Fault Detection in a Wind Turbine Hydraulic Pitch System Using Deep Autoencoder Extracted Features

Panagiotis Korkos[1], Jaakko Kleemola[2], Matti Linjama[3], and Arto Lehtovaara[4]

[1,3,4]*Tampere University, Tampere, 33014, Finland*
{firstname.lastname}@tuni.fi

[2]*Suomen Hyötytuuli Oy, Pori, 28601, Finland*
Jaakko.Kleemola@hyotytuuli.fi

ABSTRACT

A wind turbine is equipped with lots of sensors whose measurements are recorded by the supervisory control and data acquisition (SCADA) system and stored every 10 minutes. The pitch subsystem of a wind turbine is of critical importance as it presents the highest failure rate. Thus, selecting the most essential features from the SCADA system is performed in order to detect faults efficiently. In this study, a feature space of 49 features is available, referring to the condition of a hydraulic pitch system. The dimensionality of this feature space (original input space) is reduced using a Deep Autoencoder in order to extract latent information. The architecture of the Autoencoder is investigated regarding its efficiency on fault detection task. This way, effect of new extracted features on the performance of the classifier is presented. A Support Vector Machine (SVM) classifier is trained using a set of healthy (fault free) and faulty data, representing different kind of pitch system failures. The data are acquired from a wind farm of five 2.3MW fixed-speed wind turbines. The performance metric used to evaluate their effect on data is F1-score. Results show that SVM using new extracted feature by Autoencoder outperforms SVM classifier using the original feature set, underlining the power of Autoencoders to unveil latent information.

1. INTRODUCTION

Wind energy is the fastest developing renewable energy in the world, and especially in Europe. Based on the annual report of WindEurope, in 2021 the total installed wind power capacity was 236 GW (WindEurope, 2022). Wind turbine costs are strongly associated with the profitability and wind energy share in the energy production in daily basis. In particular, the total generation cost of wind energy is between 4.5 and 8.7 €cent/kWh in case of onshore wind turbine, but the costs generated by Operation and Maintenance (O&M) is estimated to be 1-1.5 €cent/kWh (Blanco, 2009). Thus, O&M associated costs are very important and the only solution for consistent monitoring and maintenance is to accurately interpret the measurements. This interpretation is allowed through advanced data analysis techniques on the measurements of each wind turbine.

For that reason, each wind turbine is equipped with Supervisory Control and Data Acquisition (SCADA) system. SCADA system stores a plethora of measurements in a wind turbine ranging from environmental measurements to pressures and temperatures. Typically, measurements are stored in 10-min intervals even though they are sampled in higher frequency, e.g., 1 sec. Processing of SCADA signals has been a common strategy for a lot of windfarm operators, since it provides a cheap solution for wind turbine monitoring, avoiding the installation of more sensors.

A notable number of researchers have developed methodologies to process those SCADA signals for condition monitoring in wind turbines (Zaher, McArthur, Infield & Y. Patel, 2009; Chen, Zappala, Crabtree & Tavner, 2014; Tautz-Weinert & Watson, 2017; Yang, Court & Jiang, 2013). In addition, Stetco, Dinmohammadi, Zhao, Robu, Flynn, Barnes, Keane and Nenadic (2019) have summarized Machine Learning techniques that have been used in literature for wind turbine condition monitoring. Furthermore, more advanced techniques from the Deep Learning area have been the subject of the review in the study of Helbing and Ritter (2018), indicating the rise of Deep Learning for performing fault detection in wind turbines.

Regarding recent advancements in this application area, Convolutional Neural Network (CNN) have been widely used by researchers (Ulmer, Jarlskog, Pizza, Manninen & Goren Huber, 2020), as well as its variants such as convolutional neural network (CNN) and bidirectional gated recurrent unit (BiGRU) with attention mechanism (CNN-BiGRU-AM) (Xiang, Yang, Hu, Su & Wang, 2022), CNN with Long Short-Term Memory (LSTM) and attention mechanism (CNN-LSTM-AM) (Xiang, Wang, Yang, Hu & Su, 2021), convolutional neural networks (CNN) and gated recurrent unit (GRU) (Kong, Tang, Deng, Liu & Hana, 2020) and generative adversarial network (GAN) coupled with a temporal CNN (TCNN) (Afrasiabi, Afrasiabi, Parang, Mohammadi, Arefi & Rastegar, 2019). Additionally, different versions of autoencoders have been used like deep joint variational autoencoder (JVAE) for gearbox monitoring (Yang & Zhang, 2021), moving window stacked multilevel denoising AE (MW-SMDAE) (Chen, Li, Chen, Wang & Jiang, 2020) and sparse dictionary learning based adversarial variational auto-encoders (AVAE_SDL) (Liu, Teng, Wu, Wu, Liu & Ma, 2021). Finally, LSTM for gearbox monitoring has been applied by Qian, Tian, Kanfoud, Lee and Gan (2019).

