ISTANBUL TECHNICAL UNIVERSITY ★ ENERGY INSTITUTE

INCIPIENT FAULT DETECTION IN WIND TURBINES

Ph.D. THESIS

Ayşe Gökçen KAVAZ TAŞKINER

Energy Science and Technology Division

Energy Science and Technology Programme

OCTOBER 2019

ISTANBUL TECHNICAL UNIVERSITY ★ ENERGY INSTITUTE

INCIPIENT FAULT DETECTION IN WIND TURBINES

Ph.D. THESIS

Ayşe Gökçen KAVAZ TAŞKINER
(301112006)

Energy Science and Technology Division

Energy Science and Technology Programme

Thesis Advisor: Dr. Lect. Burak BARUTÇU

OCTOBER 2019

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ ENERJİ ENSTİTÜSÜ

RÜZGAR TÜRBİNLERİNDE GELİŞMEKTE OLAN HATA ÖNGÖRÜSÜ

DOKTORA TEZİ

Ayşe Gökçen KAVAZ TAŞKINER
(301112006)

Enerji Bilim ve Teknoloji Anabilim Dalı

Enerji Bilim ve Teknoloji Programı

Tez Danışmanı: Dr. Öğr. Üyesi Burak BARUTÇU

EKİM 2019

Ayşe Gökçen KAVAZ TAŞKINER, a Ph.D. student of ITU Energy Institute student ID 301112006, successfully defended the thesis/dissertation entitled “INCIPIENT FAULT DETECTION IN WIND TURBINES”, which she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Dr. Lect. Burak BARUTÇU** İstanbul Technical University	....
Jury Members :	**Prof. Dr. Serhat ŞEKER** İstanbul Technical University	....
	Prof. Dr. Osman Nuri UÇAN Altınbaş University	....
	Prof. Dr. Önder GÜLER İstanbul Technical University	....
	Prof. Dr. Gökhan GÖKMEN Marmara University	....

Date of Submission : 21 June 2019
Date of Defense : 16 October 2019

To my family,

FOREWORD

I would like to express my sincere gratitude to my supervisor Dr. Lect. Burak Barutçu for the guidance and mentorship he has supplied for this thesis over the years.

I would like to thank Assoc. Prof. Dr. Niels Kjølstad Poulsen for his valuable mentorship in my research stay in the Technical University of Denmark. I would also like to acknowledge the support of the Scientific and Technological Research Council of Turkey (TUBITAK) for granting a scholarship for me in this external research duration.

In addition, I would like to thank my colleagues and friends in Istanbul Technical University, Energy Institute for their help and support through all this period.

Finally, I would like to thank my parents, brother and spouse for their endless support during the creation of this thesis and also in every aspect of my life.

June 2019

Ayşe Gökçen KAVAZ TAŞKINER
(Systems Engineer)

TABLE OF CONTENTS

Page

FOREWORD .... **ix**
TABLE OF CONTENTS .... **xi**
ABBREVIATIONS .... **xiii**
LIST OF TABLES .... **xv**
LIST OF FIGURES .... **xvii**
SUMMARY .... **xix**
ÖZET .... **xxi**
1. INTRODUCTION **1**
1.1 A General Look at the Maintenance Strategies 3
1.2 Data Acquisition in Wind Turbines 4
1.3 Model-Based and Data-Driven Fault Detection Strategies 6
1.4 Wind Turbine Fault Detection Using SCADA Data 7
1.5 Purpose of the Thesis and Contributions 10
2. BACKGROUND **13**
2.1 General Information on Wind Turbines 13
2.2 Data Characteristics 15
2.2.1 Wind parameters 16
2.2.2 Temperature data 16
2.2.3 Operational data 17
2.2.4 Status data 17
2.3 Artificial Neural Networks 18
2.3.1 Multilayer feedforward neural network 20
2.3.2 Radial basis function neural networks 23
2.3.3 Generalized regression neural networks 24
3. SENSOR VALIDATION **27**
3.1 Data Description 28
3.2 ANN Input-Output Structures 29
3.2.1 Auto-associative neural networks 29
3.2.2 MISO neural networks 30
3.3 Methodology and Network Selection 31
3.4 Results for Sensor Validation 35
4. FEATURE CONSTRUCTION AND SELECTION **43**
4.1 Feature Construction 44
4.1.1 Knowledge-based features 45
4.1.2 Difference features 45
4.1.3 Time series features 45
4.1.4 Statistical features 46
4.2 Feature Selection 46
4.2.1 Filter-based feature selection 47
4.2.1.1 Fisher method 48

4.2.1.2 Relief method 48
4.2.2 Wrapper-based feature selection 49
4.3 Results 50
5. THREE LEVEL FAULT CLASSIFICATION 55
5.1 Fault Information 56
5.2 Classification Levels 58
5.2.1 Fault detection 58
5.2.2 Fault isolation 59
5.2.3 Fault prediction 59
5.3 Data Pre-processing 60
5.4 Methods to Improve Training Performance in Imbalanced Datasets 62
5.4.1 Oversampling of minority class 62
5.4.2 Undersampling of majority class 63
5.5 ANN Architectures 63
5.6 Performance Evaluation Metrics 65
5.7 Results and Discussions 67
5.7.1 Results of the fault detection level 67
5.7.2 Results of the fault isolation level 69
5.7.3 Results of the fault prediction level 71
6. CONCLUSION 77
REFERENCES 83
CURRICULUM VITAE 89

ABBREVIATIONS

AANN	**:** Auto Associative Neural Network
ANN	**:** Artificial Neural Networks
ARX	**:** Auto-Regressive with eXogenous
BP	**:** Backpropagation
FSRC	**:** Full Signal Reconstruction
GRNN	**:** General Regression Neural Network
HAWT	**:** Horizontal Axis Wind Turbine
MFNN	**:** Multilayer Feedforward Neural Network
MISO	**:** Multi Input Single Output
MLP	**:** Multilayer Perceptron
OAA	**:** One Against All
RBFNN	**:** Radial Basis Function Neural Network
SCADA	**:** Supervisory Control and Data Acquisition System
SMOTE	**:** Synthetic Minority Oversampling Technique
UNFCC	**:** United Nations Framework Convention on Climate Change
VAWT	**:** Vertical Axis Wind Turbine
WT	**:** Wind Turbine
YSA	**:** Yapay Sinir Ağı

LIST OF TABLES

Page

Table 2.1 : Main subsystems and assemblies of wind turbines. **15**
Table 2.2 : Wind parameters. **16**
Table 2.3 : Temperature data. **17**
Table 2.4 : Operational data. **17**
Table 2.5 : Status data. **18**
Table 3.1 : ANN structures designed for the sensor validation purpose. **32**
Table 3.2 : RMSE values for each ANN. **37**
Table 4.1 : Top 10 features selected by Fisher algorithm. **51**
Table 4.2 : Top 10 features selected by Relief algorithm. **51**
Table 4.3 : Top 10 features selected by the proposed feature engineering techniques and a heuristic way. **52**
Table 5.1 : Frequent fault types. **57**
Table 5.2 : Time windows created for the pre-fault instances. **61**
Table 5.3 : Prediction horizon for generator heating faults. **75**

LIST OF FIGURES

Page

Figure 2.1 : Main parts of HAWTs. **14**
Figure 2.2 : Signal flow graph of MFNN networks. **21**
Figure 3.1 : Locations of sensors used for the sensor validation purpose. **29**
Figure 3.2 : Auto-Associative network structure. **30**
Figure 3.3 : MISO network structure. **31**
Figure 3.4 : Regression plots for (a) Auto-Associative MFNN (b) MISO MFNN. .. **33**
Figure 3.5 : Regression plots for (a) Auto-Associative RBFNN(b) MISO RBFNN. **34**
Figure 3.6 : Regression plots for (a) Auto-Associative GRNN (b) MISO GRNN. . **35**
Figure 3.7 : Measured and network values for Case 0 with Auto Associative – MFNN. **38**
Figure 3.8 : Measured and network output values for Case 1 with Auto Associative-MFNN. **38**
Figure 3.9 : Measured and network output values for Case 2 with Auto Associative-MFNN. **39**
Figure 3.10 : Measured and network values for Case 0 with MISO-MFNN. **40**
Figure 3.11 : Measured and network values for Case 1 with MISO-MFNN. **40**
Figure 3.12 : Measured and network values for Case 2 with MISO-MFNN. **41**
Figure 4.1 : Filter-based feature selection. **47**
Figure 4.2 : Wrapper-based feature selection. **49**
Figure 5.1 : Normalized power output values for the normal and faulty operation statuses. **58**
Figure 5.2 : OAA network structure used in the fault isolation level. **65**
Figure 5.3 : Performance metrics in the fault detection level for a) Original training set b) Only oversampled case c) Only undersampled case d) Oversampled and under sampled case. **68**
Figure 5.4 : Performance metrics for the fault isolation level. **69**
Figure 5.5 : Performance metrics obtained by a) Heuristic b) Systematic feature selection approaches. **70**
Figure 5.6 : Performance metrics for the fault prediction level. **72**
Figure 5.7 : ANN outputs for a section of normal operation region. **73**
Figure 5.8 : ANN outputs and fault beginning instance for Fault 1. .... **74**
Figure 5.9 : ANN outputs and fault beginning instances for Faults 4 and 5. .... **74**

INCIPIENT FAULT DETECTION IN WIND TURBINES

SUMMARY

The global goal of increasing the share of renewable energy supplies in the overall energy consumption has resulted in a rising focus on technological developments in this field. Wind energy is one of the promising options amongst renewable energy sources with a growing number of investments and rising installation number and capacities. Due to the increasing demands from wind energy industry, the requirement of more effective wind farm operations has emerged. Wind turbine maintenance systems are essential parts towards achieving this requirement.

Today, maintenance of wind turbines is mostly based on preventive and corrective actions. However, these approaches are inadequate to meet current demands from wind energy industry. With the developments in computational capabilities and data collection systems, a high potential of using advanced data-driven techniques has appeared for the maintenance of wind turbines. This thesis proposes a predictive maintenance approach using data which were collected from a wind turbine Supervisory Control and Data Acquisition System (SCADA).

SCADA is the primary interface between the wind farm operators and wind turbines which allows remote and local control and monitoring. Various kinds of data are collected by SCADA systems such as wind parameters, temperature values, operational and status data. It is a built-in part in most medium and large-scale modern wind turbines. Therefore, a major advantage of using SCADA data for fault detection purposes is that additional hardware costs are not required. However, there are imperfections in the data such as low sampling frequency and high ratio of missing values. To handle these disadvantages, a suitable approach is required which was provided by Artificial Neural Networks (ANN) in this thesis. Moreover, wind turbines are highly non-linear systems with complex control parts and ANN models are also powerful on handling such complex systems. By this way, this thesis aims to design a cost-effective maintenance system for the overall wind turbine.

Firstly, a sensor validation technique to detect faults of temperature sensors was designed. The method solely uses sensor measurements to detect calibration drifts by analyzing a set of sensors located on components with similar temperature characteristics. Auto-Associative and Multi-Input-Single-Output ANN structures were employed. The concurrent use of them provided the best outputs on the detection of the simulated calibration drift. The results prove that, validation of sensors can be realized by continuously monitoring sensor readings. It is advantageous as there is no need of dismantling sensors to test their calibration. Also, this method is a cost-effective solution in terms of not requiring redundant sensor use.

After the sensor validation part, a 3-level fault classification system to detect, isolate and predict wind turbine faults was realized. The types of faults attempted in this part

are frequent and non-fatal wind turbine faults. Distinguishing these kind faults is a challenging task because they do not show as strong indications as fatal faults do. However, as they are observed frequently in all wind turbines and decrease turbine performance, detection of them is a significant research topic. The core part of algorithms employed in this part is ANN models, in addition to them assistive methods were also designed to increase the fault classification performance.

For the initial step of this part, feature construction and selection techniques were employed to find out an effective subset of inputs to be used as inputs of ANN models. These pre-processing tasks are important to design fast and accurate models as performance of algorithms strongly depend on the feature representation of input data in artificial intelligence applications. Raw data collected by the SCADA system were used to generate new features that possibly give more information about the hidden relations indicating fault occurences comparing to the raw features. In the feature selection step, both raw and constructed features were analyzed to identify a subset of relevant features to reduce computational burden and increase accuracy of models. Two different feature selection methods were used in a hybrid way, which are filter and wrapper-based methods. The results show that, the feature construction and selection algorithms designed are useful especially in terms of reducing false fault alarms which is an important issue in fault detection systems built using SCADA data.

Finally, a 3-level classification scheme for wind turbine faults was designed using ANN models. By this way, a complete system was formed that provides required information by wind farm operators to take actions or measures in case of a current or an upcoming fault. In the detection level, the status of the turbine was analyzed to find out if the turbine is in a normal or a faulty mode. In the fault isolation level, the specific subsystem subjected to fault was attempted to be found. Therefore, this level includes distinguishing detected faults from each other. Finally, in the fault prediction level it was aimed to predict faults in advance to inform operators for possible prevention or repairing actions. We have obtained comprehensive results proving that the proposed methods are effective in all levels of fault classification. Our findings support the idea that despite the shortcomings of SCADA data, ANN models used with assistive methods are powerful on the classification of wind turbine faults. As a result, this thesis contributes to efforts of designing a cost-effective predictive maintenance approach for wind turbines.

RÜZGAR TÜRBİNLERİNDE GELİŞMEKTE OLAN HATA ÖNGÖRÜSÜ

ÖZET

Enerji talebi, dünya genelinde sanayi devriminden günümüze sürekli artmaktadır ve bu durumun ilerleyen dönemlerde de devam etmesi beklenmektedir. Küresel bazda nüfus artışı, değişen enerji kullanım alışkanlıkları ve artan sanayileşme enerji talebindeki artışın temel sebeplerindendir.

Günümüzde enerji talebi büyük ölçüde fosil kaynakların kullanımıyla karşılanmaktadır. Fakat, fosil kaynakların iklim değişiminin ana nedenlerinden olan zararlı çevresel etkileri nedeniyle tüketim miktarının küresel anlaşmalarla düşürülmesi hedeflenmektedir. Ayrıca fosil kaynakların hızla tükenmekte olan sınırlı kaynaklar olması ve yüksek oranda kullanımının fosil yakıt ithalatçısı olan ülkelere bağımlılığı arttırması gibi nedenler de tüketimlerinin düşürülmesi yönündeki çalışmaların gerekçelerindendir.

Rüzgar enerjisi, enerji kaynaklarının çeşitliliğinin arttırılması konusunda yüksek potansiyele sahip olan alternatifler arasındadır. Bu nedenle, rüzgar enerjisi konusunda yatırımlar ve teknolojik gelişmeler önem kazanmaktadır. Rüzgar türbinlerinin tüm alt sistemlerinde yapılan geliştirmelerle maliyetlerinin düşürülmesi hedeflenmektedir. İşletme ve bakım çalışmaları, rüzgar türbinlerinin ana maliyet kaynaklarındandır. Bu tezde, rüzgar türbinlerinde zaman içinde gelişmekte olan hataların tespiti ve öngörüsü için yöntemler sunulmaktadır.

Günümüzde, rüzgar türbinleri için genellikle önleyici ve onarıcı bakım yöntemleri uygulanmaktadır. Fakat rüzgar enerjisi için taleplerin hızla artmakta olması nedeniyle daha etkili bakım çalışmalarının yapılması gerekliliği doğmuştur. Ayrıca rüzgar türbinlerinin yerleşim yerlerinden uzakta konumlandırılması ve faaliyet gösterdikleri koşulların çevresel açıdan zorlayıcı olması da işletim ve bakım yöntemlerinde geliştirme yapılmasını önemli hale getirmektedir.

Bu tez çalışmasında, bir rüzgar türbininden alınan çeşitli veriler kullanılarak türbin genelinde oluşan hataların tespiti ve öngörüsü üzerinde çalışılmıştır. Rüzgar türbinlerinde başlıca iki veri toplama yöntemi bulunmaktadır. Birincisi, belirlenen bileşenler için özel olarak seçilen sensörler yerleştirilerek gereken verilerin toplanmasıdır. İkincisi ise Denetim Kontrol ve Veri Toplama (Supervisory Control and Data Acquisition - SCADA) sistemi sayesinde türbin geneli ile ilgili bilgi verebilecek verilerin kaydedilmesidir. Bu sayede sıcaklık verileri, rüzgar parametreleri, türbinin mevcut operasyon parametreleri ve bulunduğu durum ile ilgili veriler elde edilebilmektedir. SCADA, modern rüzgar türbinlerinin çoğunda ekstra maliyet gerektirmeden bulunan bir sistemdir. Bu nedenle, SCADA verileri değerlendirilerek tasarlanan hata öngörü sistemleri maliyet etkin bir çözüm sunabilmektedir. Öte yandan, SCADA sistemlerinin temel tasarım amaçları türbin aktivitelerinin izlenmesidir. Dolayısıyla hata öngörü sistemleri için veri kalitesi açısından özel olarak yerleştirilmiş sensörler kadar uygun değildir. Örnekleme periyodu genellikle 10 dakikadır ve sık sık eksik verilerle karşılaşılmaktadır. Bu

nedenle, SCADA verilerinin hata öngörüsü amaçlı kullanımında, bu dezavantajlara toleransı olan gelişmiş bir algoritma yapısının kurulması önem kazanmaktadır. Ayrıca, rüzgar türbinleri, doğrusal olmayan birçok alt sistemden ve kompleks kontrol bölümlerinden oluşmaktadır. Bu tezde önerilen algoritmaların temelinde bulunan Yapay Sinir Ağları (YSA) bu tür problemlerde etkili çözümler sunabilmektedir. Bu sayede, sistemin giriş-çıkışları arasındaki ilişkilerin çözümlenerek hata tespitinin yapılabilmesi hedeflenmiştir.

Bu tez çalışması 3 ana bölümden oluşmaktadır. Birinci bölümde, sıcaklık sensörleri için bir sensor validasyon tekniği tasarlanmıştır. İlgili metotta, SCADA sisteminden alınan 4 sıcaklık sensörünün ölçümleri kullanılarak herhangi birinde hata olup olmadığı tespit edilmeye çalışılmıştır. Bu sayede, sensörlerin yerlerinden alınarak kontrol edilmesi yerine sürekli durum izleme ile hata tespiti yapılması amaçlanmıştır. Öz-İlişkili ve Çok-Giriş-Tek-Çıkışlı YSA yapıları ile farklı başlangıç koşulları ve ağ mimarileri kullanılarak problemin çözülmesi sağlanmıştır. SCADA sistemi, sensor hatalarına dair bilgi içermediği için, sensörlerden birinde kalibrasyon kayması şeklinde bir hata yapay olarak modellenmiştir. Kalibrasyon kayması, yüksek ölçekte olmadığı sürece genel davranıştan çok farklı ölçümlere neden olmadığı için bu tip bir durum hata tespiti açısından zorlayıcı bir koşuldur. Önerilen sistemin etkinliğinin değerlendirilebilmesi ve kalibrasyon hatasının çevresel koşullardan kaynaklanabilecek gerçek sıcaklık değişiminden ayrıştırılabilmesi için, YSA modelleri eğitildikten sonra farklı koşullarda test edilmiştir. Öncelikle tüm ölçümlerin orijinal test veri setinden alındığı, ikinci durumda sensörlerden birinde kalibrasyon hatasının modellendiği, üçüncü durumda ise çevresel nedenlerden kaynaklı olabilecek şekilde tüm sensör ölçümlerinin değiştirildiği bir test yapısı kurulmuştur. Alınan sonuçlar, tasarlanan sistemin kalibrasyon hatasını tespit edebildiği ve bu hatadan kaynaklanan durumun çevresel koşullardan kaynaklanan sıcaklık değişiminden ayrıştırılabildiğini göstermiştir.

Tezin ikinci bölümünde, rüzgar türbininin genelinde oluşan hataların tespit edilebilmesi için tasarlanan YSA modellerinde kullanılmak üzere özellik oluşturma ve seçme yöntemleri uygulanmıştır. Bu tür ön işlemler, yapay zeka uygulamalarında oluşturulan modellerin hızlı ve yüksek başarımlı olarak çalışabilmesi için kullanılan yöntemlerdendir. Böylece, YSA girişlerine sistematik bir şekilde karar verilerek performansın iyileştirilmesi amaçlanmaktadır. Öncelikle, SCADA'dan toplanan ham verilerden çeşitli işlemlerle yeni özellikler oluşturulmuştur. Bu sayede, hatalar hakkında ham verilerden daha iyi bilgi verebilecek özellikler elde etmek amaçlanmıştır. Yeni veriler oluşturulurken, ham veriler arasındaki ilgili ölçümler arasındaki farklar, istatistiksel parametreler, zaman serisi özellikleri ve sistemin genel prensipleri ile ilgili bilgilerden yararlanılmıştır. Ham özellikler ve oluşturulan özellikler arasından hata tespiti problemi için kullanılabilecek etkili bir alt kümenin seçilmesi için çeşitli özellik seçme yöntemleri uygulanmıştır. Öncelikle, filtreleme yöntemleriyle tüm özellikler arasından ilk eleme yapılarak problemle yüksek derecede ilgisi bulunan özellikler belirlenmiştir. Filtreleme yöntemleri olarak Fischer ve Relief algoritmalarından yararlanılmıştır. Filtre yöntemleri ile elde edilen özellikler sarmal özellik seçme yöntemi ile bir kez daha değerlendirilerek, özellikler arasındaki karşılıklı ilişkiler incelenmiş ve uygun YSA girişlerine ulaşılmıştır. Elde edilen sonuçlar, özellik oluşturma ve seçme yöntemlerinin hata tespit performansı üzerinde olumlu etkileri olduğunu göstermiştir. Özellikle, SCADA verileri ile oluşturulan hata tespit sistemlerinde karşılaşılan önemli bir problem olan yüksek sayıda yanlış hata alarmının düşürülmesi konusunda yüksek başarım gözlenmiştir.

Uygulanan özellik oluşturma ve seçme yöntemleri ile, jeneratör ısınma hatası için 3 aylık test verisinde karşılaşılan yanlış hata alarm süresi 210 dakikadan 30 dakikaya düşürülmüştür.

