually led to determining a minimum number of sensors that suffice to the development of a robust classifier, thus reducing the cost of the training phase for each transformer.

ACKNOWLEDGMENTS

This work has been financed by the Research Fund for the Italian Electrical System in compliance with the Decree of Minister of Economic Development; April 16 2018. The authors are also grateful to Fondazione Politecnico di Milano and to the National Research Council of Italy (CNR - Consiglio Nazionale delle Ricerche).

REFERENCES

Battiti, R. (1992). First- and Second-Order Methods for Learning: Between Steepest Descent and Newton's Method. In *Neural Computation*, vol.4, no.2, pp.141-166

Brownlee, J. (2018). *Better Deep Learning: Train Faster, Reduce Overfitting, and Make Better Predictions.* pp.245-255. Machine Learning Mastery, Melbourne, EN.

Caruana, R., Lawrence, S., and Giles, L. (2001). Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in Neural Information Processing Systems 13 - Proceedings of NIPS 2000.*

Duan, X., Zhao, T., Liu, J., Zhang, L., and Zou, L. (2018). Analysis of Winding Vibration Characteristics of Power Transformers Based on the Finite-Element Method. In *Energies 2018*, vol. 11, no. 9, pp. 2404-2423.

Al-Abadi, A., Gamil, A., Schatzl, F., Van Der Aa, B., De Groot, E., Declercq, J. (2017). Investigating the Effect of Winding Design and Clamping Pressure on the Load-Noise Generation of Power Transformers. In *CIGRE A2 Study Committee Intern. Colloquium*, Cracow.

Goebel, K., Saha, B., and Saxena, A. (2008). A comparison of three data-driven techniques for prognostics. In *Proceedings of 62nd meeting of the society for machinery failure prevention technology*, pp. 119-131.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*, MIT Press.

Kadkhodaie, A., Rezaee, R., and Rahimpour-Bonab, H. (2009). A committee neural network for prediction of normalized oil content from well log data: An example from South Pars Gas Field, Persian Gulf. In *Journal of Petroleum Science and Engineering,* vol. 65, pp. 23-32.

Kumar, S. K. (2017). On weight initialization in deep neural networks. *arXiv preprint arXiv:1704.08863.*

Moss, H., Leslie, D. S., and Rayson, P. E. (2018). Using J-K-fold Cross Validation to Reduce Variance When Tuning NLP Models. In *Proceedings of COLING 2018*.

Nielsen, M.A. (2015). *Neural Networks and Deep Learning (Vol. 2018)*. Determination press. San Francisco, CA.

Nwankpa, C., Ijomah, W., Gachagan, A., and Marshall, S. (2018). Activation Functions: Comparison of trends in Practice and Research for Deep Learning. *ArXiv 2018.*

Tavakoli, A., De Maria, L., Bartalesi, D., Garatti, S., Bittanti, S., Valecillos, B., and Piovan, U. (2019). Diagnosis of transformers based on vibration data. In *2019 IEEE 20th International Conference on Dielectric Liquids (ICDL).*