The literature presented so far was mainly focused on the monitoring of the wind turbine, as a whole. The pitch system in wind turbines is very crucial for their operation since they present the highest failure rate and among the highest downtime according to several surveys (Wilkinson, Hendriks, Spinato, Harman, Gomez, Bulacio, Roca, Tavner, Feng, & Long, 2010; Carroll, McDonald, & McMillan, 2016; Ribrant, & Bertling, 2007). Therefore, it is considered the most critical subsystem and it needs to be monitored as effectively as possible. Pitch system monitoring has gained attention by several researchers. Chen, Matthews, and Tavner (2013, 2015) has implemented an a -priori adaptive neuro fuzzy inference system (APK-ANFIS) to monitor pitch system using as features five basic features (i.e., power output, wind speed, blade angle, rotor speed and motor torque for only the case of electric pitch system). Their study focused only on their average values. ANFIS has been used as well by Korkos, Linjama, Kleemola, and Lehtovaara (2022), investigating the effect of average and standard deviation values of the features mentioned in Chen et al. (2013, 2015). In addition, the novelty of their research was that their dataset contained a list of diverse pitch-system faults, referring to almost every kind of components. The same technique (ANFIS) was used in the studies of Schlechtingen, Santos, and Achiche (2013) and Schlechtingen and Santos (2014) in order to build normal behaviour models. Schlechtingen and Santos (2014) particularly used the model for hydraulic oil leakage, which is a common failure in the pitch system. Additionally, a pitch system fault, with no additional provided information, has been detected effectively using a multi-level-denoising autoencoder (MLD-AE) by Wu, Jiang, Wang, Xie, and Li (2019).

Apart from ANFIS, Support Vector Machines (SVM) have been used for fault detection. SVM classifiers have been developed by Leahy, Hu, Konstantakopoulos, Spanos, C.J., & and Agogino (2016, 2018), whereas Hu, Leahy, Konstantakopoulos, Auslander, Spanos, and Agogino (2017) trained SVM classifiers in an enhanced feature set according to domain knowledge. A variation of SVM, called asymmetric SVM, has been implemented by Wu, Su, Lu, and Rui (2015) to diagnose internal leakage of hydraulic cylinder. On the contrary, pitch-system fault detection has been dealt as a regression problem by Pandit and Infield (2019).

Finally, Gaussian Processes (GP) have been popular to some researchers dealing with pitch system faults. Pandit and Infield (2018) have trained their GP model using power curve, the rotor speed curve and the blade pitch angle curve as the feature set. Guo and Infield (2020) trained a multivariable power curve model with a modified Cholesky decomposition GP.

However, scientists have developed techniques to extract latent information from the SCADA signals in order to provide more enhanced information to wind turbine operators. These techniques belong to the broad area of the so-called dimensionality reduction techniques as well. In general, traditional Principal Component Analysis (PCA) (Jolliffe, 2002) has been applied in many fields, representing a linear transformation of input space. Additionally, nonlinear transformations have been applied to input space using the kernel trick in PCA, resulting in the kernel PCA (Smola, 1998). Nevertheless, the most advanced technique, arisen from the Deep Learning field, is Autoencoders (Goodfellow, Bengio, & Courville, 2016). Autoencoders are mainly a generalization of PCA, and they are based on neural network architectures. Denoising Autoencoders, which is a specific type of regularized Autoencoder has been used for dimensionality reduction techniques in wind turbines by Liu, Cheng, Kong, Wang, and Cui (2019) and Wu et al. (2019). But use of Autoencoders for dimensionality reduction has not been focused on pitch system monitoring. Thus, investigation of them is necessary and it has high potentials to provide more information about the condition of this subsystem to the operators. The extracted information will be also enhanced if the pitch faults, which are contained in the dataset, represent different kind of the most common faults. This is particularly interesting and adds up value in literature because studies in the past have failed to refer to specific types of faults that have been taken into account when setting up their dataset or have presented very limited information. Furthermore, the advantage of having more diverse faults is beneficial when performing identification of those types and that work will be realized in the future by the authors.