Son aşamada, sistem genelinde 3 seviyeden oluşan bir hata sınıflandırması yaklaşımı tasarlanarak, rüzgar türbini genelinde hata tespiti, izolasyonu ve öngörüsü gerçekleştirilmiştir. SCADA sistemi sayesinde çeşitli alt sistemlere dair hata bilgisine ulaşılabilmektedir. Böylece, sensör validasyonu bölümünden farklı olarak bu bölümde hataların yapay olarak modellenmesi yerine gerçek hata verisi üzerinde çalışılmıştır. Bu tezde kullanılan rüzgar türbininde bir yıllık veri toplama süresince temel türbin bileşenlerinden herhangi birinin çok ciddi bir hasara maruz kalmadığı gözlenmiştir. Fakat, tüm rüzgar türbinlerinde olduğu gibi sık sık büyük sonuçlara neden olmadığı halde enerji üretiminin düşmesine ve türbin güvenilirliğinin azalmasına neden olan hatalar oluşmuştur. Bu tip hatalar önemli belirtiler vermediği için tespit edilmesi temel bileşenlerdeki büyük hatalardan daha zordur. Literatürde, aylar öncesinden tespit edilebilen temel bileşenlerdeki fatal hataların aksine, sık gerçekleşen hataların öngörü aralığının saatlerle sınırlı olduğu görülmektedir. Ayrıca, hata sınıflandırma problemlerinde sağlıklı ve hatalı veri setlerinin doğal olarak dengeli bir sayıda olmaması, veri setinin büyük oranda normal çalışmaya dair örneklerden oluşması da model başarımının düşmesine sebep olmaktadır. Bu duruma önlem olarak, özellik oluşturma ve seçme yöntemlerinin yanı sıra hata sınıfına ait yapay örnekler oluşturarak ve normal çalışma sınıfının örnek sayısı azaltılarak farklı eğitim setleriyle de eğitim gerçekleştirilmiştir. 3 seviyeli hata sınıflandırma sisteminin ilk seviyesi olan hata tespit aşamasında, türbinin normal veya hatalı bir durumda olup olmadığı tespit edilmeye çalışılmıştır. Hata izolasyonu seviyesinde, hatanın hangi alt sistemden kaynaklandığının tespit edilmesi hedeflenmiştir. Son olarak, hata öngörüsü seviyesinde ise oluşacak hatalar önceden tahmin edilmeye çalışılmıştır. Çeşitli sınıflandırma seviyelerinden oluşan bu yaklaşım sayesinde, operatörlere mevcut veya gelecekte oluşacak hatalarla ilgili bilgi verebilecek bir hata sınıflandırma sistemi oluşturulmuştur.

Elde edilen sonuçlar, oluşturulan sistemin her 3 seviyede de yüksek başarımlara sahip olduğunu göstermiştir. Böylece, SCADA verisinin dezavantajlarına rağmen, YSA modelleri ve yardımcı algoritmalar uygulanarak rüzgar türbinlerinde etkili bir şekilde hata tespiti, izolasyonu ve öngörüsü yapılabileceği görülmüştür. Önerilen sistem, rüzgar türbinlerinde akıllı bakım yöntemlerinin geliştirilmesi konusuna maliyet etkin çözümlere katkıda bulunmaktadır.

1. INTRODUCTION

Global energy demand has shown a sharp increase in the last decades and this trend is expected to continue. According to the International Energy Agency, total world energy demand is projected to expand by approximately 30% between today and 2040 which would be around twice as large without the ongoing improvements in energy efficiency [1]. The growth in the world population, growing world economy, increasing urbanization and changing energy consumption patterns are some of the reasons of the rise in energy demand.

Fossil fuels are still the leading primary energy sources to meet the world energy demand. They are advantageous in terms of their high calorific value, easy transportation, storage and globally well-developed technology. However, the share of them in the overall energy supplies should be reduced for various reasons. One of them is that fossil fuels are finite sources and they are depleting at a fast rate. Heavily relying on them would cause challenges in a global manner on meeting the energy demand in the future. Also, the negative impacts of fossil fuel consumption on environment is an important factor. United Nations Framework Convention on Climate Change (UNFCCC) agreed in 2012 to pursue actions in order to limit the global mean temperature change to below 2 ℃ compared with pre-industrial levels [2]. This target is perceived as a universally accepted goal as a safe limit. It was shown in a study that a significant part of oil, gas and coal reserves should be remained unused until 2050 for the average global temperature rise caused by greenhouse gas emissions not exceed 2 ℃ [3].

For fossil fuel importers such as Turkey, dependency to exporters cause additional risks. Turkey meets most of its energy demand by foreign resources. This causes a significant disadvantage as energy is a strategic commodity. The overdependency also brings economic vulnerability to market fluctuations as the economy is exposed to the volatility in oil and gas prices. Due to all these factors, reducing the share of fossil fuels in the energy mix is an essential requirement. Therefore, efforts towards diversification of energy supplies have been intensified globally.

Diversification of energy is the practice of using various energy sources, suppliers and transportation routes to reduce dependence on a single resource or provider. By diversifying its energy mix, a country can become able to insulate itself from energy disruptions and strengthen its energy security. Renewable energy sources carry significant opportunities especially in terms of diversification and for the reduction of greenhouse gas emissions. In addition, as renewable energy resources exist over wide geographical areas, local production of energy becomes possible which lower the dependency to fossil fuel exporters. Wind energy is one of the most promising options in this manner with the accelerating investments and technological developments.

Wind energy technology continues to improve rapidly, and energy conversion costs from wind installations continue to fall. In many countries wind power is now being deployed with good resources without any dedicated financial incentives from governments [4]. In the OECD countries, wind farms produced 6.4% of overall electricity and 25.5% of renewable electricity in 2017. Wind power capacity increased from 3.8 TWh to 696.9 TWh between 1990 and 2017, achieving an average annual growth rate of 21.2%. This is the second fastest growth rate of renewable electricity after solar photovoltaics [5]. The increase in the installed capacity is caused both by the new wind farm installations and the rising power capacity of individual wind turbines.

Both capacity and size of wind turbines have been increasing by virtue of developments in this field. Today, amongst commercially available wind turbines, the maximum rated power output capacity has reached the value of 9.5 MW [6]. These improvements result in a need for more developed strategies in wind farm operations. Moreover, wind farms are located in remote areas and they are usually operated in harsh environments which makes effective operations even more important. Wind turbine faults frequently lead to downtimes -the amount of time that equipment does not operating- in wind turbine operations, therefore result in a decrease in the amount of energy conversion. Unpredicted faults can also have detrimental effects on overall wind turbine and may contribute to decrease in systems lifetime. Fault detection and prediction is a significant part in wind turbine operations due to the increasing demand for higher performance in wind turbines as well as for increased safety and reliability requirements. Early diagnosis of faults can prevent their progression and reduce downtime durations which can contribute to reduce the operational costs and as a result unit electricity cost from wind turbines and increase safety, reliability and lifetime. Therefore, it is significant to clearly detect the current condition of the system and predict upcoming wind turbine faults as early as possible to take required actions and prevent destructive results. This thesis aims to contribute to the efforts in decreasing the costs and improving reliability and lifespan of wind turbines by designing models for overall fault detection, isolation and prediction of wind turbines using artificial intelligence methods applied to data collected from a wind turbine.

1.1 A General Look at the Maintenance Strategies

Maintenance of wind turbines are challenging due to various reasons such as their isolated locations, having several critical components working in vibratory environments and dependence of working conditions to multiple external variables. It is important to select an effective maintenance strategy considering all these factors.

In general, maintenance activities can be broadly classified in three groups namely; reactive maintenance, preventive maintenance and predictive maintenance.

- Reactive maintenance: Maintenance actions performed to return an equipment to proper working conditions from a faulty condition is considered as reactive maintenance. The unscheduled maintenance or repair of equipments/items are parts of this approach. Reactive maintenance is usually applied after an occurrence of a breakdown in system.
- Preventive maintenance: Maintenance that is regularly performed to lessen the likelihood of failing is preventive maintenance. These actions are carried out in a planned and periodic schedule to keep an equipment in working condition. These practices are precautionary steps to lower the probability of failures rather than correcting them after they occur. Regular inspection and replacement of critical components are examples of preventive maintenance actions [7].
- Predictive maintenance: Activities that focuses on finding out when equipment failures will occur and taking on actions before actual failures are predictive maintenance actions. In this approach, measurements and signal

processing methods are used to accurately diagnose state of equipment during operation. With a successful deployment of predictive maintenance techniques, maintenance frequency and costs would be minimized by preventing expenses associated with preventive maintenance and detrimental results that would be faced by reactive maintenance.

Today, majority of maintenance actions in wind farms are based on reactive and preventive approaches. Wind turbines are generally purchased with all-in-service contracts which include scheduled and unplanned maintenance actions, as well as periodic replacements and inspections. However, this approach is inadequent to meet the current demands of wind energy industry. Besides, the developments in the design of predictive algorithms and data-driven models make it possible to employ new strategies to maintenance problems. By benefiting from the modern data-processing and data acquisition systems it is possible to improve capabilities in this field. As a result, predictive maintenance of wind energy systems has been gaining increasing attention from researchers and wind energy industry. This thesis also proposes a predictive maintenance approach for wind turbines.

1.2 Data Acquisition in Wind Turbines

The types, characteristics and quality of data to be used in the design of condition monitoring and fault detection systems is one of the factors that define the performance of the system. Data collection in wind turbines can be classified in two methods. The first method is to use sensors which are specifically mounted for fault detection purposes. The second method is to use data collected from Supervisory Control and Data Acquisition Systems (SCADA).

In the first approach, depending on the characteristics of the components to be monitored, various sensors are mounted on different wind turbine components. Common measurement types for the purpose-built data collection method are vibration analysis, acoustic emission analysis, ultrasonic testing, oil particle and oil quality monitoring, analysis of converter sensor measurements, strain, torque, and bending moment sensors. The main advantage of using purpose-built sensors is, because they are selected and mounted for component-specific aims, flexible choices can be made considering the distinct requirements of target components which enables designers to collect highly useful data. Whereas, this approach causes extra costs which becomes an obstacle to developing a cost-effective solution.

SCADA system is a built-in part in most modern wind turbines. Many types of data are recorded by SCADA sytems such as temperature of various components, amount of energy production, operational data like rotational speed and power output and status data supplying information on state of wind turbine. It serves as the primary interface between the wind farm operator and individual wind turbines. It also allows remote and local control of basic wind turbine functions and collects data on the operational and environmental parameters to be used to analyse operations performance. Using data gathered from SCADA system for fault detection performance is advantageous as it provides information on overall wind turbine properties and no additional hardware costs are required as it is a built-in system. However, SCADA systems were not initially designed for fault detection purposes. Therefore, the sampling period of these systems is generally 10 min which is lower than desired for fault detection aims. Such a low data frequency causes difficulties due to the loss of noise characteristics which may carry important information on upcoming fault occurrences. Moreover, imperfections and missing values in data are common in SCADA data collection systems. Main challenges to use SCADA for fault detection aims are low frequency, late indication of fault statuses, high rate of false alarms. In spite of these challenges, as it includes a wide variety of data and is a cost-effective approach, using SCADA data carries many opportunities. To overcome the imperfections of data characteristics, suitable analysis and prediction algorithms should be employed to benefit from SCADA systems for fault detection aims.

By using historical data collected from wind farms operating in different sites and in diverse environmental conditions, more developed and generalized algorithms can be designed. However, one of the challenges on this aspect is that currently, the availability of wind farm data is very limited and publicly available wind farm data which include information on faults do not exist. The studies in the literature are designed and tested in different data sets, therefore, it is hard to compare the results as the complexity of problem and the details and quality of information is subject to change. The main reason of this issue is that currently wind turbine industry is not very open to data share. Researchers typically obtain wind farm data by non-disclosure agreements so it would not be possible to share them. As stated in [8], to produce more clean energy in a lower price, it is advisable to create data sharing platforms in wind turbine industry. By providing a better collaboration between wind energy industry and research community, energy production can increase by at least 10% and wind farm maintenance costs can be decreased developing data-driven health monitoring systems by 10% [8].

1.3 Model-Based and Data-Driven Fault Detection Strategies

Fault detection algorithms can be designed in different ways such as employing model-based or data-driven algorithms. In model-based fault detection strategy, firstly a mathematical model expressing the normal operation conditions of real system is created. The outputs produced by this model belong to given inputs are compared to the real measurements from wind turbine sensors. An alarm flag expressing a faulty condition is raised by analyzing and comparing the outputs of the mathematical model and the real wind turbine. Model-based algorithms are advantageous from the aspect of not requiring high frequency data. However, the success of this approach is highly dependent to the consistency of the mathematical model and the real behavior of the system. One of the main challenges of this approach is that wind turbines are very complicated dynamic systems, moreover they have complex control parts. Therefore, it is hard to obtain a reasonable mathematical model of the overall system.

In data-driven fault detection methods, unlike model-based algorithms, an explicit mathematical model of describing the system behavior is not required. They are designed based on processing the historical data of certain parameters and the regarding situation of the physical system. As the complexity of the real system increases, obtaining an accurate mathematical model becomes harder. Therefore, data-driven methods become an advantageous approach depending on the data availability. With the recent advances in intelligent methods, data-driven fault detection approach gained increasing attention. In this thesis, a data-driven fault prediction algorithm is proposed.

1.4 Wind Turbine Fault Detection Using SCADA Data

There are various approaches in the literature that benefit from SCADA data for the detection of wind turbine faults. These methods can be classified as trending, clustering, normal behavior modelling, damage modelling and assessment of alarms and expert systems [9].

Trending is one of the basic methods to determine if there is an anomaly in the data. It is based on gathering data for a time period and monitor how they change over time. Feng et al. analysed the relation between the gearbox efficiency and gearbox temperature increase by trending approach [9-10] [10], [11]. It was indicated that a change in the gearbox temperature is visible 6 months before a catastrophic gearbox failure. Yang et al. developed a trending method using bin averaging of wind speed, power output and generator speed [12]. They used a correlation method for the present and historical data to detect faults in two different cases which are a generator and a blade failure. Astolfi et al. analysed temperature trends depending on the rated power which helped operators to detect problems [13]. The main difficulty of the trending approach is that, a change in trends does not guarantee an incipient fault in the system. Therefore, the number of false alarms can exceed acceptable limits for the real-world applications.

Instead of visual interpretation of faulty trends, automatic classification of fault states can be developed by clustering. The advantage of clustering algorithms over trending methods is that it can provide information on distinct conditions which different turbines operate [9]. Kusiak and Zhang used vibration data to develop k-means clustering algorithm based on wind speed [13-14]. However, after acknowledging the limitations in determining the boundaries of clusters, they chose normal behavior models over clustering method. Catmull [16] and Kim et al. [17] used self organizing maps for clustering data to find out abnormalities. Catmull used normal behavior data as the training set and general ability to detect abnormalities were shown in a sensor error, reactive power loss and an unidentified generator failure. Kim et. al. showed a general ability to detect failures. Their method was able to assign subsequent wind turbine (WT) failures to corresponding clusters. Wilkinson et al. [18] also used a similar technique and presented some examples of detecting gearbox failures. Similar to the case in trending algorithms, interpretation of results is difficult in clustering algorithms as defining boundaries is challenging in real practices.

Majority of studies in wind turbine fault detection using SCADA data are based on normal behavior models. The main idea of this approach is to obtain a reference model of the real system in normal operating conditions to use it for detecting the possible faulty instances in the future data. A deviation which is higher than pre-determined limits between the reference model and real system would indicate there might be a fault in the target component. Various techniques to design normal behavior models were proposed in the former studies. The simplest approach for this aim is to use linear and polynomial models. Garlick et. al. used Auto-Regressive with eXogenous (ARX) input models to detect generator bearing failures using generator temperature measurements [19]. Cross and Ma also used ARX models [20] to analyze the normal behavior of generator and gearbox temperatures and detected some abnormalities in faulty states. Wilkinson et. al. designed a normal behavior model by full signal reconstruction (FSRC) method for drive train temperatures and tested their method on five different wind farms with a total 472 wind turbine years of data and succeeded to detect 24 out of 36 failures [18]. Schlechtingen et. al. also used linear FSRC approach to obtain a model for generator bearing temperature and detected a fatal generator fault 25 days prior to the damage [21]. As wind turbines are highly non-linear systems, modelling their behavior using non-linear models also carry significant capacity for successful applications. Artificial Neural Networks (ANN) were intensely used for this aim. Zaher et. al. developed an ANN based gearbox temperature model using 2 years of SCADA data and succeded to detect overheating problems 6 months in advance of a fault [22]. Various other studies also showed the success of normal behavior models by nonlinear methods in the detection of severe faults [13, 22–24].

Another strategy to detect faults by SCADA data is to build damage models. Instead of using a normal behavior model that is obtained as a ‘black-box’ in most of the normal behavior models, in damage modeling principle, the theoretical characteristics of failures are investigated to find out how systems react in failure modes. Breteler et. al. worked on the detection of a gearbox failure and reached large differences between the normal and faulty modes however the difference values were also large between different turbines [26]. Borchersen and Kinnaert also used damage modelling approach by designing a mathematical model using Extended Kalman Filter approach and proved the success by detecting 16 out of 18 faults in a test set with 3 years of SCADA data from 43 wind turbines [27].

Associating alarm or status information from SCADA system to fault situations is another method used for the fault detection purpose. Chen et al. trained ANN models to map from alarm patterns to detect faults, however the obtained accuracy rate was 8-47% [28]. They also used a probabilistic approach and proposed a Bayesian network to find root causes of faults and showed the feasibility to reason root causes in the presence of uncertainty [29]. Kusiak and Li predicted status codes by different machine learning methods and succeeded to predict non-fatal faults 60 min in advance [30]. Leahy et. al. also investigated detection of frequent faults by analyzing SCADA statuses and obtained high-accuracy values using support vector machines [31]–[33]. Li et. al. used Gaussian process classifiers to analyze status codes and predicted faults 30-min before they occur [34].

These studies show that the performance of fault prediction models using SCADA data depends highly on the type of the failure in terms of severity. It is possible to detect catastrophic faults of main wind turbine components by processing SCADA data months in advance. For instance, Zhang and Wang detected the initial indications of a main bearing fault 3 months in advance with a normal behavior model using ANN [25]. The overheating problem that indicates an upcoming fault 6 months prior to the real fault by Zaher et al. also focused on a severe gearbox failure [22]. Similarly, Godwin and Matthews detected prognostic signatures up to 5 months before a catastrophic gearbox fault occurred [35]. These kinds of catastrophic failures occur rarely. For example, in [35] SCADA data from 6 wind turbines for 28 months were collected and only one fatal gearbox failure happened. However less serious faults occur frequently in all wind turbines and they cause a reduction in power production and degrade performance and life expectancy of turbines. They naturally present less indications which makes it harder to predict them accurately in advance. Former research results show potential for prediction of frequent faults. In the method proposed by Kusiak et al. three-level fault prediction system was developed which includes detection of the existence and category of faults and prediction of faults in advance [30]. They used SCADA data with a sampling period of 1 second, built various data driven methods and managed to predict faults 5-60 min before they occur. Leahy et al. used 10 min SCADA data [36] which is the generally available sampling period in industrial applications and using support vector machines, they obtained high recall values between 1 and 12 hours before generator heating or excitation faults occur. Although comparing to the results on fatal faults, accuracy rate for detection is small and there are high amount of false alarms, these studies show the potential success of the use of SCADA data not only for fatal but also for non-severe faults that are harder to detect which is a challenging but beneficial task for the reliability and cost effectiveness of wind turbines.

1.5 Purpose of the Thesis and Contributions

The main purpose of this thesis is to design an overall fault detection system for wind turbines using SCADA data which is available as a part of the built-in components in most of the modern wind turbines. It was aimed to make contribution on the detection performance of frequent and non-fatal wind turbine faults which occur in every wind turbine and do not cause fatalities, however, severely reduce the system availability and performance. Due to the nature of this type of faults, they do not show as strong indications as fatal faults do and the quality and sampling frequency of SCADA data are not ideal therefore advanced models are required which was provided by Artificial Neural Networks in this thesis. This non-intrusive method brings major advantages as it does not require any additional hardware costs. The study was held for the incipient faults which do not occur abruptly but happen gradually with former indications.

One of the subgoals of the thesis is to ensure the validity of sensor measurements and detect if there is a calibration error in any of the sensors being evaluated. This was realized by solely using the temperature measurements from various parts of the turbine. A regression-based model structure was developed for this aim.

The main goal is to design a system for the overall wind turbine that determines if there is a fault or not, decides the type of the fault and predicts upcoming faults in advance by the assessment of fault statuses. For this aim, classification models that discriminate between the faulty and normal statuses were designed.

The main contributions are listed as follows;

- A sensor validation method was proposed for temperature sensor measurements and it was presented that the applied technique was effective for this purpose. A fault in one of the sensors in form of a calibration drift was detected by the designed model.

- A major advantage of this work is the effective use of SCADA data, which does not bring any additional hardware costs as it is a built-in part in most modern wind turbines.

- For the plant-wide fault detection purpose, it was shown that ANN are powerful for the analysis of wind turbine SCADA data on detecting non-fatal faults which do not show indications as strong as fatal faults of main components do.

- In addition to detect if a fault exists, the exact subsystem with faulty behavior was attempted to be found and high-performance results were also obtained in this part of the thesis.

- Generator heating faults were predicted in advance as early as 56 hours before they occur which is a highly effective result that significantly improves the current prediction horizon in the literature for this type of faults.

- Improvements in the classification performance were accomplished by applying systematic feature construction and selection methods.

- The data set naturally contains unbalanced data in terms of the amount of faulty and normal operations. The training performances are negatively affected by this characteristic. To overcome this problem, oversampling and undersampling methods were applied and proven to be effective on results.

- The overall high success rates in the fault prediction level would be beneficial for increasing the amount of energy conversion in wind turbines by informing operators about upcoming faults to enable them take necessary precautions.

The rest of the thesis is organized as follows. Chapter 2 provides the preliminary information on the research including general information and main subsystems of wind turbines, the details of SCADA data used in this thesis and background information on Artificial Neural Networks. Chapter 3 presents the sensor validation problem for the temperature sensors. A simulated calibration fault is presented and detected in the scope of this part. Chapter 4 provides methods applied on feature construction and selection to select the inputs of the ANN in a systematic manner. In Chapter 5, a three-level fault classification method which includes the detection, isolation and prediction of wind turbine faults is presented. Finally, Chapter 6 provides the conclusion part with the results and the possible future research directions.

2. BACKGROUND

This chapter provides background information on the thesis research. General information on wind turbines and their main components are given in Chapter 2.1. In Chapter 2.2, the details of the data collected from the SCADA system are described. The types, limitations and sample segments of the data are presented. Chapter 2.3 firstly presents general information on Artificial Neural Networks. In addition, the ANN types used in the scope of this research are described in more detail.

2.1 General Information on Wind Turbines

Modern wind turbines are structurally classified into two categories as horizontal axis wind turbines (HAWT) and vertical axis wind turbines (VAWT). This classification is based on the orientation of the rotation axis. Horizontal axis means that the rotating part of the turbine is parallel with the ground, whereas in vertical axis turbines, it is perpendicular to the ground. Today, high-capacity wind turbines used in industrial applications are of HAWT class due to their higher energy conversion efficiency, straightforward design configuration, higher structural integrity, and improved dynamic stability under strong wind conditions. VAWTs are only used for experimental aims or in small scale residental applications. In this thesis, the interest is the horizontal-axis 3 bladed wind turbines and the term “wind turbine” refers to these kind of turbines. Figure 2.1 shows some main components of horizontal axis wind turbines.

With the increasing importance of effective wind turbine operations, the amount of works in this field rapidly increased which resulted in differentiations in the subsystem and fault representations. This non-uniform data treatmant became a challenge in the comparison of different studies, therefore the requirement for a clear and uniform taxonomy has emerged. Reder et. al. proposed a uniform taxonomy [37] which is convenient for the representation of both the modern and historical wind turbine data by modernising the existing taxonomy.

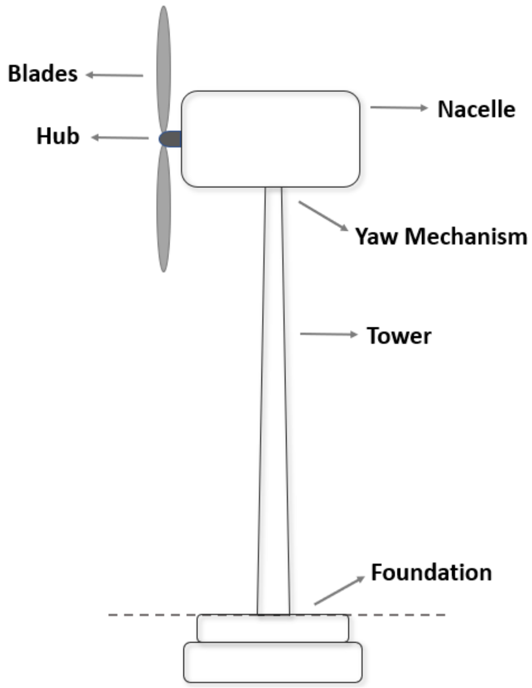


Figure 2.1 : Main parts of HAWTs.

The components and their sub-assemblies were classified based on their physical location and functionality. By this approach, wind turbine system has been divided into 7 main subsystems and several assemblies were assigned to each subsystem. The subsystems are; the power module, rotor and blades, control and communications, nacelle, drive train, auxiliary system and structure.

Power module consists of generator, converter, transformer and aiding components regarding to the power conversion process. Generally, most of these components are located in nacelle. However, in some novel MW scale wind turbines including the turbine used in this thesis transformer lies on ground level at the bottom of the tower.

Rotor & blades are the rotating parts of the turbine which face the wind and transmit wind's kinetic energy to the power module as mechanical rotation. In many wind turbines, also a pitch mechanism exists by which the angle of blades can be changed regarding to the wind speed in order to optimize the energy conversion.

Control & communications subsystem is responsible of the automatic operation and data collection parts of the system. Various types of sensors and SCADA system are also considered as a part of this assembly.

Main part of the drive train subsystem is gearbox that is responsible for connecting the low-speed shaft attached to the turbine blades to the high-speed shaft attached to the generator. Assisted by a series of gears of varying sizes, the gearbox converts the slow rotation of the outer blades to faster rates that is needed by the generator to begin energy conversion.

Nacelle subsystem is located on top of the tower and provides a protection for the components mounted in it. In MW-scale wind turbines there is a yaw system which is also considered as part of the nacelle subsystem. It changes the orientation of the nacelle and rotor to adjust them to face the wind correctly. Yaw system is comprised of bearings, gears, brakes, and and engine.

Main components are supported by auxiliary subystem which consists of assemblies that support the main operations of the turbine such as meteorological station, cooling system and lightning protection. Finally, structure subsystem is comprised of tower and foundations assemblies. Table 2.1 shows these subsystems and the assemblies for each of them.

Table 2.1 : Main subsystems and assemblies of wind turbines [37].

Power Module	Frequency converter	**Control & Comm.**	Sensors	**Auxiliary Systems**	Cooling System
	Generator		Controller		Electrical Protection
	Switch Gear		Communication System		Human Safety
	Soft starter		Emergency		Hydraulic Group
	MV/LV Transformer[1]		Control&Comm. Series		WT Meteorological St.
	Power Feeder Cables				Lightning Protection
	Power Cabinet	**Nacel**	Yaw System		Firefighting System
	Power Module Other[2]		Nacelle Cover		Cabinets
	Power Protection Unit		Nacelle Bed Plate		Service Crane
					Lift
Rotor & Blades	Pitch System	**Drive Train**	Gearbox		Grounding
	Blade Brake		Main Bearing		Beacon / Lights
	Rotor		Bearings		Power Supply
	Blades		Mechanial Brake		Electrical Aux. Cooling
	Hub		High Speed Shaft		
	Blade Bearings		Silent Blocks	**St.**	Tower
			Low Speed (Main) Shaft		Foundation

[1] Medium voltage to low voltage transformer

[2] Aiding units in power module

2.2 Data Characteristics

Data used in this research were collected from a 900 kW onshore wind turbine located on the north of Turkey. Similar to most wind turbine SCADA systems the sampling period is 10 min. The data were recorded in the 12 months period from

01.01.2015 to 31.12.2015. The producer, exact location and some additional details of the turbine are not explicitly specified due to the non-disclosure agreements signed with the wind turbine company.

The data set consists of various types of information which are; wind parameters, temperature values, operational data and status data. As presented in a former review on the use of wind turbine SCADA data, the types of data may vary significantly in different turbines [9]. The main measurements like produced power and rotation rate are available in all SCADA systems however the availability of more detailed measurements differ based on the system. Some parameters typically recorded in SCADA systems are absent in our data set such as electrical characteristics like generator voltage and phase values and control variables like pitch angle, fan status, cooling pump status etc. In the following parts, the characteristics of the data used in this thesis are described.

2.2.1 Wind parameters

Monitoring wind parameters is an essential part of wind turbine control and data collection systems as wind information is highly useful in evaluating the efficiency of power production and the instantenous operational statuses. The available information regarding to wind characteristics for each 10 min interval are presented in Table 2.2.

Table 2.2 : Wind parameters.

Data type
Minimum wind speed
Maximum wind speed
Average wind speed

2.2.2 Temperature data

The data set also contains temperature values of various components. Temperature recordings represent the 10 min averaged values for each time interval. The locations of temperature sensors mounted on different parts of the turbine are presented in Table 2.3. Different wind turbine SCADA systems may also contain measurements of blade temperatures, yaw control cabinet temperature, ambient temperature which would be beneficial for the improvement of fault prediction performance, however in our case these measurements do not exist.

Table 2.3 : Temperature data.

Location of Temperature Sensors
Generator stator
Generator rotor
Nacelle box
Front hub bearing
Rear hub bearing
Nacelle control cabinet
Control cabinet
Tower
Transformer

2.2.3 Operational data

As SCADA systems originally designed for continuous monitoring of wind turbine operations, many operational features are available. Similar to the wind parameters and temperature values, operational data also have 10 min sampling period. Details of the operational data available for this work are presented in Table 2.4.

Table 2.4 : Operational data.

Data Type	Detail
Rotation speed	Min, Average, Max
Power output	Min, Average, Max
Energy output	Total, Diff
Nacelle direction	Average

2.2.4 Status data

The last category of data recorded by the SCADA system is the status data. Status data describe the existing condition of the turbine. They include a main code, an additional code and a status description. Main code defines the general situation whereas additional code gives details on the cause of the main status. Table 2.5 shows a part of the status data used in this thesis.

Unlike other types of information represented before, data update interval for statuses is not 10 min. Instead, a new code only appears when the status of the turbine changes. A change can be caused by external situations such as a turbine stall due to low wind speed or internal situations such as a failure in one of the components. Total number of status data is much lower than other data classes. To be specific, in our case there are more than 50000 instances of wind characteristics, temperature and operational data. Whereas, there are approximately 2800 instances of status data. To match status data with other data types, a status for each 10-min time step is assigned by repeating the existing status until a new status appears. If multiple statuses occur in the same 10-min interval, the main reason of the turbine condition was tried to be determined. For example, on 6/10/2015 the turbine stops operating at 2.45 due to low wind speed value. At 4.04 wind speed becomes high enough again so the turbine attempts to operate, however due to generator heating, it stays in the stall mode. There are more than 1 statuses in the same 10-min interval from 4:00:00 to 4:10:00, however "Generator heating" label was selected over "Turbine starting" as it is the main reason of the changing situation of the turbine.

Table 2.5 : Status data.

Day	Time	Main status	Additional status	Status Text	Duration
6/10/2015	2:45:10	2	1	Lack of wind : Wind speed too low	01:19:00
6/10/2015	4:04:10	0	1	Turbine starting	00:00:28
6/10/2015	4:04:38	9	1	Generator heating : Isometer	15:54:17
6/10/2015	19:58:55	2	1	Lack of wind : Wind speed too low	00:38:09
6/10/2015	20:37:04	0	1	Turbine starting	00:00:30
6/10/2015	20.37:34	9	1	Generator heating : Isometer	06:00:49
6/11/2015	2:38:23	0	2	Turbine operational	00:01:40
6/11/2015	2:40:03	0	1	Turbine starting	00:01:38
6/11/2015	2:41:41	0	0	Turbine in operation	73:27:14

2.3 Artificial Neural Networks

Artificial Neural Networks (ANN) are computational models inspired by biological neural networks with particular properties such as the ability to adapt or learn, to generalise or to cluster and organise data [38]. Strengths of ANN models on complex problems that are hard or impossible to solve by mathematical modelling is rooted by their main characteristics which are parallel computing, learning and generalization. The use of ANN offers many capabilities and properties such as; nonlinearity, adaptivity, fault tolerance, evidential response and contextual information [39]. A large number of ANN architectures were proposed for different problems and due to the advancements in the computational facilities, both software and hardware, ANN are increasingly implemented in various areas.

The architecture of an ANN determines how its computational units are connected and how the input information is processed. Although, many different structures were proposed in the literature, the most common type consists of three main parts known as layers which are; input layer, hidden layer or layers and output layer. Input layer receives information from the external environment. It consists of nodes which are not computational units but are responsible of transmission of information to the next parts of the network. Hidden layer includes neurons which are responsible of the internal processing of the network by their activation functions. Output layer is also composed of neurons which are responsible of processing the information obtained from former parts of the network and producing the final output. More detailed explanations on this topic can be found in [39-40].

The main architectures of ANN in terms of how their layers are arranged and interconnected can be classified as feedforward and recurrent neural networks. Early feedforward ANN were single layer networks where input layer nodes projects directly to output layer neurons [41]. Single layer describes the computational output layer as input nodes are not processing units. The limitations of single layer ANN resulted in the development of multilayer feedforward ANN in which, one or more hidden layers of neurons are used in addition to input and output layers. Many feedforward ANN are proposed such as Adaline and Madaline Networks [42-43], Multilayer Perceptron Neural Networks (MLP) [44], Multilayer Feedforward Neural Networks (MFNN) [39], Probabilistic Networks [45], Radial Basis Function Networks (RBFNN) [46], Generalized Regression Neural Networks (GRNN) [47] and Self-Organizing Feature Maps [48].

In recurrent ANNs, there is at least one layer works as a feedback loop. Therefore, information also flows from outputs to inputs. Some common types of recurrent neural networks can be listed as; Hopfield Networks [49], Elman Networks [50], Jordan Networks [51], and Bi-Directional Associative Memory Networks [52], Adaptive Resonance Theory Networks [53], Long Short Time Memory Networks [54] and Echo-State Networks [55].

In artificial intelligence applications, the selection of model should be realized considering the requirements of implementation. For example, for time dependent systems, recurrent ANNs offer possible successful results. They are effective for tasks where inputs and outputs are both sequences such as speech recognition, speech synthesis, named-entity recognition, language modelling, and machine translation [56]. As proven in [57] MFNN are universal approximators for function approximation. Some of the areas they are commonly used is regression and classification problems. In this thesis, the sensor validation problem was handled as regression and the overall fault detection task was approached as a classification problem therefore, MFNN is one of the ANN types attempted in the scope of this work. RBFNN and GRNN which are also well-suited for these kind of tasks are the other types used in this thesis. In the next parts of this chapter, these ANN types are described.

2.3.1 Multilayer feedforward neural network

Multilayer Feedforward Neural Network (MFNN) is the first type of networks used in this thesis. There are three basic characteristics which are common in all MFNN networks. First, they contain one or more hidden layers. Second, each neuron includes an activation function. Lastly, there is a high degree of connectivity set up by synaptic weights between network elements [39].

The hidden neurons play an important role in the network. They perform a nonlinear transformation from input space to feature space. By this way, the patterns in the data become more seperable. Nonlinearity property of MFNN which is one of the most important characteristics of them is provided by the use of nonlinear activation functions. Some commonly used activation functions are logarithmic sigmoid, tangent sigmoid, softmax functions.

Backpropagation algorithm (BP) is the classical method in training MFNN neural networks. Training by BP involves two phases which are the forward phase and the backward phase. In the forward phase, the input signal is transmitted through the network layer by layer and the outputs are calculated. This phase finishes with the computation of an error signal. In the backward phase, the calculated error value is propogated in the backward direction through the input of the model. The adjustments for the network parameters are performed in this phase.

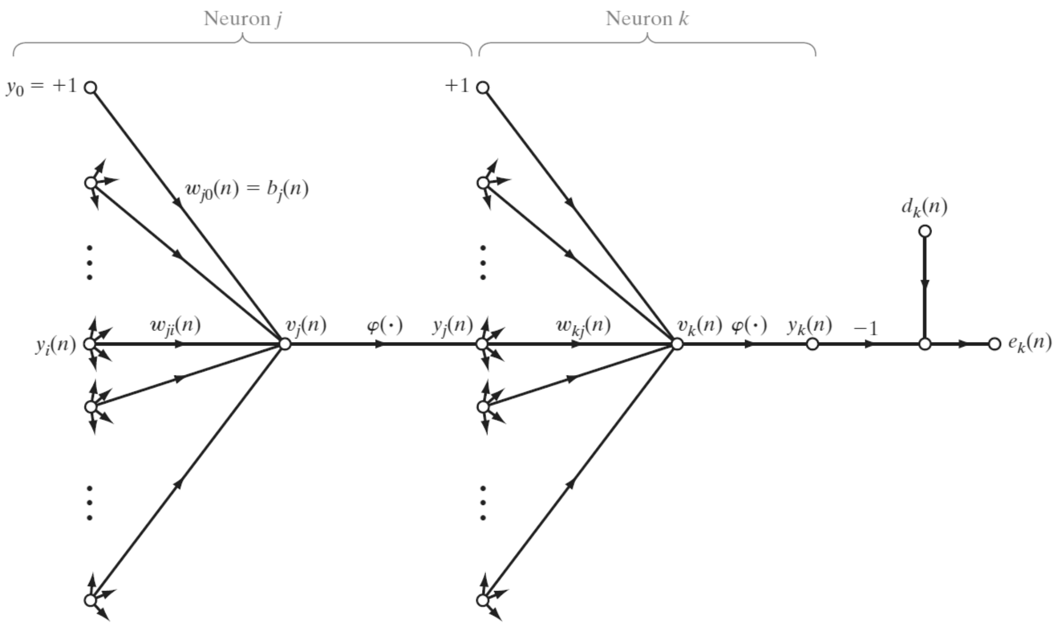


Figure 2.2 : Signal flow graph of MFNN networks [39].

Figure 2.2 shows the signal flow graph of a MFNN network for two neurons *j* and *k* from consecutive layers. Where, *w(n)* and *b(n)* are the weight and bias values for iteration *n*, *v(n)* is the induced local field which is the input for activation functions, *d(n)* is the desired outputs, *y(n)* is the output for each neuron and *φ(.)* are the activation functions. The computations start by selecting initial values for adaptive parameters like synaptic weights and bias values. This step is followed by feeding the network with an input set from the training data. The input signal is propagated through the network. In this forward phase, the induced local field for neuron *j* in layer *l* is computed as;

$$v_j^{(l)}(n) = \sum_i w_{ji}^{(l)}(n) y_i^{(l-1)}(n) \tag{2.1}$$

Where, $y_i^{(l-1)}(n)$ is the output of neuron *i* in the previous layer $l-1$ at iteration *n*, $w_{ji}^{(l)}$ is the weight of neuron *j* in layer *l* fed from neuron *i*. And the output signal for neuron *j* in layer *l* is;

$$y_j^{(l)} = \varphi_j(v_j(n)) \tag{2.2}$$

If neuron *j* is the output neuron than

$$y_j^{(l)} = o_j(n) \tag{2.3}$$

And the error between the desired and computed outputs are;

$$e_j(n) = d_j(n) - o_j(n) \tag{2.4}$$

After the computation of error signal, backward phase starts for the update of adaptive parameters. Firstly local gradients of the network are computed by;

$$\delta_j^l(n) = \begin{cases} e_j^{(L)}(n)\varphi_j'\left(v_j^{(L)}(n)\right), \textit{for output layer L} \\ \varphi_j'(v_j^{(L)}(n)) \sum_k \delta_k^{(l+1)}(n) w_{kj}^{(l+1)}(n), \textit{for hidden layer l} \end{cases} \tag{2.5}$$

Where, φ_j' is the differentiation with respect to the argument. If the neuron *j* is in the output layer, the layer was referred as *L*. As the last step, synaptic weights are adjusted according to the generalized delta rule [39].

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha[\Delta w_{ji}^{(l)}(n-1) + \eta \delta_j^{(l)}(n) y_i^{(l-1)}(n)] \tag{2.6}$$

Forward and backward computation phases are repeated with each new input set until the chosen success criteria is met. The procedure explained in this part is online learning in which input data is fed into the network sequentially. In this thesis online learning was applied. Another alternative is the batch learning where the update is not realized after the introduction of every new input but the weight and bias deltas are accumulated to aggregate set of deltas and they are applied to each weight and bias.

The basic backpropagation algorithm uses the steepest descent approach to minimize errors. To provide a faster algorithm for practical applications, many variations of it were proposed where the derivatives are also processed from the last layer of the network to the first [58]. They differ from the classical approach in the way of how the resulting derivatives are used to update weights. Levenberg-Marquardt [59] algorithm is one of them which was used as the training method of MFNN models in this thesis. It is commonly used as one of the main choices in practical applications due to its general fast and stable results.

2.3.2 Radial basis function neural networks

RBFNN is a type of ANN typically having a single layer of hidden units that are connected to linear output units. In the classical RBFNN approach, the size of the units in the hidden layer is the same as the size of the training vector [46].

RBFNN uses radial basis functions as transfer functions and also differ from MFFNN in terms of two basic principles. First, instead of performing an inner product operation between the weight and the input, the distance between the input and the rows of the weight matrix are calculated. Each row in the weight matrix is often called as the center of the corresponding neuron (radial basis function). Secondly, instead of adding a bias, in this case it is multiplied by a width factor (also referred as spread term) [58]. Width factor performs a scaling operation to the radial basis function and causes it to stretch or compress.

Output of a neuron of RBFNN is given in Eq. 2.7.

$$y = \sum_{k=1}^{N} w_k \varphi_k(\|x - c_k\|) \tag{2.7}$$

Where, N is the number of neurons in hidden layer, φ_k is the kernel of radial basis function for each unit. c_k is the center of radial basis function vector for neuron k, w_k is the weight of neuron k in the output layer.

Various radial basis functions can be used as kernel such as, gaussian function, multi-quadratic function, thin plate spline function, cubic function. The most common choice for RBFNN kernel is Gaussian activation function which was also used in this thesis. It is presented in Eq. 2.8.

$$\varphi_k(\|x - c_k\|) = exp\left(-\frac{\|x - c_k\|^2}{2\sigma_k^2}\right) \tag{2.8}$$

Where $\|x - c_k\|^2$ is the squared Euclidian distance between the associated center vector and the input vector. And σ_k is the width factor of the k^{th} hidden unit in the hidden layer which controls the smoothness properties of interpolating function. The center values determine the position, whereas standard deviation determines the widtf of the gaussian function which is the width factor. By changing these parameters, different trials can be made based on the requirements of applications.

2.3.3 Generalized regression neural networks

GRNN is a type of feedforward neural network that is originated from the theory of statistical estimation. It is mainly based on a nonparametric regression of the variable *y* on the independent variable *x*. In this approach, a specific functional form to describe the relation between inputs and outputs is not required, instead the appropriate form is expressed as a probability density function which is determined from observed data. By this way, the most probable value of output *y* is calculated given the training vector *x*. Since the parameters are directly calculated using examples, an iterative computation is not required [47].

The estimated output value is calculated by its conditional expectation which is given in Eq. 2.9.

$$\hat{y}(x) = \frac{\sum_{k=1}^{n} y_k \varphi_k(x, x_{k,}\sigma)}{\sum_{k=1}^{n} \varphi_k(x, x_{k,}\sigma)} \tag{2.9}$$

Where, $\hat{y}$ is the estimate of the output which is a weighted average of all the observed samples y_k, k is the number of sample observations, x is the input vector, x_k are the training samples, σ is the width factor and φ_k is the kernel function

In this theis, similar as RBFNN models, gaussian kernel function was applied. Therefore, φ is;

$$\varphi(x, x_k) = exp\left(-\frac{D_k^2}{2\sigma^2}\right) \tag{2.10}$$

Where, D_k^2 is the k^{th} squared distance between the training samples used to calculate the probable values which is given Eq. 2.9.

$$D_k^2 = (x - x_k)^T (x - x_k) \tag{2.11}$$

Eq 2.12 gives the resulting output estimate

$$\hat{y}(x) = \frac{\sum_{k=1}^{n} y_k exp\left(-\frac{D_k^2}{2\sigma^2}\right)}{\sum_{k=1}^{n}\left(-\frac{D_k^2}{2\sigma^2}\right)} \tag{2.12}$$

IN GRNNs, centers and heights can be directly determined from the data without training. Therefore, it belongs to the class of memory-based networks, in which the operation highly dependes on the storage of data. Memory-based concepts are different from optimization-based concepts such as MFNN which are characterized by high optimization efforts. Due to this lack-of optimization, GRNNs can be inefficient in noise attenuation however, they are trained much faster than RBFNN models [60].

3. SENSOR VALIDATION

Data collected from wind turbine sensors are used for various aims such as control systems design, plant-wide condition monitoring, interconnection and communication within wind farm. Ensuring the validity of sensor measurements is essential for healthy operation of all these subsystems.

The reliability and safety of complex systems are highly dependent on the reliability of sensors used. Measurement errors may cause a degraded performance and reduction in the power output of wind turbines. Diagnosis of faulty sensors in wind turbines can enable required actions to be taken, such as rescheduling maintenance, reconfiguration of corrupted control loops or initialization of emergency shutdown operations [61]. Therefore, fault detection and calibration of sensors is an essential part of the overall condition monitoring of wind turbines.

Currently, wind turbine sensor calibration is generally performed during scheduled maintenance actions. Instead of diagnosing sensor faults by taking the system offline periodically, it would be more feasible and practical to detect such faults during the online monitoring of sensor readings.

A sensor validation technique applied to wind turbine temperature sensors is presented in this part of the thesis. Considering the low-frequency outputs of SCADA measurements, nonlinear dynamics of the system and imperfections in data such as high rate of missing values, Artificial Neural Networks were selected to find out possible drifts in sensor measurements.

Effective implementation of ANN for sensor validation purpose in many fields have been reported in former studies. Kramer introduced Auto-Associative ANN architectures for sensor validation in temperature data for a simulated distillation column [62]. Various research results proved the power of ANNs in sensor validation in nuclear power plants [63–65], chemical processes [66], gas turbines [67] and turbofan engines [68].

Although ANN are widely used for fault detection of many parts of wind turbines, research on sensor faults is relatively limited. A possible reason for that is, generally data collection systems in wind turbines do not provide information on sensor faults. In most cases, sensor fault is simulated as sensor fault data in real system is generally not available. Simani et. al. and Bakir et. al. showed the success of ANNs on the detection of wind turbine pitch and rotor speed sensor faults [69-71]. They worked on a simulated wind turbine model as a part of the challenge started by Odgaard et. al. [72] to find out the capabilities of different methods to detect and isolate various wind turbine faults such as faults on sensors and actuators. Sensor faults in these works also created manually.