The objective of this study is to investigate the development of a Denoising Autoencoder (DAE), as a feature extraction technique, for fault detection of a wind turbine hydraulic pitch system. DAE makes use of nonlinear transformations of input space and its feature extraction potential is assessed through the performance of Support Vector Machine, which is used as classifier. This research has collected the most informative features for the hydraulic pitch system and the training dataset includes normal and faulty points derived from nine different faulty events. These faulty events include diverse faults of every single component in the hydraulic pitch system, whose effect have not been investigated in earlier studies. The performance of the new latent dimensions on the classifier of SVM shows greater performance than using the original input space as input of SVM.

The paper is organized as follows. In Section 2, Deep Autoencoder for dimensionality reduction and feature extraction is described. In Section 3, the theory of SVM for classification problems is presented. Section 4 refers to the dataset of this research, which is referred to the hydraulic pitch system. Section 5 demonstrates the results, followed by the conclusions in the last section.

2. DEEP AUTOENCODER FOR DIMENSIONALITY REDUCTION

Autoencoders have been primarily used for dimensionality reduction tasks. Their clear advantage over other traditional dimensionality reduction techniques such as PCA is that they are based on nonlinear transformation of the input space. An autoencoder is composed of an encoder and a decoder. The encoder transforms the ambient space to a lower-dimensional space, in case of an undercomplete or to a higher dimensional if it is an overcomplete one. On the contrary, the decoder transforms the new feature space back to the original space.

Essentially, an autoencoder is a neural network which tries to learn the copying task of the input space. It requires only the input space and not the label, thus it belongs to unsupervised techniques. The encoder and decoder are typically nonlinear using several activation functions including sigmoid function, hyperbolic tangent function (tanh) or Rectified Linear Unit (ReLU).

More specifically, an encoder maps an input $x \in \mathbb{R}^m$ to a hidden representation h through the activation function f_s, shown in Eq.(1).

$$\mathrm{h} = f_s(Wx + b) \qquad (1)$$

where W is a $m \times m$ weight matrix and b is a bias vector. The decoder tries to reconstruct x from the latent representation, resulting in $\hat{x}$ (Eq. (2)).

$$\hat{\mathrm{x}} = f_s(W'h + b') \qquad (2)$$

Where W' and b' are the parameters of the decoder in a similar way as in the encoder. An autoencoder is said to have tied weights if $W' = W^T$. The parameters of the autoencoder, represented shortly by $\theta = \{W, b\}, \theta' = \{W', b'\}$, are estimated after minimization of the average reconstruction error, demonstrated in Eq. (3).

$$\theta^*, \theta'^* = \underset{\theta,\theta'}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} L(x^{(i)}, \hat{x}^{(i)}) \qquad (3)$$

The loss function L is the traditional mean squared error $L(x, \hat{x}) = \|x - \hat{x}\|^2$.

Even though an autoencoder deals with the copying task of its input to its output, exact reconstruction is useless and no new latent information is extracted. In addition, if both the encoder and decoder functions are given too much capacity, it fails to learn anything useful. That is the reason why researchers suggested regularized autoencoders, which additionally provide sparsity of the representation, smallness of the derivative of the representation and robustness to noise and missing inputs (Goodfellow et al, 2016). Such regularized autoencoders are sparse autoencoders and denoising autoencoders.

Denoising Autoencoders (DAE) are similar to the traditional autoencoders, but the input of them is a corrupted version of original input space. Furthermore, the end goal is to predict the original input and not the corrupted one. Consequently, before implementing autoencoder, the input is corrupted by either adding Gaussian noise or salt-and-pepper noise or masking noise (Vincent, Larochelle, Bengio, & Manzagol, 2008). In other words, the input of a DAE will be the corrupted $\tilde{x}$ and not x, and the loss function is the L^2 norm between the reconstruction of corrupted datapoints and original datapoints.

3. SVM AS CLASSIFIER

Support Vector Machines (SVM) (Cortes & Vapnik, 1995) have gained a lot of attention from 2000 onwards due to its ability to provide better classification performance, compared to Artificial Neural Networks. However, it can be also used for regression problems. In particular, SVMs nonlinearly map the input space into a higher-dimensional space and then a linear decision boundary is set to separate the classes. Therefore, it may seem that a linear decision line has been constructed, but in reality, this line is nonlinear in the original space. Finally, the decision boundary is based on the support vectors, which are essentially a small amount of datapoints that allow to define the best separation boundary between two classes.

SVM is given in Eq. (4), which clearly shows its dependence on a nonlinear transformation φ.

$$f(x) = u^T \varphi(x) + d \qquad (4)$$

where $\varphi(x)$ is the nonlinear transformation of the input space to the high-dimensional feature space. The output of SVM is not probabilities, but the class label. In other words, if f is positive, SVM predicts the positive class and if f is negative, it predicts the negative class.

The u and d parameters are determined by minimizing the regularized risk function. However, most of the times some of the datapoints are allowed to be misclassified, leading to soft margin SVM. Soft margin SVM is given by minimizing the dual Lagrangian $\tilde{L}$ function, shown in Eq. (5), where some of the datapoints are allowed to be misclassified.

$$\tilde{L}(a) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m y_n y_m K(x_n, x_m) \quad (5)$$

where $K(x_n, x_m) = \varphi(x_n)^T \varphi(x_m)$, that represents a kernel, thus leading to nonlinear SVM. a_n are non-negative Lagrangian multipliers, which define the final solution for u and d shown in Eq. (6) and Eq. (7) respectively.

$$u = \sum_{n=1}^{N} a_n y_n \phi(x_n) \quad (6)$$

$$d = \frac{1}{N_{\mathcal{M}}} \sum_{n \epsilon \mathcal{M}} \left(y_n - \sum_{m \epsilon S} a_m y_m K(x_n, x_m) \right) \quad (7)$$

where $\mathcal{M}$ represent the set of indices of data points where $0 < a_n < C$, C is the regularization constant and y_n are the target values of the input x_n.

Common kernels for kernel SVM are polynomial kernels, sigmoid kernel and Radial Basis Function (RBF) kernels (Eq. (8)), whose hyperparameter γ is half of the variance of the standard normal density.

$$K(x_n, x_m) = \exp\left(-\gamma \|x_n - x_m\|^2\right) \quad (8)$$

4. DATASET

This study makes use of 10-year long available data, derived from the SCADA system of a windfarm in western Finland. The studied windfarm includes five wind turbines of 2.3 MW, which are fixed-speed and have a hydraulic pitch system. These SCADA data include average, standard deviation, maximum and minimum of several measurements stored in 10-min intervals. Nevertheless, the objective of this research is fault detection in the hydraulic pitch system, thus the most effective parameters have been selected which have the biggest impact on its operation.

These features have been preprocessed and labelled according to Korkos et al (2022). Then, features have been normalized using Min-Max normalization (Eq. (9)). The normalized values would be in the range between 0 and 1.

$$x_{new}^{i} = \frac{x^{i} - x_{min}}{x_{max} - x_{min}} \quad (9)$$

where, x^i and x_{new}^i are the original and normalized feature respectively and x_{min} and x_{max} are the minimum and maximum values of each feature.

Table 1. SCADA features and their short names demonstrates the list of features that were used as input at the dimensionality reduction technique. Their names are mentioned using shortened form followed by {"_mean", "_stdev", "_max", "_min"}. However, only gust wind speed contains a single value, instead of the statistical quantities mentioned before. For example, if maximum value of power output is mentioned, the shortened name will be "PO_max". In total, the original feature space is 49-dimensional.

Table 1. SCADA features and their short names

Name	Description	Blade
RS	Rotor speed	-
BAA	Blade angle A	A
BAB	Blade angle B	B
BAC	Blade angle C	C
WS	Wind speed	-
PO	Power output	-
Gust_WS	Gust wind speed	-
HPrA	Hub Pressure A	A
HPrB	Hub Pressure B	B
HPrC	Hub Pressure C	C
HydP	Hydraulic Pressure	-
AmbT	Ambient Temp.	-
HubT	Hub Temp.	-

This study collected a dataset which contains normal and faulty operation datapoints. More specifically, faulty dataset contains data when different kind of events of faults were occurred. In particular, Table 2 shows the nine pitch events that have been taken into account for this study. For normal data points the label has been assigned to zero and for faulty data points the label is one. The data are owned by Suomen Hyötytuuli Oy and are not publicly available due to confidentiality reasons.