Sensor faults can appear in various ways. Typical sensor faults encountered in WTs can be listed as; multiplicative, additive, offset faults and faults resulting in changing dynamics in the system [72]. Small drifts in calibration are especially hard to detect as they do not cause significant changes in characteristics of measurements. In this study, multiplicative fault in temperature sensor measurements is investigated. Multiplicative faults may arise from calibration drifts and act like a scaling factor on sensor measurements. This type of a fault was selected to be investigated because unless scaling factor of a multiplicative fault is too big, it does not cause an easily recognizable change comparing to normal response of sensors which makes them harder to detect.

3.1 Data Description

Data used for the sensor validation purpose were collected from temperature sensors of the target turbine. As stated in Chapter 2.2.2, there are 10 temperature sensors mounted on various parts of the turbine. To present the performance of the approach used for sensor validation, a case study was set up by selecting measurements of 4 temperature sensors. One month of data were used which was collected from 01.11.2015 to 30.11.2015. 75% of data were used in the training phase and the remaining data were used to test the performance of networks.

The sensors providing information on rear hub bearing temperature *(S_1)*, control cabinet temperature *(S_2)*, tower temperature *(S_3)* and transformer temperature *(S_3)*

were used to evaluate the performance of the proposed approach. The locations of the selected sensors are presented in Figure 3.1.

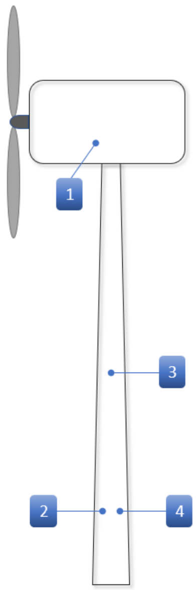


Figure 3.1 : Locations of sensors used for the sensor validation purpose.

The proposed method is based on finding out sensor calibration drifts by using solely the measurements of a group of sensors which are related to each other. Therefore, the subset of sensors was determined by considering the resemblance in the temperature characteristics of the areas they were installed in. By this approach, without hardware redundancy, validation of sensors would be realized by computational methods.

3.2 ANN Input-Output Structures

To find out ANN models that characterize sensor measurement faults effectively, in addition to designing networks with various computational characteristics given in Chapter 2.3, different input-output structures were also created. From this aspect, networks with two different structures were developed. First of them is Auto-Associative Neural Network (AANN) and the second is Multi-Input-Single-Output (MISO) ANN.

3.2.1 Auto-associative neural networks

In Auto-Associative models, input vector is associated with itself. In our application, as presented in Figure 3.2 the measurements from the selected 4 temperature sensors are introduced to the network from the input layer, where the desired output layer also consists of these 4 measurements.

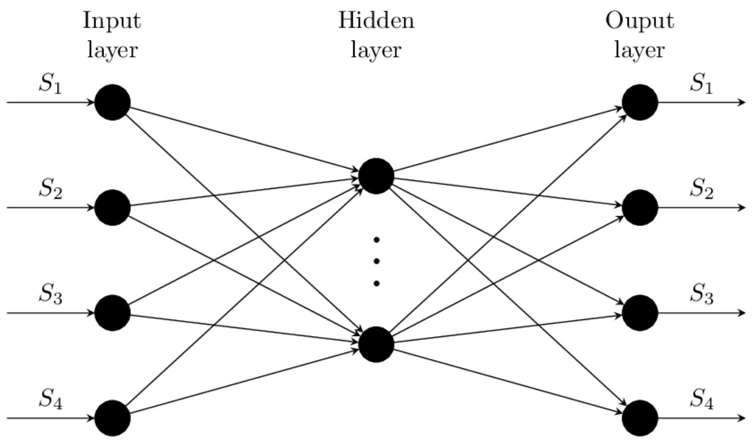


Figure 3.2 : Auto-Associative network structure.

As the input and the desired outputs are the same, the function to be learned by the network is the identity function. However, learning the identity function perfectly would not be useful because it would not cause a transformation on the data. For MFNN AANN, the power of Auto-Associative structure comes from their internal constraints that cause prevention of perfectly learning the identity function. This is provided by the “bottleneck layer”, which is the hidden layer with a less number of neurons from the input and output vectors [62]. Compression of information by the bottleneck results in the acquisition of a correlation model of the input data, which is useful for performing a variety of data processing tasks. For RBFNN and GRNN, this constraint is not valid due to their different computational characteristics than MFNN.

The network reduces measurement noise by mapping inputs into the space of the correlation model, and the residuals of this mapping can be used to detect sensor faults. Values for missing and faulty sensors can be estimated using the network. Auto-Associative networks can be used to preprocess data so that sensor-based calculations can be performed correctly even in the presence of large sensor biases and failures [62].

3.2.2 MISO neural networks

To design Multi-Input-Single-Output (MISO) architectures for the sensor validation problem, measurements of 3 out of 4 temperature sensors are used to be fed from the input layer and the remaining sensor's measurement is used as the reference desired output. This procedure was repeated 4 times with each one of the sensors used as output separately. Figure 3.3 presents the MISO network structure with Sensor 4 is the output and the remaining sensors are the inputs.

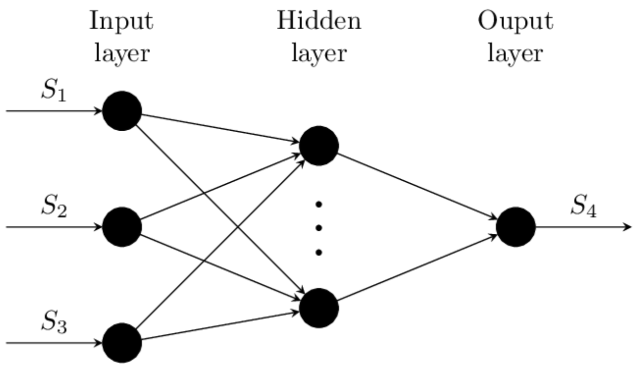


Figure 3.3 : MISO network structure.

Decomposing a multivariable system into multiple MISO models brings some advantages. Each MISO model is simpler than possible Multi-Output networks, which makes implementations become easier in complicated systems. Another advantage is that the required accuracy for each model can be adjusted separately so there is no need for a single loss function results in an accuracy tradeoff between the different model outputs. Moreover, different model architectures, structures and optimization techniques can be applied to each MISO subsystem, so the modelling approach become more flexible [60]. However, as a disadvantage, this approach requires more training time than one overall model.

3.3 Methodology and Network Selection

The sensor validation task was handled as a regression problem. The measurements that were taken during the healthy behavior phase of the temperature sensors were introduced to the networks as inputs and the networks were trained in a supervised way to make close approximations to sensor readings used as ANN outputs. The data set was split into 2 classes as training and test sets. After training the networks using non-faulty measurements, the effectiveness of each network was tested by the original test set and its altered versions as different cases to design situations representing the faulty situation. In case of a fault occurrence, the residuals between the real measurements and the ANN outputs are expected to grow.

Multiple networks with the different computational principles and input-output structures described in the former parts of the thesis were created to find out an effective model that produces the required ANN outputs to detect calibration drifts. Table 3.1 shows the groups of ANN types designed for the sensor validation task.

Table 3.1 : ANN structures designed for the sensor validation purpose.

Network Type	Computational Principle	Input-Output Relation
Type 1	MFNN	Auto-Associative
Type 2	MFNN	MISO
Type 3	RBFNN	Auto-Associative
Type 4	RBFNN	MISO
Type 5	GRNN	Auto-Associative
Type 6	GRNN	MISO

Success of the networks for each group were evaluated using the R^2 performance criteria. R^2 is a statistical measure of how close the estimated outputs fit the actual data. It is one of the commonly used measures to test the effectiveness of regression problems. Equation 3.1 represents the calculation of R^2 value.

$$R^2 = \frac{RSS}{TSS} = \frac{TSS - ESS}{TSS} = 1 - \frac{ESS}{TSS} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2} \tag{3.1}$$

Where, y_i and $\hat{y}_i$ are the actual and estimated outputs for each data point i and $\bar{y}$ is the mean of the output values. RSS is the regression sum of squares; measure of the variation off the fitted regression values around the mean, TSS is the total sum of squares; measure of the variation of the observed values around the mean, ESS is the error sum of squares; measure of the variation of the observed values around the regression line. Equations 3.2-3.4 show the calculation of these values.

$$ESS = \sum_i (y_i - \hat{y}_i)^2 \tag{3.2}$$

$$TSS = \sum_i (y_i - \bar{y})^2 \tag{3.3}$$

$$RSS = \sum_i (\hat{y}_i - \bar{y})^2 \tag{3.4}$$

From each group of networks given in Table 3.1, network with the highest R^2 value was chosen to be used in the next parts of the process. In the subsequent step, outputs of the resulting 6 networks were compared for different inputs to find out a solution for the problem handled.

MFNN networks with Auto-Assosiative structure were created using 2 and 3 neurons in the hidden layer to supply the networks with the bottleneck characteristics. 10 trials were made with each architecture. The reason of multiple trials is to ensure the reach of a minimum point in the cost function space as the performance of the networks alter depending on the initial values of the network weights. Various activation functions were tried such as; *"logarithmic sigmoid - linear", "logarithmic sigmoid - logarithmic sigmoid", "tangent sigmoid - linear" and "tangent sigmoid - tangent sigmoid"*. The former function represents the hidden layer's and the latter represents the output layer's activation function. For the MISO MFNN networks, again different activation functions with multiple trials were applied. As a difference from the Auto-Associative case, the number of neurons in the hidden layer was changed between 2 and 15.

Amongst Auto-Associative MFNN networks, the best scores from the different trials were obtained by the network with 3 neurons in the hidden layer and the activation function pair of logsig-purelin. R^2 values of this architecture are 0.999, 0.999, 0.996 and 0.993 for sensors 1 through 4, respectively. For MISO MFNN networks, activation function pair was also found to be logsig-purelin. Best number of hidden neurons were obrained as 3, 3, 6 and 8 from sensors 1 to 4 in the output layer, respectively. The resulting highest R^2 scores are 0.836, 0.979, 0.984 and 0.985. Figure 3.4 shows the regression graphs for networks with the best performances.

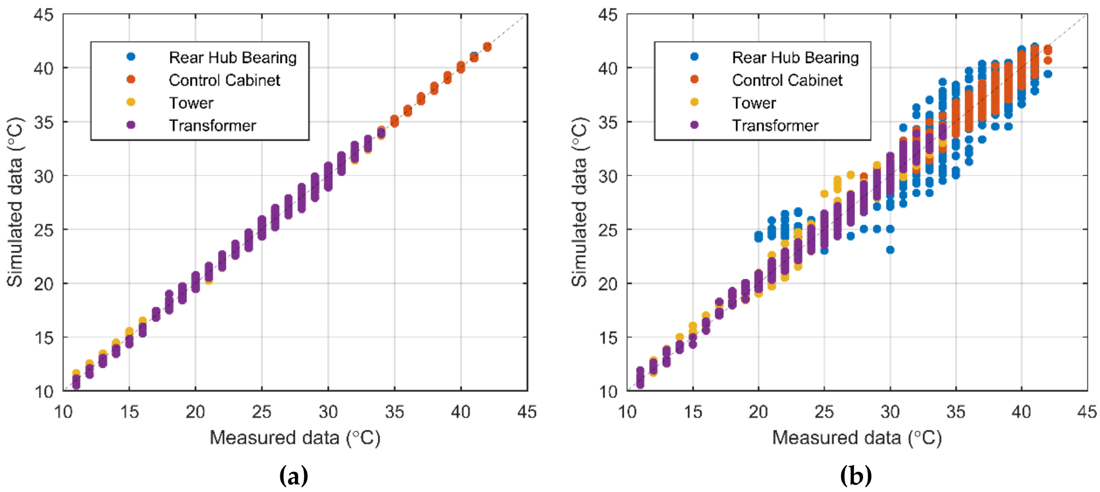
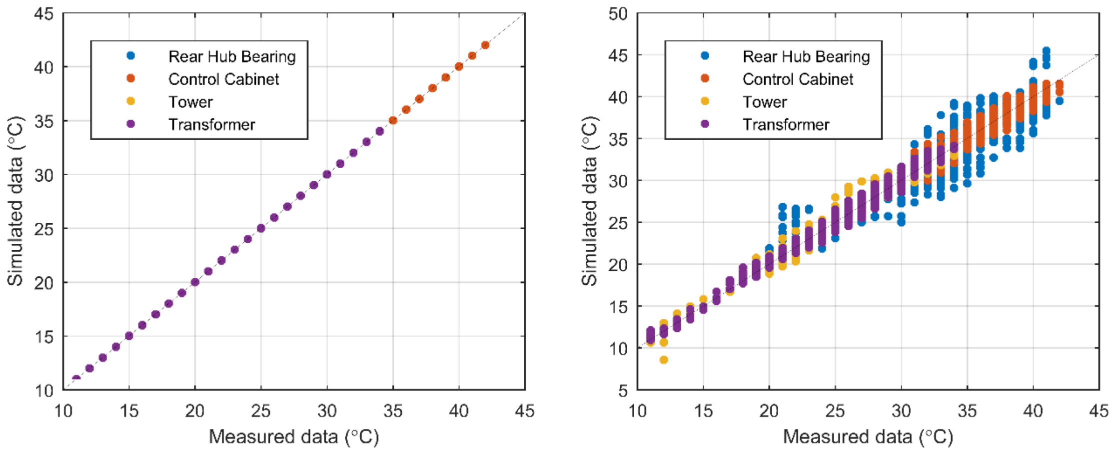


Figure 3.4 : Regression plots for (**a**) Auto-Associative MFNN (**b**) MISO MFNN.

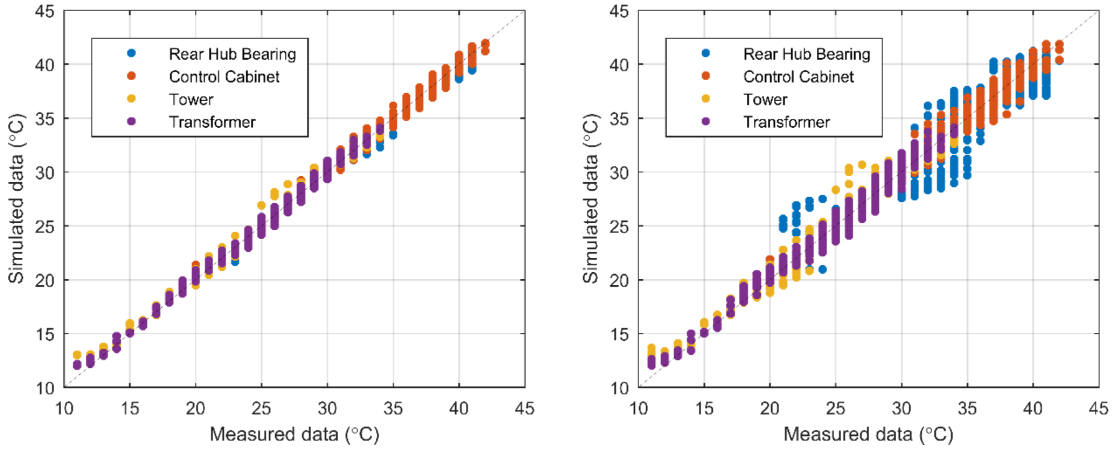
Similarly, several Auto-Associative and MISO networks with using RBFNN were developed. In this case, the difference between the networks are rooted by the varying width factors. Width factors of the networks were selected amongst varying values between 0.1 and 150. In the initial attempts, it was seen that the best R^2 scores amongst Auto-Associative RBFNN networks were bigger than 0.999 for all sensors. This situation shows that the network could not realize a nonlinear mapping. In classical RBFNN approach, the number of hidden neurons is equal to the number of training samples. To reduce the hidden neuron number, training of RBFNN networks was repeated using a smaller amount of training samples, however, even using only 100 samples, the networks could not succeed the mapping between the inputs and outputs. The highest scores for MISO RBFNN networks for sensors 1 to 4 are; 0.824, 0.979, 0.982 and 0.984. Regression plots for the test set of RBFNN networks are presented in Figure 3.5.



(a) **(b)**

Figure 3.5 : Regression plots for (**a**) Auto-Associative RBFNN (**b**) MISO RBFNN.

GRNN models were also designed with varying width factors from 0.1 to 150. The highest R^2 values for Auto-Associative GRNN were 0.992, 0.990, 0.990, 0.994 for sensors 1 through 4, respectively and the best scores for MISO GRNN networks were 0.808, 0.976, 0.973 and 0.981. Figure 3.6 shows the regression plots for Auto-Associative and MISO GRNN models with the highest goodness of fit results.



(a) **(b)**

Figure 3.6 : Regression plots for (**a**) Auto-Associative GRNN (**b**) MISO GRNN.

3.4 Results for Sensor Validation

For analysing the performance of the networks selected in the previous stage of the sensor validation problem, a set of cases were designed by altering the original test set. The main idea relies on the fact that the sensors were selected from relevant places having similar temperature characteristics. Therefore, by solely using the measurements of these 4 sensors, a possible fault in one of them can be detected. Also, it is required to distinguish variations caused by calibration drifts from variations appeared due to the real temperature changes such as changes originated from environmental conditions. To ensure that the proposed algorithm distinguishes the root reason for the temperature change, networks were tested in 3 different cases. Performances of the ANN models for the test set without any temperature drifts was observed in Case 0. In Case 1, a calibration fault was simulated by multiplying only one of the sensor's output with a constant factor. An overall shift in all temperature measurements were simulated in Case 2 which represents a change originated from not a fault but a real environmental temperature change.

The details and different expectations from ANN outputs for all cases are summarized as follows;

Case 0 – No fault case.

In this case, temperature measurements gathered from the sensors were directly used as network inputs. The data presents the temperature values in the normal behaviour of the turbine without any faults, therefore the expectation from the networks is to produce as close output values as possible to the real sensor measurements.

Case 1 – Multiplicative fault in one of the sensors.

Case 1 was designed to introduce a fault to be detected to the measurement system. A multiplicative fault was artificially created in one of the sensors. The test set for the measurements of Control Cabinet Temperature Sensor (S_2) was multiplied by the constant term 1.2. The performances of the networks for this case were evaluated by analyzing the residuals between network outputs and sensor measurements. The expectation is to obtain greater residuals between measurement and estimation of the faulty sensor comparing to other sensors.

Case 2 – No fault case. Overall shift in all measurements due to an environmental temperature rise.

In this case, the measurements from all 4 sensors were shifted by multiplying by the constant term 1.2. The aim for this overall shift is to ensure that networks used for this fault detection algorithm do not produce a false fault alarm when none of the sensors are faulty but instead a real temperature rise is recorded. The expectation from networks for this case is again to produce as close estimation values as possible to actual temperature values.

Input data for all 3 cases were implemented to the resulting 6 models with the best R^2 values. Outputs of the networks were analyzed based on different requirements for each case.

Table 3.2 presents the Root Mean Square Error (RMSE) values of ANNs for each case in ℃. The results indicate that the most convenient network to distinguish the faulty case from non-faulty ones is Auto-Associative MFNN. In Cases 0 and 2, the RMSE between the real values and ANN outputs are less than 0.8 ℃ for all sensors, whereas in Case 1 RMSE values are 1.6 ℃ and 1.3 ℃ for sensors 2 and 4, respectively. Although Auto-Associative RBFNN outperforms the former network in Case 0 and Case 2, it was unable to produce the required residuals in the faulty case (Case 1).

Table 3.2 : RMSE values for each ANN.

	Auto-Assoc. MFNN			MISO MFNN		
Sensor No	Case0	Case1	Case2	Case0	Case1	Case2
S1	0.1	0.4	0.5	1.8	2.7	3.4
S2	0.1	1.6	0.4	0.6	7.2	1.5
S3	0.3	0.3	0.7	0.6	2.6	2.2
S4	0.4	1.3	0.7	0.6	3.2	2.1
	Auto-Assoc. RBFNN			MISO RBFNN		
	Case0	Case1	Case2	Case0	Case1	Case2
S1	0.2e-4	0.3e-3	0.8e-3	1.8	20.8	8.9
S2	0.1e-4	1.3e-3	0.5e-3	0.6	7.2	4.8
S3	0.2e-4	0.5e-3	0.4e-3	0.6	20.8	44.4
S4	0.1e-4	0.4e-3	0.5e-3	0.6	4.9	5.9
	Auto-Assoc. GRNN			MISO GRNN		
	Case0	Case1	Case2	Case0	Case1	Case2
S1	0.5	1.1	3.9	1.9	2.5	4.7
S2	0.4	4.2	2.3	0.7	7.2	2.8
S3	0.4	1.8	1.3	0.7	3.3	1.6
S4	0.4	1.8	1.3	0.7	2.9	1.7

Figures 3.7-3.9 present the Auto-Associative MFNN results for each case. As can be seen from Figures 3.7 and 3.9, real temperature values and ANN outputs are very close in Case 0 and Case 2. Figure 3.8 shows that the residuals are visible in Case 1, which is a sign of fault existence. However, the magnitude of residuals for Control Cabinet and Transformer are very close to each other. This situation prevents finding the exact location of the fault. Therefore, to diagnose which sensor gives the faulty measurements, the necessity of using different network estimations in a combined way has emerged.

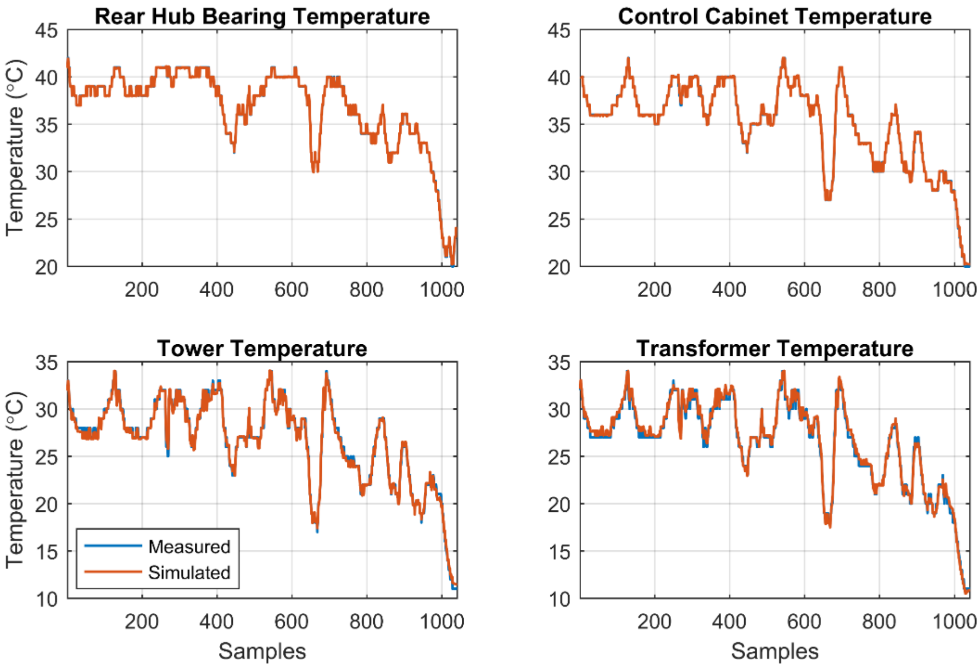


Figure 3.7 : Measured and network values for Case 0 with Auto Associative-MFNN.

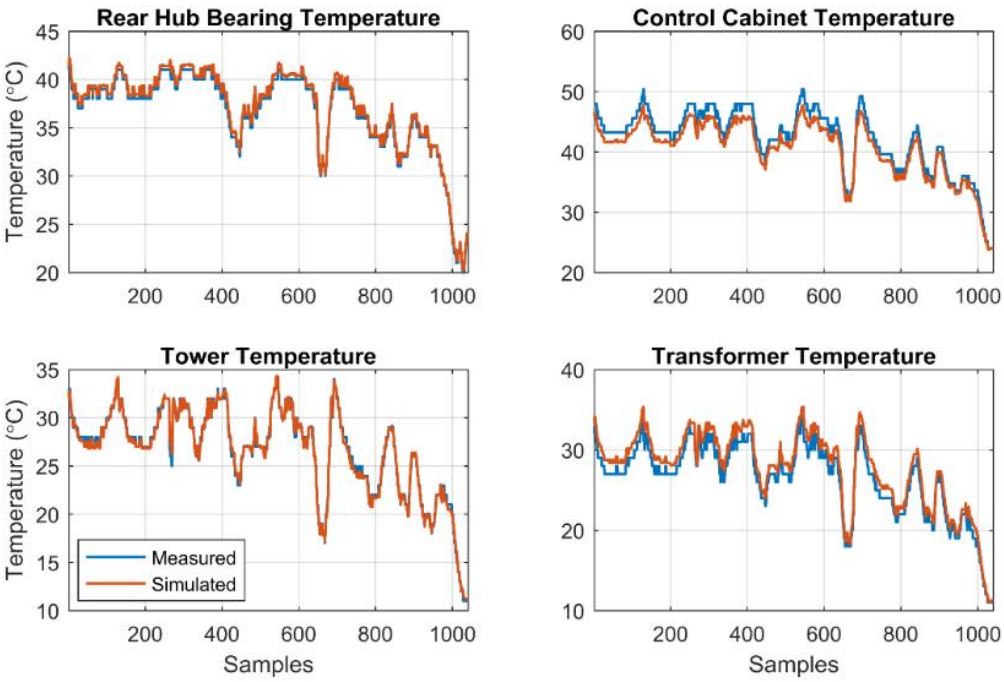


Figure 3.8 : Measured and network output values for Case 1 with Auto Associative-MFNN.

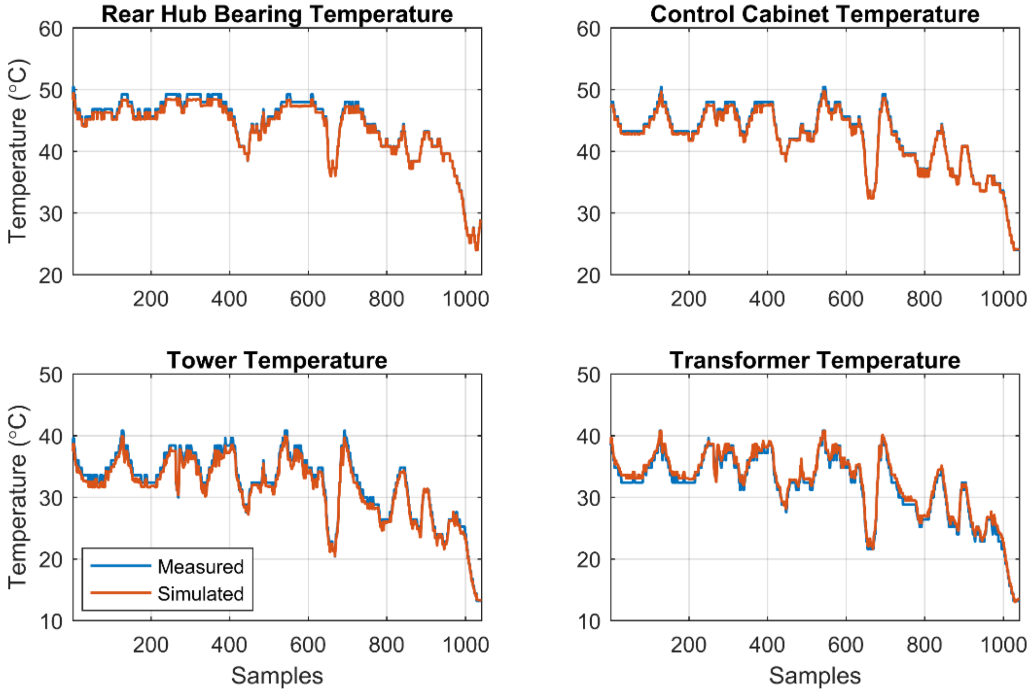


Figure 3.9 : Measured and network output values for Case 2 with Auto Associative-MFNN.

The network providing the best results to determine the location of the fault was found to be MISO MFNN. Where the RMSE value of the faulty sensor (S_2) is significantly larger than other sensors in Case 1 and in other cases, the RMSE values are comparatively smaller. Figures 3.10-3.12 present the results obtained by the specified networks. The corresponding figures show a combination of 4 neural networks with a different output sensor in each subplot due to the single output architecture of MISO networks.

As shown in Figure 3.10, the real data and the network outputs are consistent in the normal operation case. Figure 3.11 shows that in Case 1, unlike the results from the Auto-Associative network, this time the residuals for Control Cabinet Temperature are significantly bigger than the residuals for other sensors with an RMSE value of 7.2 ℃. Therefore, this network can be used for the isolation of fault location. For Case 2 (as presented in Figure 3.12), the networks produce residuals which is undesired for this scenario, however the decision of fault existence would be made with the support of Auto-Associative MFNN to prevent false alarms. By this combined decision-making algorithm using different networks, the fault detection system would be able to distinguish faulty and non-faulty situations in a more sensitive way and give information on the exact location of faults.

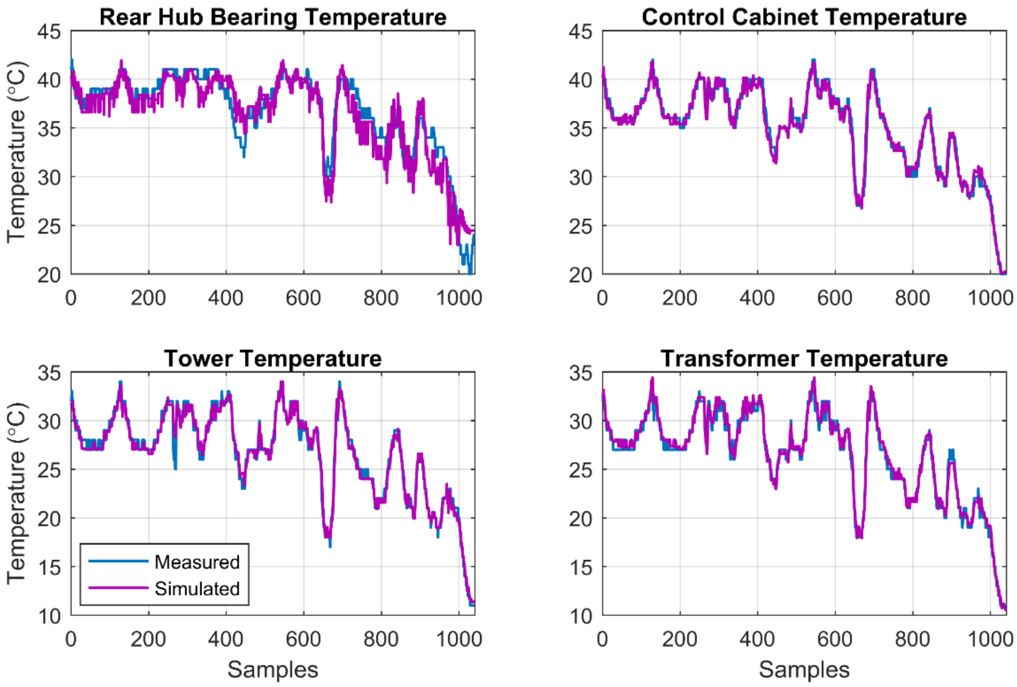


Figure 3.10 : Measured and network values for Case 0 with MISO-MFNN.

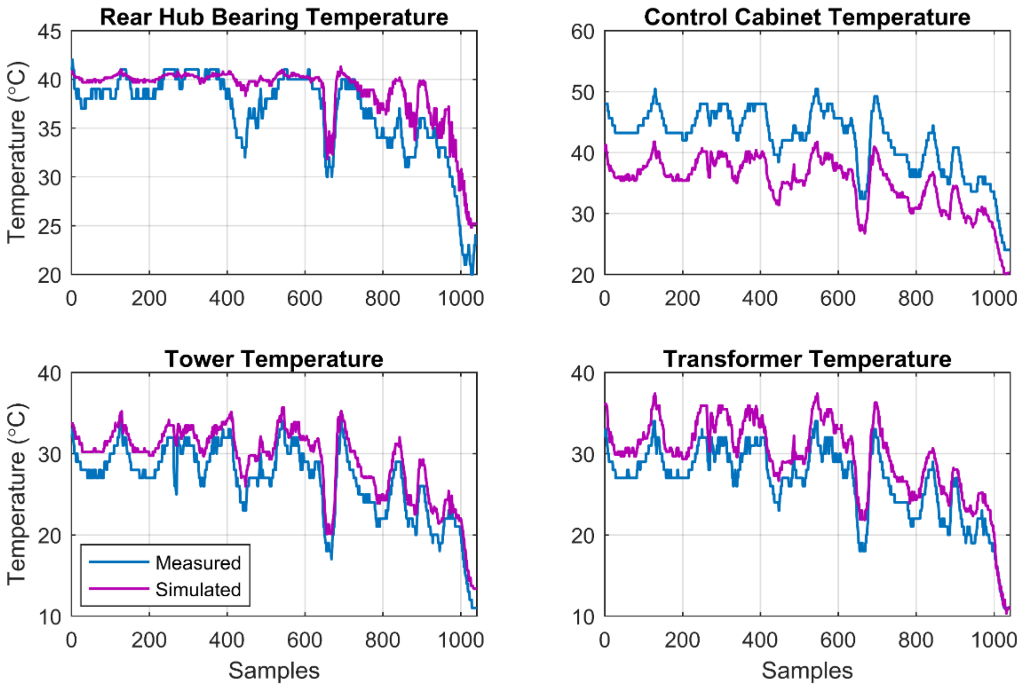


Figure 3.11 : Measured and network values for Case 1 with MISO-MFNN.

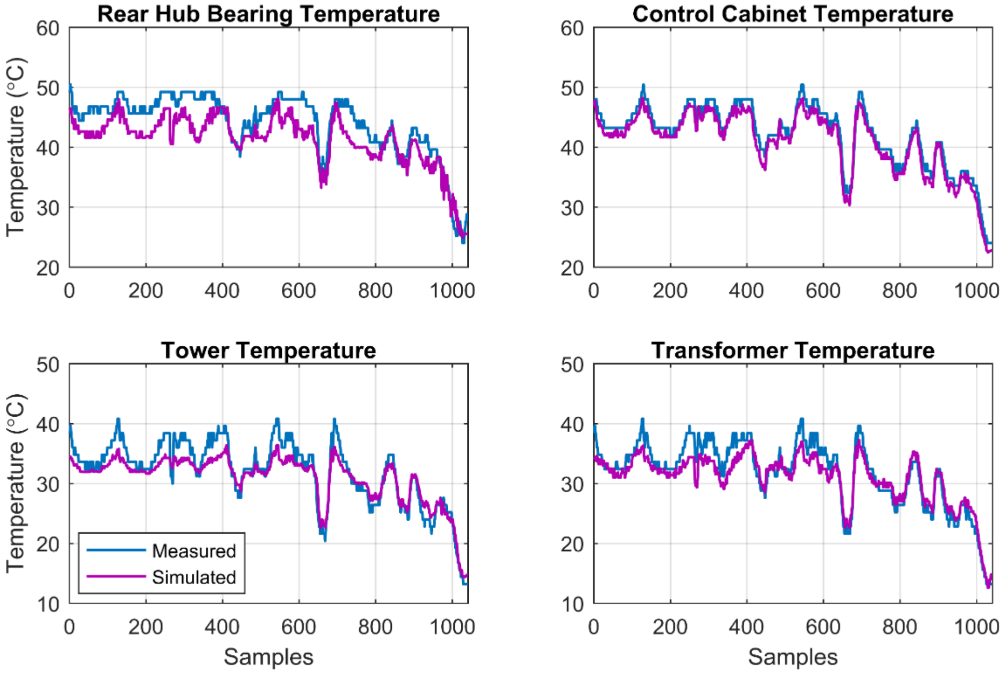


Figure 3.12 : Measured and network values for Case 2 with MISO-MFNN.

Based on the results presented, it is proven that the method implemented is successful on the detection of a calibration drift in temperature sensors. Also, it can discriminate well between the situations of a calibration drift and an overall drift due to an environmental change. This method solely uses the measurements of sensors with similar characteristics, therefore is advantageous of not requiring hardware redundancy or additional usage of sensors measuring different parameters to obtain more information to be used in the network inputs. The calibration status of sensors can be monitored without taking the wind turbine into stall mode and dismantling the sensors to be tested. Therefore, this approach provides a cost-effective solution providing information about drifts that start to affect the health of measurements by continuously monitoring sensors without the need of corrective and preventive maintenance actions.

4. FEATURE CONSTRUCTION AND SELECTION

Determining the features to act as inputs of models is an important step in artificial intelligence applications as performance of algorithms strongly depend on the feature representation of input data. An appropriate set of features is essential on creating fast and accurate models. Feature construction and selection are methods that are commonly used to find out features that successfully characterize the behavior of the system and to improve performance of artificial intelligence models in various ways. In this chapter, feature construction and selection methods that were used to determine a set of inputs that is effective for the fault analysis of the wind turbine is presented. The findings obtained in this part are used in a comparative analysis which is presented in Chapter 5.

Feature construction is to generate new features from raw features. By this way, hidden information about the relations amongst features can be discovered and augmented to the feature space. A new feature can be constructed in various ways depending on the task and requirements of the system. It can be realized in a manual or automated technique. Common methods in manual feature construction can be listed as; knowledge-based, time domain and frequency domain operations. It can also be automatized by aggregating, combining or transforming raw features. This procedure can result in a rapid growth in the number of overall features. For example, after the construction step, in [33] the number of features became more than 400 in a wind turbine fault detection study and in [73] it reached from 49 to 490 to solve a failure diagnosis problem for a robot operation.

Such large numbers of features bring numerous disadvantages. After a certain point, increasing the number of input features would degrade the performance instead of improving that is commonly referred as “Curse of dimensionality” which was initially introduced in the optimization of dynamic programming problem [74]. With the excessive increase in the number of features, the available data become sparse which is an obstacle for tasks requiring statistical significance. For this reason, reduction of features by finding out the inputs that characterize the target task effectively is useful in such cases.

Feature selection is to identify a subset of relevant features from the overall feature set which is a compulsory process in machine learning applications involving moderate and high number of inputs. It brings many advantages such as preventing overfitting that can be caused by large number of features, reducing computational burden and training time, increasing accuracy of the model. Feature selection process can be employed by different approaches namely filter, wrapper and embedded techniques. Filter approaches evaluate features without utilizing any classification algorithm. They rank features independently based on a selected criteria [75-76]. Wrapper methods select and evaluate a subset of features together and search for the best subset describing the model [77]. In embedded approaches, the selection is a part of the learning process [78].

In this thesis, firstly a feature construction procedure was realized to obtain possible features that are more effective on characterizing the problem than the raw inputs. After the feature construction step, a hybrid feature selection method was employed to the acquired features for the fault analysis of the overall turbine. In the feature selection part, at first filter methods were applied to find out and exclude the features that are non-discriminant. Features determined as relevant by the filter methods were than evaluated in a wrapper-based approach to get the knowledge about mutual relations or additional redundancies. By combining these two selection approaches, it was aimed to benefit from the advantages of both. Filter methods are practical in large data sets in terms of training time and complexity. However, they are not able to evaluate mutual dependencies between features. Therefore, after using the filter approach as a pre-processing step, wrapper method was employed to eliminate redundancies and obtain subsets based on evaluating mutual relations.

4.1 Feature Construction

In this part, the methods used to create additional features from the raw features are explained in detail. The constructed features are classified in 4 groups as knowledge-based features, difference features, time series features and statistical features.

4.1.1 Knowledge-based features

Knowledge-based features were generated using the information on wind turbine working principles. First kind of the knowledge-based features are available power values. Using the minimum, maximum and mean wind speed measurements, available power for each 10 min interval was calculated.

The ratios of the available power to the produced power values were also calculated as part of knowledge-based features as it can indicate abnormalities in the health of the system. The last type of knowledge-based features is the sinusoidal components of nacelle position which can also be helpful in providing extra information on possible anomalies.

4.1.2 Difference features

Differences between the relevant parameters of the turbine can be useful in the detection of faults in the related subsystems. Therefore, the differences of features in the temperature measurements, operational data and wind speed data were calculated. For example, the differences between the minimum, mean and maximum rotor power values, wind speeds, produced power values and the difference between the generator rotor and generator stator temperatures are amongst these features. The total number of difference-based features is 17.

4.1.3 Time series features

Raw measurements collected from the wind turbine is a kind of time series data that collectively represents how the system and its behaviour change over time. Time series data are useful to understand the underlying structure and characteristics that produce the observations. To benefit from the time-dependent characteristics of the measurements, the models were supplied with the past measurements on a rolling basis together with the current ones. The original features were lagged to construct new possible inputs to networks. Each original feature was delayed from 10 min to 120 min with the aim of benefiting from time series characteristics. By this way, 132 new features were generated.

4.1.4 Statistical features

Statistical parameters have also a potential of supplying information on fault indications. Therefore, statistical features from original features were generated as the last part of the feature construction step. Moving mean, standard variation and median values of the original features were calculated in a rolling basis from 30 to 120 min time windows. The number of constructed features by the calculation of statistical parameters is 198.

By gathering the constructed features with the original features, a data set with 377 features was obtained. 22 of them are original features and the remaining 355 are generated features. The feature selection methodology is explained in the following part.

4.2 Feature Selection

In artificial intelligence applications, input data sets mostly tend to be high-dimensional which can emerge naturally due to the high number of raw features or such as in our case, additive high-number of inputs could be intentionally created to obtain features that characterize the problem effectively. However, many of these features can either be partially or compeletely irrelevant/redundant to target concept [78]. Feature selection is a mandatory step to define the relevant features. Because irrelevant and redundant features decrease the performance scores and increase the training time of models. In many applications, the size of a dataset is so large that learning might not work as well before removing these unwanted features. Reducing the number of irrelevant/redundant features drastically reduce the time required for the learning process and yields a more general concept [78].

As described in the initial part of this chapter, filter-based and wrapper-based feature selection approaches were applied in this thesis to benefit from the advantages of both. Through this way, the number of features obtained after the feature construction step was reduced significantly and an effective subset of inputs for ANN models was determined. The details of filter-based and wrapper beased methods used are explained in the Sections 4.2.1 and 4.2.2, respectively.

4.2.1 Filter-based feature selection

Filter-based feature selection methods sort features based on the characteristics of the training data by observing which features are more relevant to the outputs. They exploit information contained in input data sets analyzing various characteristics like information gain, entropy, consistency values [79]. These methods are seperate processes working independently on their performance and parameters. They can be treated as kind of pre-procesing procedures as they carry out the feature selection task as an independent step from the induction algorithm [80]. Due to this independency, filter-based methods are generally computationally efficient. The general nature of filters makes them applicable for most cases, however as they disregard the performance of the resulting learning algorithm, it is not guaranteed that the features with the highest scores are able to produce successful results when they constitute an input set [79]. Figure 4.1 shows the structure of filter-based feature selection algorithms.

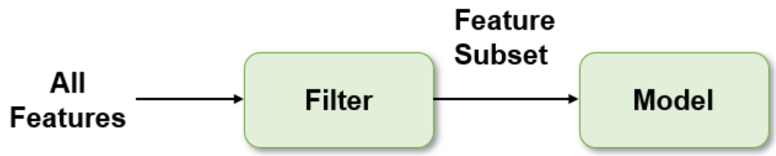


Figure 4.1 : Filter-based feature selection.

In former researches, many feature selection methods have been proposed and different approaches have proven to be successful in different tasks and data sets. Some commonly applied filter-based feature selection methods are; Mutual Information [81], Correlation-based Selection [82], Fisher Method [83-84], Relief Algorithm [85], Laplacian Score [86], Hilbert Schmidt Independence Criterion [87] and Trace Ratio Criterion [88]. In this thesis, initially four of these filter-based feature selection methods were attempted which are; Fisher Score, Relief algorithm, Mutual information and correlation-based feature selection. The initial simulations showed that Fisher score and Relief methods had supplied effective results. Therefore, the detailed subset selection had been conducted using the results based on these two techniques.

4.2.1.1 Fisher method

Fisher method has proven to be a fast yet effective method in many diverse applications. It evaluates features by using the ratio of interclass separation and intraclass variance. It assigns a score for each feature by attempting to find a subset of features that provide in the dataspace spanned by the features, the distances between data points in different classes are as large as possible, while the distances between the data points in the same class are as small as possible. The Fisher score of the i^{th} feature is calculated as follows;

$$f(i) = \frac{\sum_{k=1}^{c} n_k(\mu_k^i - \mu^i)^2}{(\sigma^i)^2} \tag{4.1}$$

$$(\sigma^i)^2 = \sum_{k=1}^{c} n_k(\sigma_k^i)^2 \tag{4.2}$$

Where, n_k is the size of the k^{th} class, μ_k^i and σ_k^i are the mean and standard deviation of the i^{th} feature when considering the samples of the k^{th} class. μ^i and σ^i are the mean and standard deviation of the whole data set corresponding to the i^{th} feature. A higher Fisher score means that the informative value of the corresponding feature is also higher.

4.2.1.2 Relief method

Relief method uses an Euclidian distance metric and nearest neighbor technique to rank the features based on their discriminative capabilities. It works by randomly selecting instances from the training set and calculating a score based on the Equation 4.3 shows the calculation of the Relief score $r(i)$ for feature i. The score of each feature is updated after processing every selected instance based on the difference between the selected instance and the two nearest instances of the same and opposite classes.

$$r(i) = r(i) - \frac{1}{2}\sum_{t=1}^{M}(||x_{t,i} - NM(x_t)_i|| - ||x_{t,i} - NH(x_t)_i||) \tag{4.3}$$

Where, *r(i)* is the calculated score of feature *i*, $x_{t,i}$ is the value of feature *i* in the instance *t*. *M* is the number of the instances randomly selected from the data. *NH(x)* are the nearest sample from the same class *('nearest hit')* and *NM(x)* are the nearest sample from the opposite class *('nearest miss')* and ||.|| is the measurement of distance. The algorithm calculates the discriminative success of each feature with respect to whether the feature differentiates two instances from the same class which is an undesired property and whether it differentiates two instances from opposite class which is a desired property.