Table 2. Event list

No	Pitch event
1	Hydraulic hoses and oils replacement
2	Hub oil leakage + Hyd. Oil replacement + Bl. valve 6 replacement
3	Block replacement at blade B (No3)

4	Block leakage in blade B(No1)
5	Replacement of A- blade valve 102 (No3)
6	Replacement of A, B, C- blade valve 116 (No3)
7	Nitrogen accumulator (No 4) replacement of Blade A (No5)
8	Blade tracking error during stop/operation of Blade A (No1)
9	Replacement of hyd. cylinder (No2)

5. Results and discussion

New features have been extracted using a Denoising Autoencoder (DAE). Autoencoders have the advantage to use nonlinear transformation of input space. Thus, they belong to nonlinear dimensionality reduction techniques. This study investigated different architectures of DAEs. These architectures are presented on the Table 3, as well as their activation functions. If *n* is the dimension of original dataset, *n* = 49 for this study which is the dimension of both input and output layer.

Table 3. Different architectures of DAEs under investigation

No	**Architecture**	**Activation Function**
1	[*n*,64,32,16,8,16,32,64,*n*]	ReLU
2	[*n*,32,32,16,8,16,32,32,*n*]	ReLU
3	[*n*,32,32,16,8,16,32,32,*n*]	sigmoid
4	[*n*,32,8,32,*n*]	sigmoid
5	[*n*,32,24,16,10,6,10,16,24,32,*n*]	sigmoid

The best architecture was achieved by [*n*,32,32,16,8,16,32,32,*n*] (Figure 1) using sigmoid function as activation function. In addition, Mean Squared Error (MSE) was chosen as loss function and the optimization algorithm was Adam algorithm. The corruption of input was selected to be a Gaussian noise. Gaussian noise was represented by the standard normal distribution $N(0,1)$ multiplied by 0.02. This multiplier has been chosen after appropriate tuning.

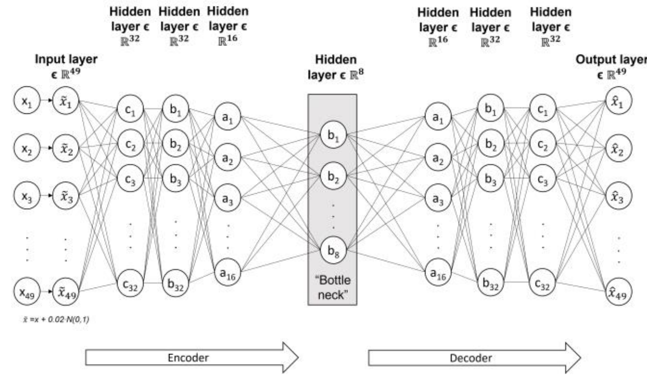

Figure 1. Denoising Autoencoder (DAE) [*n*,32,32,16,8,16,32,32,*n*] architecture

Figure 2 demonstrates a two-dimensional representation of 8D latent space. T-distributed Stochastic Neighbor Embedding has been applied to the new extracted features (8D) in order to provide a visualization of them. Figure 2 shows that the two classes can be clearly separated. Thus, features extracted by the developed DAE, shown in Figure 1, really extracts hidden information and helps to separate the two classes more clearly.

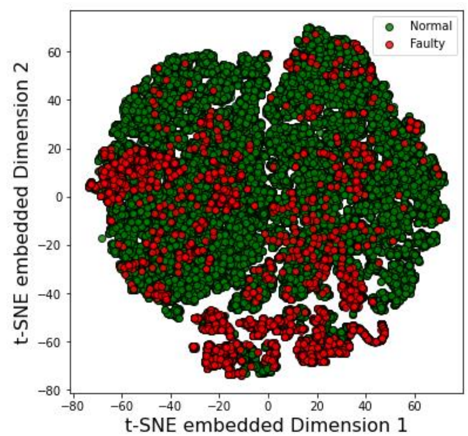

Figure 2. Two of the new extracted features using Autoencoder [*n*,32,32,16,8,16,32,32,*n*]

After the reduction of the dimensions and the extraction of the new features, Support Vector Machine classifier was trained in order to perform the fault detection task. Hyperparameter tuning of SVM has been performed through cross validation between the regularization constant C, type of kernel and hyperparameter γ, in case of Radial Basis Function (RBF) kernel. More specifically, this research investigated values of C in the list {0.01, 0.1, 1, 10, 100, 1000} (being either linear or RBF kernel) as well as 'γ' values in the list {0.1, 1, 10, 50. 100, 500} should the kernel is RBF.

Dataset has been split in two parts, i.e., 80% for training and 20% for testing. Training dataset is separated in training dataset and validation set during cross-validation process in order to determine the hyperparameters. The final training of the SVM classifier has been done in the whole training set. The performance of the classifier was assessed based on the F1-score, shown in Eq. (10). This performance metric was chosen instead of other such as accuracy because normal operation class represents the vast majority of the datapoints and a correct evaluation requires to take into account that missed faulty points will be shown at the performance metric.

$$F1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \qquad (10)$$

where TP represent True Positive, meaning that the faulty points (label "1") were truly detected. The same notion is followed for FP (False Positive) and FN (False Negative), whose actual label was "0" and "1" respectively, but the opposite class was predicted.

Table 4 summarizes the results of F1-score when performing 3-fold cross validation for every combination of kernel, C and γ values. F1-scores are given as an average value during calculation of it in the 3-fold datasets and standard deviation is presented within parentheses. Best linear SVM model is acquired for C value of 1000. In contrast, the highest F1-scores using RBF SVM is received when using C = 1000 and γ = 10, which has the best performance in the validation set among all investigated classifiers.

Table 4. Best F1-scores of SVM for different pairs of C, γ and kernel

F1-score	C	γ	kernel
0.731 (+/-0.013)	1000	-	linear
0.582 (+/-0.018)	0.01	10	RBF
0.816 (+/-0.021)	0.1	50	RBF
0.917 (+/-0.006)	1	100	RBF
0.936 (+/-0.007)	10	50	RBF
0.937 (+/-0.004)	100	10	RBF
0.938 (+/-0.005)	1000	10	RBF

Therefore, when using the developed Denoising Autoencoder, as feature extractor shown in Figure 1, the SVM performance for C = 1000 and γ = 10 is 0.9457%, according to F1-score. This study uses as benchmark the SVM performance when using only the original features. Benchmark's performance is 0.8538%, thus the developed DAE provides increase of 10.8%. This result outperforms the performance of Adaptive Neuro Fuzzy Inference system (ANFIS) presented in Korkos et al. (2022). Results from other similar studies could not be directly compared to the present one. The reason is the dataset variability since each researcher uses a different dataset. However, Leahy et al. (2016) attained 65% F1-score for fault detection task without mentioning the details of the faults. Moreover, Hu et al. (2017) achieved 90% F1-score by increasing their feature set, which contained only the original SCADA features. Finally, APK-ANFIS model, developed by Chen et al. (2015), achieved 50% of F1-score for fixed-speed wind turbines using some pitch faults, providing no information about them. Consequently, the attained F1-score of the present study leads to the conclusion that Denoising Autoencoders are very powerful at extracting useful information out of the dataset.

6. Conclusion

In this paper, a Denoising Autoencoder (DAE) has been developed to extract hidden information that will contribute to more efficient monitoring of wind turbine hydraulic pitch system. The efficiency of DAE has been evaluated based on the performance of a Support Vector Machines (SVM) classifier, which uses the new extracted features as input. More specifically, the original feature set had been 49-dimensional, including from environmental parameters to several pressures in the pitch system. Hence, the nonlinear transformations, employed by the developed DAE, attained 0.9457%, which was 10.8% better than the case of SVM using directly the original feature set. As a result, pitch system, which is crucial for a wind turbine, can be monitored more effectively and accurately. Additionally, those extracted features may be used in future studies for diagnosing each fault separately. That information would provide great assistance to wind turbine operators and will lower maintenance costs. Possible other classifiers, from the Deep Learning field, may be investigated in the future such as 1D Convolutional Neural Network or Long Short-Term Memory network (LSTM).

Acknowledgement

This research was funded by the Doctoral School of Industry Innovations (DSII) of Tampere University and Suomen Hyötytuuli Oy.

References

Afrasiabi, S., Afrasiabi, M., Parang, B., Mohammadi, M., Arefi, M. M., & Rastegar, M. (2019). Wind turbine fault diagnosis with Generative-Temporal Convolutional Neural Network, *2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*, pp. 1-5. doi: 10.1109/EEEIC.2019.8783233

Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*, New York: Springer Science+Business Media, LLC.