4.2.2 Wrapper-based feature selection

Wrapper-based feature selection algorithms require an induction algorithm which is a predetermined learning algorithm to find out which subset of features are effective for the task handled. They use the results achieved from the induction algorithm to evaluate the performance of the selected feature subsets. In wrapper-based selection, the induction algorithm is used as a black box for evaluating features, therefore the behavior of the corresponding feature evaluation function is usually highly nonlinear [89]. In this case, to obtain a global optimal solution is infeasible for high-dimensional data. To address the problem, wrapper-based feature selection algorithms conduct a search in the space of possible features.

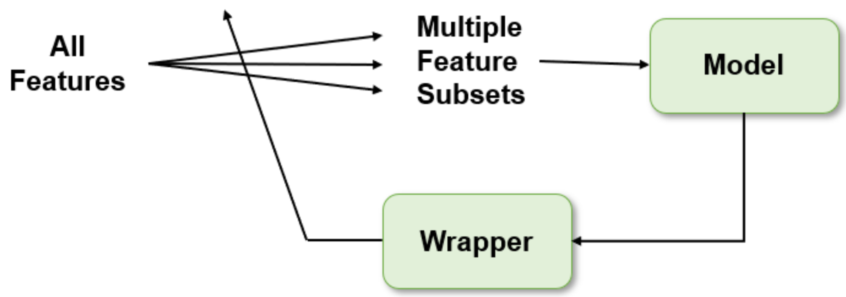


Figure 4.2 : Wrapper-based feature selection.

Various search algorithms can be used to decide the subsets to be used in wrapper models. Common search algorithms can be classified as exponential (also known as complete), sequential and randomized algorithms [90-91]. In exponential algorithms, number of subsets increases exponentially with the number of elements in the feature space. For instance, exhaustive search is a kind of exponential search algorithms where all possible subsets of the feature space are used in the wrapper models to find the best combination. For a dataset with *n* possible features, 2^n number of features are tested in exhaustive search which guarentees an optimal subset. However this approach is not practical in the existence of moderate or high number of features. Sequantial search algorithms add or remove features sequentially. Sequential Forward Search, Sequential Backward Search, Sequential Floating Search Algorithms are amongst the main heuristic search methods [92]. The main idea of randomized algorithms is to use their randomness to avoid the algorithm to stay on a local minimum and to allow temporarily moving to other solutions [91]. Random subset feature selection [93] and genetic algorithms [94] are some common examples for this class.

For the problem handled in this thesis, a sequential search algorithm was used after reducing the number of possible features by selecting the most relevant features in the former part by the filter methods. A sequential search was applied as it provides a balance on the optimality and computational efficiency. Sequential Backward Floating Search (SBFS) used which is a top down search procedure where the initial set starts by the whole feature set which is the most relevant features obtained in the filter-based selection in our case. The least significant feature is excluded in each step which is followed by conditional inclusions [92]. The search continues as long as the resulting subsets are better than the previously evaluated ones at that level. MFNN ANN models were used as the induction algorithm of the wrapper method.

4.3 Results

Methods explained in this part of the thesis were applied to the problem of detecting generator heating faults as a case study. For this aim, the possible input features were analyzed by associating them to generator heating faults as outputs of the networks.

Top 10 ranked features by Fisher and Relief methods were taken to be used in the wrapper selection phase. Table 4.1 presents the features and their ranks selected by Fisher algorithm.

Table 4.1 : Top 10 features selected by Fisher algorithm.

Rank	Features Selected by Fisher Algorithm
1	Difference between generator stator and generator rotor temperature
2	Difference between generator rotor and transformer temperature
3	Difference between generator rotor and nacelle temperature
4	120 min moving median of max power output
5	60 min moving median of max power output
6	30 min moving median of max power output
7	Maximum power output
8	Difference between current and 10-min previous maximum power output
9	30 min moving mean of maximum power output
10	60 min moving mean of maximum power output

As it is seen in Table 4.1, most of the top ranked features are constructed ones instead of original features. The only original feature selected by Fisher method is the maximum power output. There are 4 difference and 5 statistical features amongst top 10 ranked features. It was observed that the temperature differences of generator stator, rotor and other components have the highest impact in this approach to distinguish generator heating faults.

The features with the highest scores assigned by Relief algorithm can be seen in Table 4.2.

Table 4.2 : Top 10 features selected by Relief algorithm.

Rank	Features Selected by Relief Algorithm
1	30 min moving standard deviation of mean rotor speed
2	120 min moving median of maximum power output
3	Difference between minimum and mean rotor speed
4	Difference between minimum and maximum rotor speed
5	Minimum rotor speed
6	Maximum available power from wind
7	Difference between current and 10 min previous minimum rotor speed
8	60 min moving standard deviation of nacelle control cabinet temperature
9	60 min moving median of maximum rotor speed
10	Cosinus of nacelle direction

As presented in Table 4.1 and 4.2, there are no common features which are selected by Fisher and Relief methods. Using Relief algorithm, again only one original feature is amongst the top 10 ranked ones whereas all other features are from constructed features. The original feature selected is the minimum rotor speed. 3 difference features, 4 statistical features and 2 knowledge-based features selected by Relief method. This time, the most relevant features were found to be the statistical features which are the 30 min moving standard deviation of mean rotor speed and 120 min moving median of maximum power output.

After obtaining the highest ranked features by both Fisher and Relief methods, they were analysed in the wrapper-based feature selection algorithm to find out the resulting input set.

In addition, another feature set was selected by heuristic methods in order to compare the performance of the proposed approach. In this method, only the original features and the statistical features of generator rotor and stator temperatures were used in a Sequential Backward Search method.

The resulting features acquired by the proposed feature construction and selection appraoch and heuristic way are presented in Table 4.3.

Table 4.3 : Top 10 features selected by the proposed feature engineering techniques and a heuristic way.

Systematic Feature Engineering Method	Heuristic Method
Difference between generator stator and generator rotor temperature	Minimum wind speed
Difference between generator rotor and transformer temperature	Minimum power output
Difference between generator rotor and nacelle temperature	Minimum rotor speed*
60 min moving median of maximum power output	Mean rotor speed
30 min moving median of maximum power output	Mean wind speed
Maximum power output*	Maximum power output*
30 min moving mean of max power output	Generator rotor temperature
60 min moving mean of max power output	Generator stator temperature
Minimum rotor speed*	60 min moving variance of generator stator temp.
Maximum available power from wind	60 min moving variance of gen. rotor temp.

*The common features selected by both methods.

After applying the wrapper-based selection, it was observed that most of the resulting features come from the Fisher selection algorithm, whereas 2 out of 10 are from

Relief algorithm which are minimum rotor speed and maximum available power from wind.

When compared to the heuristic method, it is seen that the original features selected by the systematic feature engineering methods which are maximum power output and minimum rotor speed were also found to be amongst the best features in the heuristic method.

After using both feature sets as ANN inputs, it was observed that the proposed feature construction and selection methods increase the detection success of generator heating faults. Detailed performance scores obtained by both approaches are presented in Chapter 5.7.2.

5. THREE LEVEL FAULT CLASSIFICATION

A three level fault classification system that consists of the fault detection, isolation and prediction for the overall wind turbine is presented in this chapter. Determining if the turbine is in the normal or faulty operation mode, in case of a faulty operation, finding out the type of the fault, and the prediction of possible upcoming faults are the objectives of the proposed system. These operations form a complete system that provides required information to the wind farm operators to take the actions or measures in the case of a current fault or upcoming fault.

Unlike the sensor validation method designed in Chapter 3, this part of the thesis benefit from the actual fault information collected from the target turbine instead of a simulated fault. All types of collected data explained in Chapter 2.2 were used for the design of the fault analysis system. Information of fault instances were obtained from the status data. The historical status data, historical and current information of temperature data, operational data and wind data were used for the current or future estimations of fault statuses.

During the data collection period from the wind turbine, a fatal failure of the main components which would result in a replacement and a long duration of downtime did not occur. However, a large number of non-fatal but frequent faults observed. As described in Chapter 1.4, former research results show that the prediction success of fatal faults are comparatively very high which can be realized months in advance. However, as frequent but non-fatal faults show less apparent indications, the prediction horizon is not very far. But becuse these type of faults happen in all of the wind turbines and cause a decreasing efficiency and loss of power conversion, detection of them are recently focused on by various studies.

The types, explanations and durations of the faults appeared in the target turbine are presented in Chapter 5.1. Also, information on power conversion in normal and faulty operations were given.

The characteristics and different aims of the three fault classification levels are presented in Chapter 5.2 in detail. To introduce the data to the classification models with different requirements, the data set should be pre-processed accordingly. This pre-processing step is a significant part of the design especially because it includes assigning labels to each 10 min instance to be used in the classification models. Different data classes were created for each classification level based on varying aims. Chapter 5.3 provides information on the data labeling to constitute data classes for each level.

The data set used in this study is a very unbalanced set. Which means the amount of data in different classes are very distant from each other which possibly causes complications for the models to be trained. To handle the unbalanced dataset problem, some techniques were used which are explained in Chapter 5.4. ANN models were designed to classify faults in all of the levels. Chapter 5.5 presents the information on ANN models that were used in this part of the thesis. Finally, the performance metrics and the results of the fault classification system are presented in Chapters 5.6 and 5.7, respectively.

5.1 Fault Information

The instantenous condition of the wind turbine and the reason of that condition can be obtained from the "status data" of the SCADA system. If the turbine is in the working condition without any abnormalities, the status code is "Turbine in operation". Also, if there is no fault and the turbine is about to start working, "Turbine operational" and "Turbine starting" codes consecutively appear before it starts working. All other status codes different than these 3 conditions show some abnormalities about the wind turbine or the environmental conditions.

Stall of the turbine due to the environmental conditions are generally caused by reasons related to wind speed. In accordance with the working principles of wind turbines, they operate in specific wind speed intervals. Therefore, it was observed that the status code was "Lack of wind" where the wind speed is too low for the operation and in the "Storm" status where it was higher than the cut-off speed of the wind turbine.

Amongst the fault statuses, it was seen that fault types which have sufficient amount of samples to be used in the classification models are; “Mains failure”, “Generator heating” and “Feeding fault”. Mains failure describes the problems related to the mains electricitiy supply of the turbine. Additional information regarding the details of mains statuses show that mains failures occurred due to many reasons like; underfrequency, undervoltage and overvoltage. Mains failure statuses result in a blackout of the turbine. Generator heating faults refer to the problems about overheating of the generator which appeared due to the problems in isometer or mains. Feeding faults show problems in the power feeder cables of the turbine. After these faults appear, they continue for a varying amount of time. The number of occurences of these faults during the one-year period of data collection and the total 10 min instances that give information on the duration of each type of fault are presented in Table 5.1.

Table 5.1 : Frequent fault types.

Fault type	Frequency	Number of instances
Feeding fault	3	80
Generator heating	14	222
Mains failure	54	65

Frequency in Table 5.1 refers to the independent number of occurrences of each fault type. For instance, 14 generator heating faults were observed and the total number of these 14 faults correspond to 222 10 min samples.

Other than these frequent faults, some different faults have also been observed in the data set. Namely, “Pitch control error” and “Semiconductor fuse blown” faults happened both for once in the entire duration. However, they were not attempted in the fault classification levels as there are not sufficient amount of data to train and test the models. Also, some different conditions about physical limitations were observed such as “Cable twist”.

Status data also supply information on the maintenance actions. A high amount of “Turbine in operation during maintenance” were observed. Besides, for 8 times, “Maintenance” status code appeared where the turbine goes into a stall condition during the maintenance actions.

Figure 5.1 presents the normalized power values depending on the wind speed for each 10 min instances and the normalized ideal power curve of the turbine. The values were presented in a normalized way due to the non-disclosure agreements signed with the wind turbine company. As it is seen in the figure, the turbine is operational in some fault situations and in the stall mode for the others. Also, it does not track the ideal power conversion line in every normal operation status. These are some indications of the non-linear relations within the turbine operations.

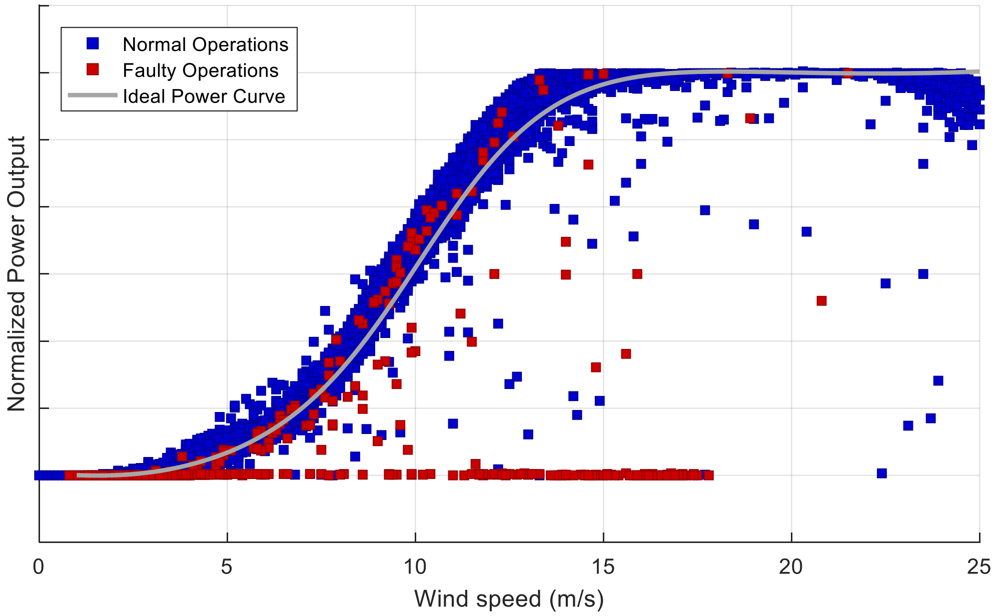


Figure 5.1 : Normalized power output values for the normal and faulty operation statuses.

5.2 Classification Levels

In this part, the three different fault classification levels that were designed for the generation of the overall condition monitoring and fault classification aim were explained. The characteristics and objectives of the fault detection, isolation and prediction levels are described in Chapters 5.2.1, 5.2.2 and 5.2.3, respectively.

5.2.1 Fault detection

Fault detection is the first level of classification where the aim is to distinguish faulty instances from non-faulty ones. The distinction between fault types is not important in this level. Therefore, the data were split in two classes describing the normal operation class and the faulty class. The fault class consists of all the frequent fault types presented in Table 5.1 and the instances of statuses which show non-frequrent faults like semiconductor fuse blown are also taken as elements of the fault class. All the remaining data constitute the normal behavior class. There is a significant imbalance between the number of elements in the normal operation (majority) class and the fault (minority) class. The number of elements in the majority class is more than a hundred times greater than the instances in the minority class. This situation may degrade the success of the classification models. Therefore, some techniques were applied to increase the performance of the models which are explained in Chapter 5.4.

5.2.2 Fault isolation

Fault isolation (which is also described in some sources as fault diagnosis) is to determine which subsystem is subject to fault. In this level, in addition to determine if a fault occurred, the exact location of the fault was also attempted to be detected. As they are in a sufficient amount for the ANN models to be trained and tested, the frequent fault types presented in Table 5.1 were attempted to be isolated in this level. For this aim, the data set was split in four classes according to the status codes. Three of the classes represent each fault type and all the remaining samples were assigned to the normal operation class including the situations like non-frequent fault instances and maintenance actions.

5.2.3 Fault prediction

The aim of this level is to predict fault statuses and isolate the exact fault types in advance. The fault prediction level is the most advanced step and it serves for the aim of designing a system which enables operators to be informed about future faults and take required actions to prevent or minimize downtime and possible detrimental effects resulted by faults.

Fault types investigated in the second level were also attempted in this level. First simulations show that the results of generator heating faults are more promising whereas prediction of mains failure and feeding faults could not show successful results. The reason for this is that they occur by the status of the grid not the turbine. Therefore, after observing the prediction success of all 3 types of faults in the initial part of this level, a more detailed research was carried out on prediction of generator faults.

5.3 Data Pre-processing

Pre-processing of data is an essential step effecting the training success. Due to the reason that SCADA data collection systems have many imperfections, the importance of data pre-processing even gets more important. As the initial step, the data set was cleared from blank entries which reduced the number of 10 min instances from 52560 to 50952. This shows 268 hours of data became lost during the data collection period.

After clearing the blank entries, the following step was to assign a status code for each 10 min instance. The matching of the status data to the other types of data is a complicated part of the pre-processing. As explained in Chapter 2.2.4, status log changes whenever the condition of the turbine changes, whereas other data sets renewed in every 10 mins. This results in the situation that, in the same 10 min interval, more than one statuses may appear which is seen frequently. In the presence of this case, the possible main reason of the status data was selected as the main status of the regarding instance and they were continued until a new status appears. By this way, data classes were built for each level depending on the varying aims.

In the fault detection level, the labeling of data belong to each 10 min interval were performed by classifying them based on the corresponding fault statutes. For instance, for the fault detection level if any kind of fault happens less than 10 min and not more than 10 min from a fault status that instance was considered as a part of the fault class [36]. Equation 5.1 presents the labeling in the fault detection level.

$$y(t) \;=\begin{cases} F, & t_s - 10min < t < t_e - 10min \\ N, & otherwise \end{cases} \tag{5.1}$$

Where, F is the label assigned for faulty states, N is the label for the normal states, $y(t)$ is the label at time t, t_s is the start time of the fault, t_e is the end time of the fault.

10 min threshold is used to capture all the information related to the faulty instances. By this way, if a fault status appears between 20:32-20:48, all instances starting from 20:30 to 20:50 will be assigned to the fault class.

In the fault isolation level, the labeling has the same main principle with the process in the detection level. The only difference is that, all types of frequent faults are assigned to a different class. As a result, 4 different data classes were generated. The 10 min threshold was used again, to capture all the information on the fault.

Data labelling in the fault prediction level was managed by a different technique as in this level, different from the first two levels, not the current but the future fault estimation is the objective. As stated in Chapter 5.2.3, in this level only the generator heating fault was attempted due to the potential successful results. Two classes were created, first of them is the pre-fault class for generator heating faults and the second class contains all the remaining data including other types of faults. Data labeling was performed by using the approach proposed by Leahy at el. [31] which significantly improves the capability of early prediction. In this approach, the times during which a fault occurs are not labelled as fault class, instead a time band before the fault was labeled as pre-fault data. Training of the models were realized by using pre-fault and normal operation classes.

$$y(t) = \begin{cases} PF, & w_s - 10min < t < w_e - 10min \\ N, & otherwise \end{cases} \tag{5.2}$$

Where, *PF* is the pre-fault class, *N* is the normal operation class, w_s is the start and w_e is the end of the pre-fault time window. Table 5.2 shows the different time bands for generating data sets. For instance, in Case C, all the data instances with timestamps between 6 to 12 hours before a generator heating fault were selected for the pre-fault class. Positive prediction in this time band means that, the model succeeds to identify an upcoming generator fault at least 6 hours before the fault starts.

Table 5.2 : Time windows created for the pre-fault instances.

Case	Time window (hours)
A	0-1
B	2-12
C	6-12
D	12-24
E	12-48

5.4 Methods to Improve Training Performance in Imbalanced Datasets

A dataset is imbalanced if the classification categories are not approximately equally represented [95]. In such situations, classification algorithms are generally likely to classify new observations in the majority class because in the training phase, they were built to minimize errors and a tendency to classify new data as part of majority class would reduce the overall cost value.

Fault detection is one type of many areas with naturally imbalanced datasets as they have much more samples in the normal class than the faulty class. In our dataset, for instance for the fault detection level, the number of elements in the majority class is more than a hundred times greater than the instances in the minority class. This high imbalance rate creates a tendency towards new observations to be classified as normal instances, however, in many fault detection systems it is more important to correctly classify faulty states than normal states.

There are two common practices to solve this tendency of models built in imbalanced datasets. One of them is to assign distinct costs to training examples and the other is to re-sample the original training data set [96]. In this thesis, the second approach was implemented to increase the classification success of the faulty states.

The techniques applied are oversampling the minority class and undersampling the majority class. Under sampling of the majority class data was realized by randomly selecting some samples to be deleted. For the oversampling of minority class, Synthetic Minority Oversampling Technique (SMOTE) [96] which is one of the most common oversampling techniques in machine learning applications was used. After the classes were constituted and split into training and test sets, these techniques were applied only to the training sets. Data in the test sets were kept in their original form for all the classification levels.

5.4.1 Oversampling of minority class

Synthetic Minority Oversampling Technique (SMOTE) is a common method used in diverse areas due to its less application-specific manner than other similar algorithms. Because it operates in “feature space” instead of “data space” [96]. In this method, the minority class is over-sampled by taking each minority class sample and introducing synthetic examples along the line segments joining any/all of the *k* number of minority class nearest neighbors. Depending upon the amount of over-sampling required, neighbors from the *k* nearest neighbors are randomly chosen. To generate the synthetic samples, firstly the difference between the feature sample under consideration and its nearest neighbor is taken. This difference is multiplied by a random number between 0 and 1 and added to the feature sample. This causes the selection of a random point along the line segment between two specific features. By this approach, the decision region of the minority class becomes more general.

5.4.2 Undersampling of majority class

The second technique used to handle the problem was to undersample the training examples in normal states. The undersampling procedure was implemented by randomly selecting the samples to be removed. The number of instances to be removed was decided empirically and for the fault detection level it was found out that the best results were taken by removing 20000 out of 35000 samples from the training data set. Undersampling and oversampling methods were both applied independently and in conjunction for an overall performance comparison of the attempted re-sampling techniques.

5.5 ANN Architectures

Various ANN models were designed in all levels of classification to obtain effective structures for the problems handled. After some initial tests, it was seen that MLP networks are successful in distinguishing the faulty states from normal ones. Therefore, MLP networks with different parameters were focused on to obtain the final architectures.

For the fault detection level, as the number of possible outputs is 2 -which are the faulty and normal classes- MISO (Multi-Input-Single-Output) models were used. The inputs were selected by heuristic ways and by the feature engineering techniques given in Chapter 4 in different trials. The number of hidden neurons was changed from 1 to 15. Various types of activation functions such as *"logarithmic sigmoid", "linear", "tangent sigmoid", "soft-max"* were tried in the hidden and output layers. Also, multiple trials with random initial ANN weights for each architecture were held to ensure that the models do not stuck in a local minimum in the cost function space. The results obtained with the most successful networks are given in Chapter 5.7.

For the third classification level, in which the fault prediction problem is handled, the network architecture was again MISO type. Where, the possible outputs are the pre-fault for the generator heating faults and the remaining instances. Therefore, searching for a successful network architecture was realized in a similar way to the fault detection level.

The problem to be handled differs in the fault isolation level. This time there are 4 possible outputs representing 3 types of faults and in addition the remaining instances as the 4^{th} class. Therefore, the case turns into a multiclass classification problem which requires a different architecture than the other two levels. One possible solution for this case is to use multi output ANN which uses the selected features as inputs and provide more than 2 possible outcomes as the response of the network. However, the initial test results with multi output ANN showed that the networks had difficulties with distinguishing faults from each other in the multi-output structures. Therefore, One-Against-All (OAA) ANN models were created. Distinguishing each type of faults from the remaining instances was performed by 3 different ANN and all the results were evaluated by a decision rule to reach the final decision of the algorithm. To generate the decision rule, the performance of the best networks for each fault type was evaluated in the training set. It was determined based on the false alarm tendencies of the models. The ANN output that belongs to the fault type which shows less false alarms was selected against the other ANN outputs. A basic scheme of the OAA architecture is shown in Figure 5.2.