Blanco, I. (2009). The economics of wind energy, *Renewable and Sustainable Energy Reviews*, vol. 13, pp. 1372-1382. doi: 10.1016/j.rser.2008.09.004

Carroll, J., McDonald, A., & McMillan, D. (2016). Failure rate, repair time and unscheduled O & M cost analysis of offshore wind turbines, *Wind Energy*, vol. 19, pp. 1107-1119. doi: 10.1002/we.1887

Cortes, C., Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, vol. 20, pp. 273-297. doi: 10.1007/BF00994018

Chen, J., Li, J., Chen, W., Wang, Y., & Jiang, T. (2020) Anomaly detection for wind turbines based on the reconstruction of condition parameters using stacked denoising autoencoders, *Renewable Energy*, vol. 147, pp. 1469-1480. doi: 10.1016/j.renene.2019.09.041

Chen, B., Matthews, P.C., & Tavner, P.J. (2013) Wind turbine pitch faults prognosis using a-priori knowledge-based ANFIS, *Expert Systems with Applications*, vol. 40, pp. 6863-6876. doi: 10.1016/j.eswa.2013.06.018

Chen, B., Matthews, P.C., & Tavner, P.J. (2015) Automated on-line fault prognosis for wind turbine pitch systems using supervisory control and data acquisition, *IET Renewable Power Generation*, vol. 9, pp. 503-513. doi: 10.1049/iet-rpg.2014.0181

Chen, B., Zappala, D., Crabtree, C.J., & Tavner, P.J. (2014) *Survey of commercially available SCADA data analysis tools for wind turbine health monitoring*. Technical Report. Durham University School of Engineering and Computing Sciences

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*, Cambridge, MA: MIT Press

Guo, P., & Infield, D. (2020). Wind turbine power curve modeling and monitoring with Gaussian process and SPRT, *IEEE Trans. Sustain. Energy*, vol. 11, pp. 107-115. doi: 10.1109/TSTE.2018.2884699

Helbing, G., & Ritter, M. (2018). Deep Learning for fault detection in wind turbines, *Renewable and Sustainable Energy Reviews*, vol. 98, pp. 189-198. doi: 10.1016/j.rser.2018.09.012

Hu, R.L., Leahy, K., Konstantakopoulos, I.C., Auslander, D.M., Spanos, C.J., & Agogino, A.M. (2017). Using domain knowledge features for wind turbine diagnostics, *Proceedings of 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*. pp. 300-305. doi: 10.1109/ICMLA.2016.172

Kong, Z., Tang, B., Deng, L., Liu W., & Hana, Y. (2020). Condition monitoring of wind turbines based on spatio-temporal fusion of SCADA data by convolutional neural networks and gated recurrent units, *Renewable Energy*, vol. 146, pp. 760-768. doi: 10.1016/j.renene.2019.07.033

Korkos, P., Linjama, M., Kleemola, J., & Lehtovaara, A. (2022). Data annotation and feature extraction in fault detection in a wind turbine hydraulic pitch system. *Renewable Energy*, vol. 185, pp. 692-703. doi: 10.1016/j.renene.2021.12.047

Leahy, K., Hu, R.L., Konstantakopoulos, I.C., Spanos, C.J., & Agogino, A.M. (2016). Diagnosing wind turbine faults using machine learning techniques applied to operational data, *2016 IEEE International Conference on Prognostics and Health Management (ICPHM), pp.* 1–8. doi: 10.1109/ICPHM.2016.7542860

Leahy, K., Hu, R.L., Konstantakopoulos, I.C., Spanos, C.J., Agogino, A.M., & O'Sullivan, D.T.J. (2018). Diagnosing and predicting wind turbine faults from SCADA data using support vector machines, *International Journal of Prognostics and Health Management*, vol. 9 (1), pp. 1-11. doi: 10.36001/ijphm.2018.v9i1.2692

Liu, Y., Cheng, H., Kong, X., Wang, Q., &. Cui, H. (2019). Intelligent wind turbine blade icing detection using supervisory control and data acquisition data and ensemble deep learning, *Energy Science Engineering*, vol. 7, pp. 2633-2645. doi: 10.1002/ese3.449

Liu, X., Teng, W., Wu, S., Wu, X., Liu, Y., & Ma, Z. (2021), Sparse dictionary learning based adversarial variational auto-encoders for fault identification of wind turbines, *Measurement*, vol. 183. doi: 10.1016/j.measurement.2021.109810

Pandit, R.K., & Infield, D. (2018). Gaussian process operational curves for wind turbine condition monitoring, *Energies*, vol. 11 (7). doi: 10.3390/en11071631

Pandit, R.K., & Infield, D. (2019). Comparative assessments of binned and support vector regression-based blade pitch curve of a wind turbine for the purpose of condition monitoring, *International Journal of Energy and Environmental Engineering*, vol. 10, pp. 181-188. doi: 10.1007/s40095-018-0287-3