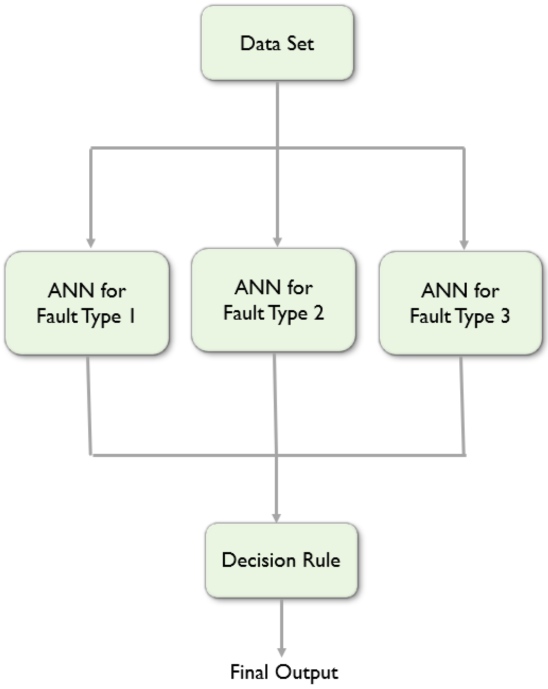


Figure 5.2 : OAA network structure used in the fault isolation level.

5.6 Performance Evaluation Metrics

In classification problems, the most widespread performance indexes to evaluate the success and compare the performance of models are accuracy, specificity, recall (also known as sensitivity), precision and f-score. Equations 5.3 to 5.7 present the calculation of these metrics.

$$accuracy = (tp + tn)/(tp + tn + fp + fn) \tag{5.3}$$

$$specificity = tn/(fp + tn) \tag{5.4}$$

$$recall = tp/(tp + fn) \tag{5.5}$$

$$precision = tp/(tp + fp) \tag{5.6}$$

$$f - score = \frac{2tp}{2tp + fp + fn} = 2 * \frac{recall * precision}{recall + precision} \tag{5.7}$$

Where, t_p is true positives; number of correctly classified fault instances, t_n is true negatives; number of correctly classified normal instances, f_p is false positives; normal instances incorrectly predicted as fault instances; f_n is false negatives; fault instances incorrectly labelled as normal instances.

In applications like fault detection where there is a natural imbalance between the number of samples in different classes, accuracy and specificity metrics may not be informative. Because of the excessive amount of the normal samples comparing to the faulty samples, even if there is no correctly classified fault samples, accuracy and specificity can still reach values close to 1 due to the high number of correctly classified normal samples. For instance, in a dataset with 990 majority and 10 minority class samples; if the correctly classified samples from the majority class is 900 and there are no correctly classified samples from the minority class, the accuracy value would still be 90% despite non of the minority class samples are correctly classified which means non of the faults are detected in problems like ours. Recall, precision and f-score are informative in such problems. Recall can be described as how successfully the model can predict the faults, and precision is how successfully the model can identify only the relevant points as fault instances. F-score is the harmonic mean of the former two metrics, which provides information on both recall and precision. Recall and precision are conflicting metrics and the goal of classification in imbalanced datasets is to improve recall without degrading precision [95]. Because there is a significant imbalance through the classes in the fault detection and isolation levels, accuracy and specificity results found to be always greater than 0.95. Therefore, recall, precision and f-score were used to select the best model. In the fault prediction level, the imbalance is comparatively smaller due to the long time spans of the pre-fault windows so the evaluation was made by also considering accuracy value.

The classical performance metrics given in Equations 5.3-5.7 are calculated in a point-based manner which means the network outputs for all the time instances were evaluated independently to calculate the metric values. In fault detection and isolation levels, this approach is convenient as the previous and following values are not of interest. However, for the fault prediction level it is not the case. In this level, a window-based interpretation of results is more appropriate than a point-based approach to evaluate the performances of models. Therefore, such an implementation was realized to find out the effective prediction horizon. In this method, the response of the network is monitored using a sliding-window. If the network claims a fault indicator for a specified time period, an alarm flag is raised. By this way, it becomes possible to determine the prediction horizon effectively and the number of false alarms are reduced as the alarm flag is not activated after each estimation that exceeds the threshold value but it monitors the ANN results a specified period of time.

5.7 Results and Discussions

This chapter presents the details and performance results of the networks created for each classification level.

5.7.1 Results of the fault detection level

As described in the former sections, many different ANN architectures were designed for the fault detection problem changing the hidden neuron numbers, activation functions, initial weights. Also, the training set was modified in some trials by undersampling the majority and/or oversampling the minority classes to increase the classification performance. Approximately 7/10 of the data (35000 instances) were selected to train the networks and the remaining samples were used to test the performance. The best activation function pair was found to be logsig-softmax functions for the fault detection level. The best performance metrics reached for the original, under sampled and over-sampled training sets are presented in Fig. 5.3. The selection was realized based on the highest recall value reached for each case. To observe the effects of undersampling and oversampling of the training set, the same threshold value which was selected empirically as 0.8 was used to evaluate the outputs of the networks. Outputs which are smaller than this threshold value were considered as faulty and the rest as normal class estimations. Hidden neuron numbers between 3 and 9 were found to be more successful in general. The ANN with the best recall value has 7 hidden neurons for all the cases.

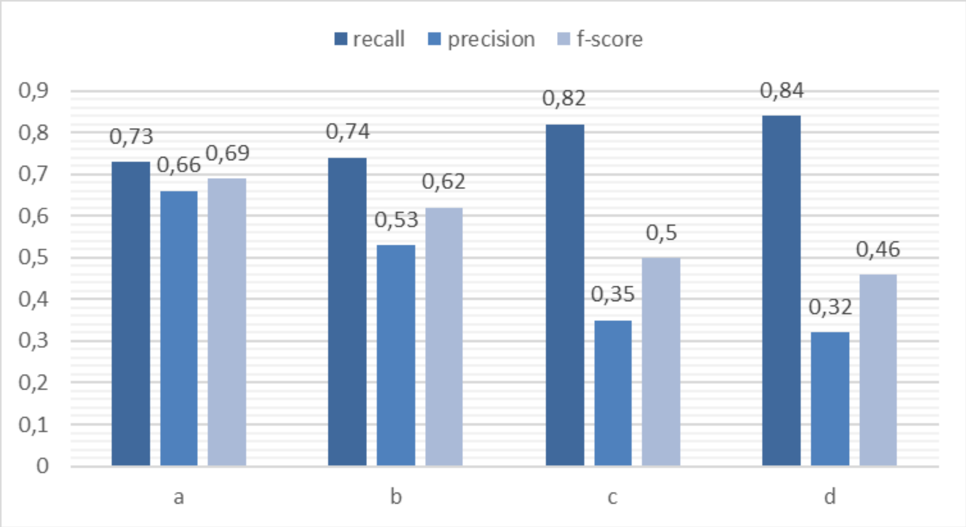


Figure 5.3 : Performance metrics in the fault detection level for a) Original training set b) Only oversampled case c) Only undersampled case d) Oversampled and under sampled case.

The results show that the application of ANNs for fault detection level provides high-performance outputs. For all the cases, the accuracy and specificity results are close to 1, however as they are not informative in highly imbalanced data sets, the evaluation was realized analyzing recall, precision and f-score. Based on the f-score values, which provide information on both recall and precision, the training processes realized by the original training set (Case a) and by oversampling of minority class (Case b) resulted in successful outcomes of 0.69 and 0.62, respectively. However, as the correct classification of minority samples is more important in fault detection systems, the main aim is to obtain a high recall score. The best recall score was obtained in Case d where both under sampling of majority and oversampling of minority class were held in the training set. In this case, recall value is 0.84 and precision value is 0.32.

As presented in Figure 5.3, by modifying the training set with undersampling and oversampling techniques, recall score increases whereas precision decreases. These data show that more faults can be predicted by these methods in the expense of having more false fault alarms. In terms of comparing the undersampling and oversampling methods, it was seen that the undersampling of the normal class is more advantageous. Although the recall values are very close to eachother in Case c and d, after investigating the results of all the networks produced, it was seen that only 1 out of 150 ANN in Case c produced a recall score more than 0.8, whereas it was 10 in Case d. Therefore, the results show that using both methods together is beneficial on detecting more fault instances.

5.7.2 Results of the fault isolation level

In this level, the imbalance rate is higher than the first level. The reason for this is that in OAA ANN structure, a fault class consists of only one type of fault and all the remaining data including other faults are assigned to the normal operation class. The same procedure was repeated for all the three types of faults. Therefore, to reduce the negative effects of the imbalance, both under sampling of majority class and oversampling of minority class for the training sets were applied. Multiple models for 3 different classifiers were generated with various hidden neuron numbers, activation functions and random weight initializations. Then the outputs of the best models for each fault type were evaluated together by a decision rule to reach the final decision of the overall model. The decision rule was determined based on the false alarm tendencies of the models. The ANN output belongs to the fault type which shows less false alarms was selected against the other ANN outputs. ANN for isolating mains failure shows the least number of false alarms. As a result, if the mains failure fault classifier gives a fault estimation output, the overall decision was considered as mains failure ignoring the decisions from the models of feeding fault and generator heating. The second dominant classifier was found as generator heating. Fig. 5.4 shows the resulting performance scores for all fault types.

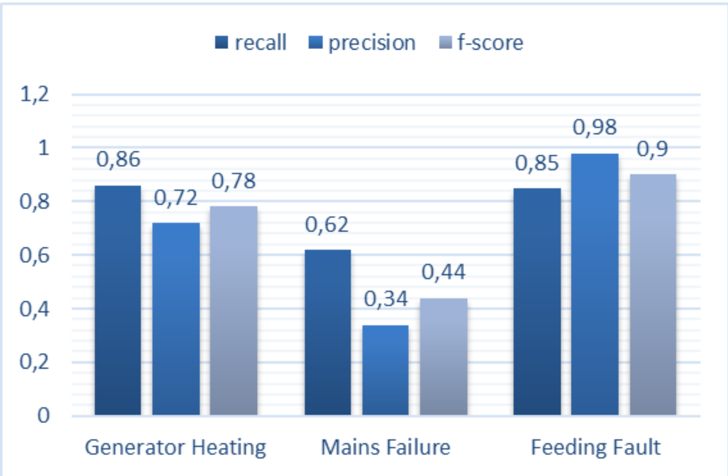


Figure 5.4 : Performance metrics for the fault isolation level.

As presented in Fig. 5.4, both recall and precision values for feeding fault are very high 0.85 and 0.98, respectively which results in a high f-score value which is 0.9. The results for the isolation of generator heating faults come second with 0.86 and 0.72 recall and precision values. Isolation of mains failure reveals lower rate of success results which are 0.62 recall and 0.34 precision values.

Amongst the models built, the highest performances were obtained as follows. The best architecture for the isolation of generator heating faults was obtained by logsig-tansig activation functions for the hidden and output layers with 4 hidden neurons. For mains failure, logsig-softmax activation functions and 5 hidden neurons and for feeding fault, again logsig-tansig activation functions with 11 hidden neurons were found to be the best activations functions.

Results presented in Figure 5.4 belong to the networks generated by the heuristic feature selection method. The case study on feature construction and selection methods for the generator heating faults improved the detection of this kind of faults. In order to analyze the effect of the systematic feature generation and selection processes, the final selected features given in Table 4.3 were used without changing the other parameters of the best network architecture. Figure 5.5 presents the performance metrics obtained by the selected features as inputs of the network with the best performance results.

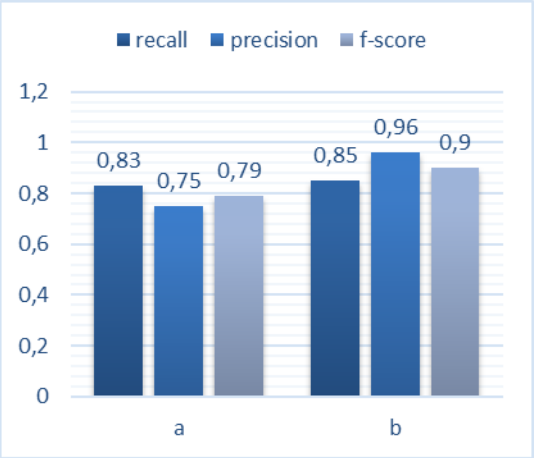


Fig. 5.5 : Performance metrics obtained by a) Heuristic b) Systematic feature selection approaches.

Figure 5.5 shows the performance metrics for isolation of generator heating faults with heuristic and systematic feature selection approaches. The heuristic method uses the same features used to obtain Figure 5.4, however the differences in performance metrics between 2 cases is rooted from the OAA structure in Figure 5.4. Figure 5.5 shows the raw metrics for the generator heating faults without a decision rule.

The results presented in Figure 5.5 show that the systematic feature engineering techniques helped to obtain better results for all the metrics. The improvement in the recall value is comparatively smaller which increased from 0,83 to 0,85. However, in precision metric there is a more significant improvement. It increased from 0,75 to 0,96. For the 3 months test period, this result implies that the amount of false generator heating alarms decreased from 210 to 30 mins. This is an important advantage as high rate of false alarms is one of the most significant problems in wind turbine fault detection systems using SCADA data.

5.7.3 Results of the fault prediction level

The results of this level show that contrary to the first two levels, accuracy and specificity are not close to 1 in all trials. Therefore, in this level model selection was made considering also accuracy score in addition to recall, precision and f-score. A possible reason for the smaller accuracy and specificity could be that pre-fault indications of this level are not as strong as the fault indications of the first two levels, so the relations are harder for the models to interpret which results in more false outputs. Also, as the pre-fault band gets wider, the number of samples in the minority class becomes higher in comparison to the first two levels as they are made of the time instances within a pre-specified time window which can be up to a 36 hours window worth of instances. This situation results in a less imbalanced data set and makes accuracy and specificity become concerns. Besides, fault indications before the beginning and after the end of the pre-fault time windows are seen as “false positives”, however they are successful indicators of upcoming faults which also contribute to the low scores in classical performance metrics. Because of this factor, a sliding window interpretation of ANN outputs is more convenient for this level, however, results for the classical performance metrics are also provided to serve as a basis for the future researches.

Initial simulations show that the prediction performance for mains failure and feeding faults are of low performance, therefore only generator heating fault was tried to be predicted in this level. The reason for this is, mains failure and feeding faults are not faults caused by the internal dynamics of the turbine but occur due to the external factors. Therefore, solely monitoring turbine parameters is not enough for successful prediction of these faults.

The best results for the generator heating faults by classical point-based performance metrics for the specified time windows given in Table 5.2 can be seen in Fig. 5.6.

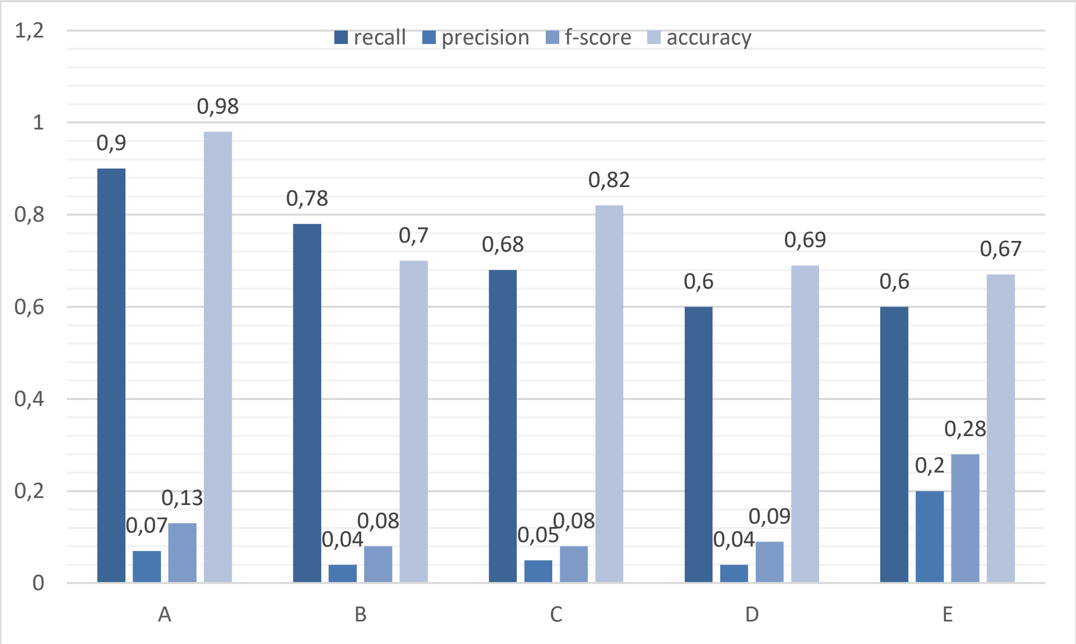


Figure 5.6 : Performance metrics for the fault prediction level.

Although recall values which are almost 1 were reached in some other ANN models for all the cases, the accuracy limit we considered decreased the balanced scores. Only the models where accuracy scores were higher than 0.65 were taken into the consideration for the selection of the models with the best recall values. For Case A, more than half of the samples in the minority class are missing due to imperfections of SCADA system which contributes to the low performance scores. As it is seen in Figure 5.6, with the increase of the duration of pre-fault time bands, recall value decreases approximately from 0.9 to 0.6. Precision score is less than 0.1 in most of the cases except for Case E where it is 0.2. The comparatively higher precision and f-score values show there are fault indicators as long as 48 hours before beginning of faults.

The classical performance metrics given in Figure 5.6 were calculated in a point-based manner which means the network outputs for all the time instances were evaluated independently to calculate the metric values. As a window-based interpretation is more appropriate for this level, such an implementation was realized to find out the effective prediction horizon. If a false alarm flag is raised after ensuring that the network claims a fault indicator for a specified time period, the number of false alarms decrease significantly. Besides, most of the faults are predicted in advance. With an appropriate threshold value for ANN outputs which was determined as 0.975, it was observed that 5 out of 7 generator heating faults in the test set were predicted successfully. The duration to monitor the persistency of values lower than threshold was selected as 12 hours. If ANN outputs are less than the threshold value for more than 12 hours, a fault alarm is raised. These parameters were determined empirically.

To better interpret the ANN outputs, fistly the response of the network for a region where no generator heating faults present in the test set is shown in Figure 5.7. 1000 instances from a no-fault region are given in the figure.

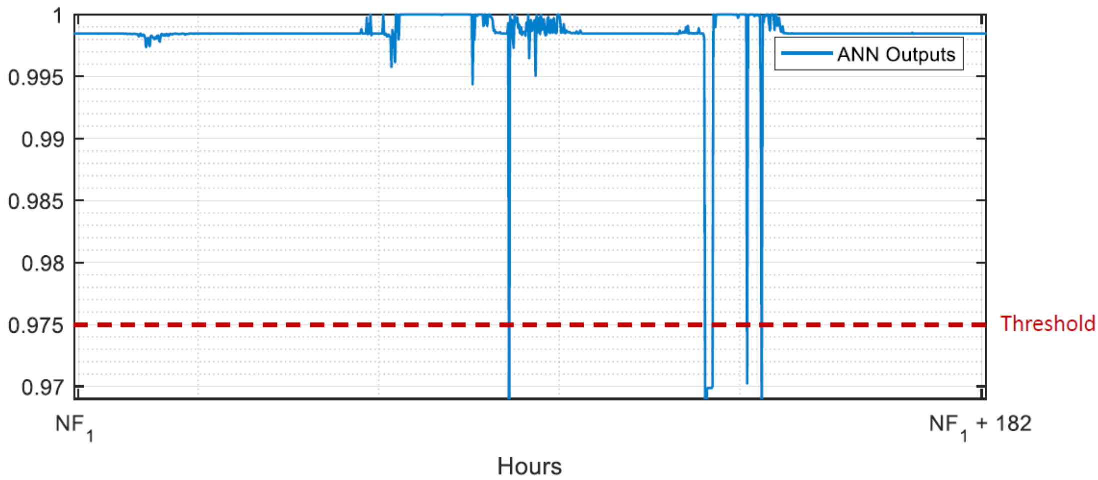


Figure 5.7 : ANN outputs for a section of normal operation region.

As can be seen in the figure, ANN outputs are close to 1 in almost every instance. Only for some single points, the outputs reach the thereshold value, however the window-based interpretation prevents a false alarm as the duration of these values are much lower than 12 hours. NF_1 in the x axis of the graph implies the starting point of the no-fault region given in this figure. If there were no missing values, 1000 instances of data shown would results in 166 hours of data would be present in this period. However, due to the missing data, between the last and the first samples shown, there are 182 hours of operation, therefore it was shown as NF_1+ 182 in the end of the section.

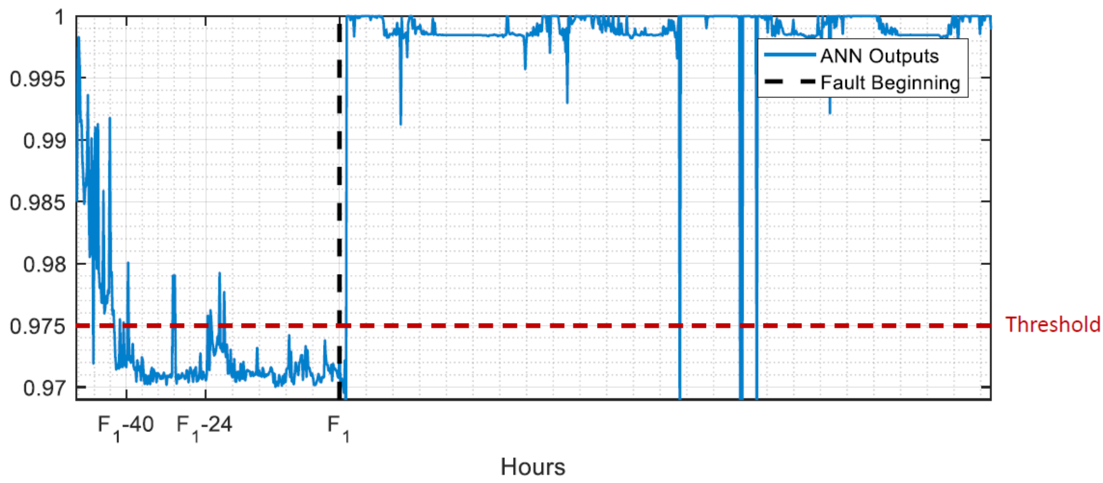


Figure 5.8 : ANN outputs and fault beginning instance for Fault 1.