Qian, P., Tian, X., Kanfoud, J., Lee, J.L.Y., & Gan, T.H. (2019). A Novel Condition Monitoring Method of Wind Turbines Based on Long Short-Term Memory Neural Network, *Energies*, vol. 12 (18). doi: 10.3390/en12183411

Ribrant, J., & Bertling, L.M. (2007) Survey of failures in wind power systems with focus on Swedish wind power plants during 1997-2005, *IEEE Trans. Energy Convers.*, vol. 22, pp. 167-173. doi: 10.1109/PES.2007.386112

Schlechtingen, M., Santos, I.F., & Achiche, S. (2013) Wind turbine condition monitoring based on SCADA data using normal behavior models. Part 1: System description, *Applied Soft Computing*, vol. 13, pp. 259-270. doi: 10.1016/j.asoc.2012.08.033

Schlechtingen, M., & Santos, I.F. (2014) Wind turbine condition monitoring based on SCADA data using normal behavior models. Part 2: Application examples, *Applied Soft Computing*, vol. 14, pp. 447-460. doi: 10.1016/j.asoc.2013.09.016

Stetco, A., Dinmohammadi, F., Zhao, X., Robu, V., Flynn, D., Barnes, M., Keane, J., & Nenadic, G. (2019) Machine learning methods for wind turbine condition monitoring: A review, *Renewable Energy*. vol. 133, pp. 620-635. doi:10.1016/j.renene.2018.10.047

Tautz-Weinert, J., & Watson, S.J., (2017). Using SCADA data for wind turbine condition monitoring - A review, *IET Renewable Power Generation*, vol. 11, pp. 382-394. doi: 10.1049/iet-rpg.2016.0248

Ulmer, M., Jarlskog, E., Pizza, G., Manninen, J., & Goren Huber, L. (2020). Early Fault Detection Based on Wind Turbine SCADA Data Using Convolutional Neural Networks. *PHM Society European Conference*, vol. 5(1), 9. doi: 10.36001/phme.2020.v5i1.1217

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P.A. (2008). Extracting and composing robust features with denoising autoencoders, *Proceedings 25th International Conference on Machine Learning* (pp. 1096-1103), Helsinki, Finland. doi: 10.1145/1390156.1390294

Wilkinson, M., Hendriks, B., Spinato, F., Harman, K., Gomez, E., Bulacio, H., Roca, J., Tavner, P., Feng, Y., & Long, H. (2010). Methodology and results of the Reliawind reliability field study, *European Wind Energy Conference Exhibition, EWEC 2010*. April 20-23, Warsaw, Poland.

WindEurope. (2022). *Wind energy in Europe - 2021 Statistics and the outlook for 2022-2026*, Annual report, Brussels, Belgium

Wu, X., Jiang, G., Wang, X., Xie, P., & Li, X. (2019). A Multi-Level-Denoising Autoencoder approach for wind turbine fault detection, *IEEE Access*, vol. 7, pp. 59376-59387. doi: 10.1109/ACCESS.2019.2914731

Wu, X., Su, R., Lu, C., & Rui, X. (2015). Internal leakage detection for wind turbine hydraulic pitching system with computationally efficient adaptive asymmetric SVM, *Proceedings of 2015 34th Chinese Control Conf.*, pp. 6126-6130, July 28-30, Hangzhou, China. doi: 10.1109/ChiCC.2015.7260599

Xiang, L., Wang, P., Yang, X., Hu, A., & Su, H. (2021). Fault detection of wind turbine based on SCADA data analysis using CNN and LSTM with attention mechanism, *Measurement*, vol. 175. doi: 10.1016/j.measurement.2021.109094

Xiang, L., Yang, X., Hu, A., Su, H., & Wang, P. (2022). Condition monitoring and anomaly detection of wind turbine based on cascaded and bidirectional deep learning networks, *Applied Energy*, vol. 305, doi: 10.1016/j.apenergy.2021.117925

Yang, W., Court, R., & Jiang, J., (2013). Wind turbine condition monitoring by the approach of SCADA data analysis, *Renewable Energy*, vol. 53, pp. 365-376. doi: 10.1016/j.renene.2012.11.030

Yang, L., & Zhang, Z. (2021). Wind turbine gearbox failure detection based on SCADA data: A Deep Learning-based approach, *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-11, doi: 10.1109/TIM.2020.3045800

Zaher, A., McArthur, S.D.J., Infield, D.G., & Patel, Y. (2009) Online wind turbine fault detection through automated SCADA data analysis. *Wind Energy*, vol. 12, pp. 574-593. doi: 10.1002/we.319