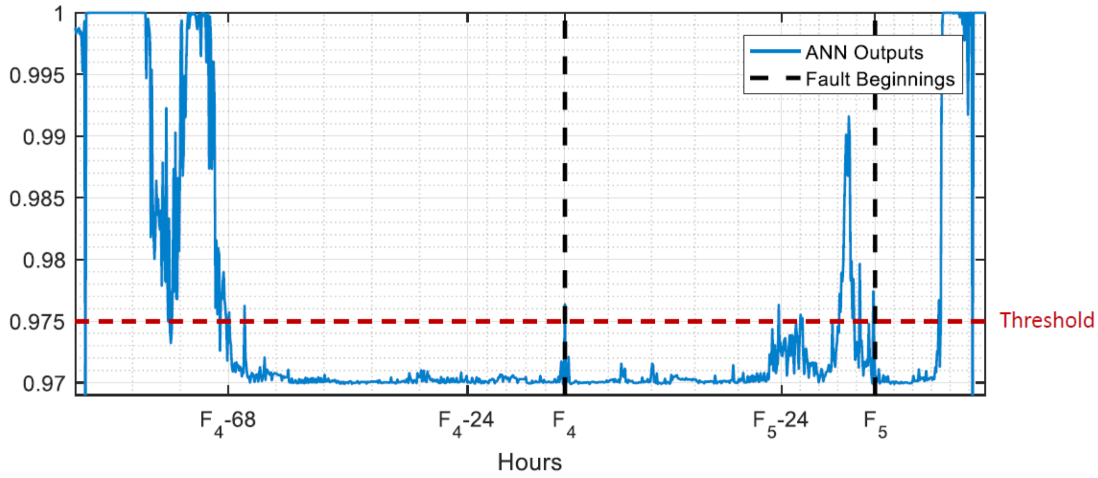


Figure 5.9 : ANN outputs and fault beginning instances for Faults 4 and 5.

Figures 5.8 and 5.9 show the ANN outputs and the fault beginning instances for the first, fourth and fifth generator heating faults in the test set for Case D which provided the best results for the window-based fault prediction analysis. F_1, F_4, and F_5 refer to the beginning of the first, fourth and the fifth faults respectively. Similar to Figure 5.7, Figure 5.8 and Figure 5.9 also display 1000 instances of operation.

When ANN outputs are closer to 1, the network claims that there is no fault. Decrease in ANN output implies that there might be indications of upcoming faults. As it is seen from the figures, the fault indications start to appear even earlier from the pre-fault time-bands (which was between 12-24 hours before fault beginnings for case D) and the network continue to show fault indications at least until the beginning of the faults. After the beginnings of fault occurrences, ANN output increases again. Figure 5.8 shows that the indicators of Fault 1 start 40 hours in advance. After monitoring the values for 12 hours, a fault alarm is raised which is 28 hours before the beginning of Fault 1. The alarm stopped 20 min after the fault ended.

As can be seen in Figure 5.9, there is a steady decrease in ANN outputs that begin 68 hours ahead of Fault 4. After the 12 hours monitoring period of the consistency of indication for this fault, it was predicted 56 hours in advance with the same approach. Fault 4 and Fault 5 occur in close time-proximity. There are 47 hours between the end of Fault 4 and the beginning of Fault 5. Therefore, it is hard to evaluate certainly how long in advance Fault 5 was predicted. Because, the indications of Fault 4 may continue for some time after it ends or indications for Fault 5 could start earlier than 47 hours which would be observable if Fault 4 did not exist. As a result, there is an uncertainty about the exact prediction instant for Fault 5.

After analyzing ANN outputs for all the test set, results of the fault prediction level for each generator heating fault are summarized in Table 5.3.

Table 5.3 : Prediction horizon for generator heating faults.

Fault no	Time of prediction in advance (hours
F1	28
F2	--
F3	--
F4	56
F5	~35
F6	44
F7	16

The other successful predictions are for Fault 6 and Fault 7 with 44 and 16 hours, respectively. Fault 2 and Fault 3 could not be predicted. During the whole test set period which covers approximately 3 months, 12 false generator heating alarms appeared.

The results of this part of the thesis show that, proposed methods and the use of SCADA data are beneficial in the detection, isolation and prediction of non-fatal but frequent wind turbine faults which are one of the reasons of the long downtime durations. Even though the data set used is comparatively limited as it does not contain some information which are typically involved in most SCADA systems, successful outcomes were acquired. High-performance scores in all three levels of the scheme that expand the current finding performances in the literature were obtained. The methods especially show a promising capacity in the prediction level. Most of the generator heating faults were predicted dozens of hours in advance which is a significant improvement in the prediction horizon comparing to the former studies on non-fatal wind turbine faults.

.

6. CONCLUSION

This thesis presents a predictive maintenance approach for wind turbines using artificial intelligence techniques. The aim is to design a cost-effective approach to predict wind turbine faults to increase amount of energy conversion and performance of the overall system. The proposed methods were developed by using the data collected from SCADA system of a wind turbine. A major advantage of this approach is that, as SCADA is a built-in part in many medium or large-scale modern wind turbines, no additional hardware costs are required for the real-world application of these methods which serves for the aim of designing a cost-effective system. Temperature data, operational data, status data and wind parameters are the types of information commonly available in SCADA systems which were also used in this thesis. As in line with expectations, data collected from the SCADA system includes many imperfections such as high rate of missing values and low sampling frequency. To handle these challenges and the highly non-linear behavior of wind turbines, ANN models were used as the core parts of the algorithms developed.

During the data collection period from the target wind turbine, serious faults that cause fatal results in the overall turbine or one of its main components did not happen. However, many non-fatal faults which typically occur frequently in most turbines were observed and the algorithms were developed to analyse and classify this type of faults. Weak indications of these faults make it a challenging task to detect or predict them. The results of former studies in this field show a clear distinction between the prediction performances of fatal and non-fatal faults. In fatal faults of main components, some predictions were successfully made months in advance, however prediction horizon of frequent, non-fatal faults are described in the scale of hours. These outcomes show the significance of the severity of indications in the prediction horizon of faults.

In the first part of the thesis, a validation method for wind turbine temperature sensors is presented. The problem was approached as a regression task where the deviations between the real temperature measurements and the estimated temperature outputs from ANN models were analysed for the diagnosis of a simulated sensor fault in the form of a calibration drift. It was observed that the simulated calibration drift in one of the sensors was successfully detected by the concurrent use of Auto-Associative and MISO ANN models. Various test cases were generated to examine the performance of the proposed algorithm on the distinction between a calibration drift and real temperature alterations that possibly happen due to environmental changes, and it was shown that the proposed algorithm was capable of distinguishing the faulty case from non-faulty cases. The results support the findings in the literature in different fields asserting that ANN models that were constructed solely using measurements of a group of related sensors as inputs can be used for the validation of these sensors. By this way, without hardware redundancy or dismantling the sensors to control their state of health, calibration process can be realized by continuously monitoring and evaluating measurements.

In the second part, a systematic pre-processing scheme with the aim of improving the performance of fault classification is presented. Using the raw features directly collected from the SCADA system and various data processing principles, additional features were generated to obtain inputs for ANN models that possibly give better information on hidden relations in the system. The new features were constructed in 4 different principles by using knowledge-based, statistical, time-series and difference characteristics of the original features. After obtaining the full set containing the raw and generated features, a hybrid feature selection algorithm that consists of filter and wrapper selection steps was employed to find out a successful subset characterizing the problem. The results of this part show that the methods applied in this level are effective on increasing the performance and reducing the computational time. The highest impact was seen in the precision metric. It was increased from 0.75 to 0.96 in the detection of generator heating faults which means the duration of false alarm was decreased from 210 to 30 mins in the 3 months test period. This result contributes to efforts aiming to decrease the false alarm rate of wind turbine fault detection systems which is a significant problem in systems designed using SCADA data.

The last part of the study presents the design of a 3-level fault classification system that aims to detect, isolate and predict frequent, non-fatal wind turbine faults. The fault types attempted in this level are the generator heating fault, mains failure and feeding faults. Other faults faced by the turbine was in insufficient amounts to train and test ANN models. In this part, the problem was handled as a classification problem in contrary to the sensor validation level where the ANN models were applied to solve a regression problem. The results obtained in the fault detection level are very promising in terms of accuracy and competitive towards the similar works in the literature. In fault isolation level, the performance metrics are even higher than the first level. One possible reason for this result can be the use of One-Against-All ANN structure that enables to select input features more flexible than the first level as each fault type have different indications than others. In the fault prediction level, only the generator heating faults were attempted as the other two frequent faults are related to outer circumstances and only monitoring the turbine parameters do not provide indications of these faults. The results for the prediction of generator heating faults show that the methods proposed are very effective. 5 out of 7 generator heating faults in the test set was successfully predicted and the prediction horizon was found as large as 56 hours which is a significant expansion to similar studies on generator heating faults in the literature.

In addition to the powerful computational characteristics of ANN models, achieving high performance metrics are also partly rooted from the implementation of the assistive methods. From this aspect, on top of feature construction and selection methods, techniques to handle the imbalance rate of the training data set were also proven to be effective on improving the results.

The results obtained in this thesis expand the performance of the findings in the current literature and support the fact that despite the shortcomings of SCADA data, they are useful to increase the reliability of wind turbines. It was shown that using SCADA data and ANN models with additional assistive methods, significant opportunities lie in the design of cost-effective fault detection systems.

A possible interesting topic for the future works can be spanning more techniques about feature engineering methods. The feature construction and selection methods attempted in this thesis were found to be useful. However, there are many other methods proposed in the literature that can be applied to further improve the results in the future works.

The most important limitation of this thesis lies in the fact that the results belong to a single wind turbine due to the non-availability of more comprehensive data sets. Therefore, investigating the generalizability of success of the proposed methods can be an important topic for future works. Similar methods proposed in this thesis can be applied to different turbines in distinct working environments to test the generalizability of the methods.

Another limitation is that in the sensor validation part, due to the absence of sensor fault information the work was carried out creating a simulated fault assuming that a calibration drift would result in a similar behavior that was generated in the test cases. The other parts of this thesis do not include this kind of a limitation and the data sets were not altered as they already include information on faults.

In terms of the quality of the data collected by the SCADA system used in this thesis, non-availability of some types of data that could have been beneficial to further improve the fault classification performance is one of the constraints faced in this work. For instance, electrical parameters like generator voltage, current and frequency are some of the basic SCADA parameters that could possibly be informative on generator heating faults. However, despite their absence, the results show a high potential. Therefore, the methods proposed in this thesis would have a chance of producing even better results by applying a more complete data set.

The high-performance results for the detection, isolation and early prediction of frequent wind turbine faults achieved in this thesis are very useful in two different aspects. First, they contribute to the transition from preventive to predictive maintenance approach for wind turbines which is important to reach an optimum point between cost and performance. Secondly, an early detection system based on the findings of this thesis would be effective in increasing the availability of wind turbines and with the help of the precautions that can be taken before faults occur, increase in energy conversion and system performance and preventing system degradation would become possible.

In addition to the technical findings, during this thesis work it was seen that, working towards creating platforms that support open data share policy in wind energy field is a significant topic to accelerate the technological developments in wind farm operations. By this way, more comprehensive comparisons between the methods presented in the literature can be realized and the industry would benefit from the knowledge of experts working in this field.

In terms of the practical applicability, the methods presented can be directly used in other wind turbines for the real-world applications. Moreover, the flexibility of the methodology allows its use in fault detection purposes for other systems with a sufficient amount of available data types and fault information.

As a result, this thesis contributes to works on improving wind turbine maintenance strategies by presenting a methodology that was designed by implementing data-driven methods that bring only computational costs without any additional hardware costs. The results indicate that it is a promising field to further investigate in the way of reducing costs of wind turbine operations and as a result contributing to the global renewable energy aims.

REFERENCES

[1] **International Energy Agency**. (2018). *World Energy Outlook 2018.*

[2] **United Nations**. (2013). *Framework Convention on Climate Change*.

[3] **Mcglade, C., and Ekins, P.** (2014). The geographical distribution of fossil fuels unused when limiting global warming to 2 °C. *Nature*, *517*(7533), 187-190.

[4] International Energy Agency. (2013). *Wind 2013 Roadmap*.

[5] International Energy Agency. (2018). *Renewables Information 2018*.

[6] Knutti, R., Rogelj, J., Sedlacek, J., Fischer, E. M. (2016). A scientific critique of the two-degree climate change target. *Nature Geoscience. 9*, 13-18.

[7] Dhillon, B. S. (2002). *Engineering Maintenance A Modern* Approach. CRC Press.

[8] Kusiak, A. (2016). Renewables: Share data on wind energy. *Nature*, *529*(7584), 19–21.

[9] Tautz-Weinert, J. and Watson, S. J. (2017). Using SCADA data for wind turbine condition monitoring – a review. *IET Renew. Power Gener.*, *11*(4), 382–394.

[10] Feng, Y., Qiu, Y., Crabtree C., Long H., and Tavner P. (2011). Use of SCADA and CMS signals for failure detection and diagnosis of a wind turbine gearbox. *EWEA 2011*, Brussels.

[11] Feng Y., Qiu Y., Crabtree, C. J., Long, H., and Tavner, P. J. (2013). Monitoring wind turbine gearboxes. *Wind Energy*, *16*(5), 728–740.

[12] Yang, W., Court, R. and Jiang J. (2013). Wind turbine condition monitoring by the approach of SCADA data analysis. *Renew. Energy*, *53*, 365–376.

[13] Astolfi, D., Castellani, F., and Terzi, L. (2014). Fault prevention and diagnosis through SCADA temperature data analysis of an onshore wind farm. *Diagnostyka*, *15*(2), 71–78.

[14] Kusiak, A. and Zhang Z. (2010). Analysis of Wind Turbine Vibrations Based on SCADA Data. *J. Sol. Energy Eng.*, *132*(3), 031008-1–031008-12.

[15] Zhang, Z. and Kusiak, A. (2012). Monitoring Wind Turbine Vibration Based on SCADA Data. *J. Sol. Energy Eng.*, *134*(2), 021004-1–021004-12.

[16] S. Catmull. (2011). Self Organizing Map Based Condition Monitoring of Wind Turbines. *European Wind Energy Association (EWEA) Annual Conference*, January 2011, Brussels.

[17] Kim, K., Parthasarathy, G., Uluyol, O., Foslien, W., Sheng, S., and Fleming, P. (2011). Use of SCADA Data for Failure Detection in Wind Turbines *ASME 2011 5th Int. Conf. Energy Sustain. Parts A, B, C*, no. October, 2071–2079.

[18] Wilkinson, M., Harman, K., van Delft, T. and Darnell, B. (2014). Comparison of methods for wind turbine condition monitoring with SCADA data. *IET Renew. Power Gener., 8*(4), 390–397.

[19] Garlick, W. G., Dixon, R. and Watson, S. J. (2009). A Model-based Approach to Wind Turbine Condition Monitoring using SCADA Data. *20th Int. Conf. Syst. Eng.*, 7-12.

[20] Cross P. and Ma X. (2015). Model-based and fuzzy logic approaches to condition monitoring of operational wind turbines. *Int. J. Autom. Comput.*, *12*(1), 25–34.

[21] Schlechtingen, M. and Santos, I. F. (2014). Wind turbine condition monitoring based on SCADA data using normal behavior models. Part 2: Application examples. *Appl. Soft Comput., 14*, 447–460.

[22] Zaher, A., McArthur, S. D. J., Infield, D. G. and Patel, Y. (2009). Online wind turbine fault detection through automated SCADA data analysis. *Wind Energy, 12*(6), 574–593.

[23] Kusiak, A. and Verma, A. (2012). Analyzing bearing faults in wind turbines: A data-mining approach," *Renew. Energy*, vol. 48, 110–116.

[24] Wang, Y. and Infield, D. (2013). Supervisory control and data acquisition data-based non-linear state estimation technique for wind turbine gearbox condition monitoring. *IET Renew. Power Gener., 7*(4), 350–358.

[25] Zhang, Z. Y. and Wang, K. S. (2014). Wind turbine fault detection based on SCADA data analysis using ANN. *Adv. Manuf.*, *2*(1), 70–78.

[26] Breteler, D., Kaidis, C., Tinga, T. and Loendersloot, R. (2015). Physics based methodology for wind turbine failure detection, diagnostics & prognostics. *Ewea 2015*, 1–9.

[27] Borchersen, A. B. and Kinnaert, M. (2015). Model-based fault detection for generator cooling system in wind turbines using SCADA data. *Wind Energy.,19*(4), 593-606.

[28] Chen, B., Qiu, Y. N., Feng, Y., Tavner, P. J. and Song, W. W. (2011). Wind turbine SCADA alarm pattern recognition. in *IET Conf. Renewable Power Generation (RPG 2011)*, 2011, 363–368.

[29] Chen, B., Tavner, P. J., Feng, Y., Song, W. W. and Qiu, Y. N. (2012). Bayesian network for wind turbine fault diagnosis. *EWEA 2012*, 16-19 April, Copenhagen.

[30] Kusiak, A. and Li, W. (2011). The prediction and diagnosis of wind turbine faults. *Renew. Energy, 36*(1), 16–23.

[31] Leahy, K., Hu, R. L., Konstantakopoulos, I. C., Spanos, C. J., Agogino, A. M. and O'Sullivan, D. T. J. (2018). Diagnosing and Predicting Wind Turbine Faults from SCADA Data Using Support Vector Machines. *Int. J. Progn. Heal. Manag.*, *9*(1), 1–11.

[32] Leahy, K. B., Gallagher, C., O'Donnovan, P. (2018). A Robust Prescriptive Framework and Performance Metric for Diagnosing and Predicting Wind Turbine Faults based on SCADA and Alarms Data with Case Study. *Energies. 11*(1738).

[33] Hu, R. L., Leahy, K., Konstantakopoulos, I. C., Auslander, D. M., Spanos, C. J., and Agogino, A. M. (2017). Using domain knowledge features for wind turbine diagnostics. *Proc. - 2016 15th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2016*, 300–305.

[34] Li, Y., Liu, S., and Shu, L. (2019). Wind turbine fault diagnosis based on Gaussian process classifiers applied to operational data. *Renew. Energy*, *134*, 357–366.

[35] Godwin, J. L., and Matthews, P. (2014). Rapid labelling of SCADA data to extract transparent rules using RIPPER. *Proc. - Annu. Reliab. Maintainab. Symp.*

[36] Leahy, K., Hu, R. L., Konstantakopoulos, I. C., Spanos, C. J. and Agogino, A. M. (2016). Diagnosing wind turbine faults using machine learning techniques applied to operational data. *2016 IEEE Int. Conf. Progn. Heal. Manag.*, 1–8.

[37] Reder, M. D., Gonzalez, E. and Melero, J. J. (2016). Wind turbine failures - Tackling current problems in failure data analysis. *J. Phys. Conf. Ser*ies. 2016/09.

[38] Krose, B. and Van Der Smagt, P. (1996). *An introduction to Neural Networks*. 8th ed. University of Amsterdam.

[39] Haykin, S. (2009). *Neural Networks and Learning Machines*. Prentice Hall.

[40] Nunes, I. and Hernane, D. Flauzino, R. A., Liboni, L. H. B., Reis Alves, S. F. (2017). *Artificial Neural Networks*. Springer.

[41] Rosenblatt, F. (1962). *Principles of Neurodynamics : Perceptrons and the Theory of Brain Mechanisms*. Washington DC: Spartan Books.

[42] Widrow, B. and Hoff, M. E. (1962). Associative Storage And Retrieval of Digital Information in Networks of Adaptive 'Neurons,'. *Biol. Prototypes Synth. Syst.*, 1962, 1962.

[43] Widrow, B. and Lehr, M. A. (1990). 30 Years of Adaptive Neural Networks : Perceptron , Madaline , and Backpropagation. *Proc. IEEE*, *78*(9), 1415–1442.

[44] Ruck, D. W., Rogers, S. K., Kabrisky, M., Oxley, M. E. and Suter, B. W. (1990). The Multilayer Perceptron as an Approximation to a Bayes Optimal Discriminant Function. *IEEE Trans. Neural Networks*, *I*(4), 296–298.

[45] Specht, D. F. (1991). Probabilistic Neural Networks. *Neural Networks*, *3*, 109-118.

[46] Park, J. and Sandberg, I. W. (1991). Universal Approximation Using Radial-Basis-Function Networks. *Neural Computation*, *3*, 246–257.

[47] Specht, D. F. (1991). A General Regression Neural Network. *IEEE Trans. Neural Networks*, *2*(6), 568–576.

[48] Kohonen, T. (1990). The Self-Organizing Map. *Proc. IEEE*, *78*(9), 1464–1480.

[49] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Biophysics. 79*(April), 2554–2558.

[50] Elman, J. (1990). Finding Structure in Time. *Cognitive Science, 211*, 179–211.

[51] Jordan, M. I. (1986). Serial order: A parallel distributed processing approach. Tech. Rep. No. 8604. University of California, Institute for Cognitive Science.

[52] Gopalsamy, K. (1994). Delay-Independent Stability in Bidirectional Associative Memory Networks. *IEEE Transactions on Neural Networks, 5*(6), 998-1002.

[53] Carpenter, G. A., Grossberg, S. and Reynolds, J. H. (1995). A Fuzzy ARTMAP Nonparametric Probability Estimator for Nonstationary Pattern Recognition Problems. *IEEE Transactions on Neural Networks*, *6*(6), 1330-6.

[54] Hochreiter, S. and Schmidhuber, J.(1997). Long Short-Term Memory. *Neural Computation*, *9*, 1735–1780.

[55] Jaeger, H. (2010). The ‘echo state’ approach to analysing and training recurrent neural networks – with an Erratum note. GMD report. 1–47.

[56] Sutskever, I. (2013). *Training Recurrent Neural Networks.* (PhD Thesis).

[57] Hornik, K., Stinchcombe, M. and White, H. (1989). Multilayer Feedforward Neural Networks are Universal Approximators. *Neural Networks, 2,* 359–366.

[58] Hagan, M. T., Demuth, H. B., Beale, M. H. and De Jesus, O. (1996). *Neural Network Design*, PWS.

[59] Marquardt, D. W. (1963). An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *SIAM J. Appl. Math.*, *11*(2), 431–441.

[60] Nelles, O. (2010). *Nonlinear System Identification - From Classical Approaches to Neural Networks and Fuzzy Models*. Berlin: Springer-Verlag Berlin Heidelberg.

[61] Bartzoudis, N. and Mcdonald-Maier, K. (2007). An embedded sensor validation system for adaptive condition monitoring of a wind farm. 2nd NASA/ESA Conference on Adaptive Hardware and Systems.

[62] Kramer, M. A. (1992). AutoAssociative Neural Networks. *Comput. Chem. Eng.*, *16*(4), 313–328.

[63] Hines, J. W., Uhrig, R. E. and Wrest, D. J. (1997). Use of Autoassociative Neural Networks for Signal Validation. *Neural Network Application (NEURAP),* Marseille, France.

[64] Upadhyaya B. R. and Eryurek, E. (1992). Application of Neural Networks for Sensor Validation and Plant Monitoring *Nucl. Technol.*, *97*(2), 170-176.

[65] Barutçu, B., Şeker, S., Ayaz, E. and Türkcan, E. (2003). Real time reactor noise diagnostics for the Borssele (PWR) nuclear power plant. *Prog. Nucl. Energy*, *43*(1), 137–143.

[66] Yu, D. L., Gomm, J. B. and Williams, D. (1999). Sensor fault diagnosis in a chemical process via RBF neural networks. *Control Eng. Practise. 7,* 49–55.

[67] Palmé, T., Fast, M. and Thern, M. (2011). Gas turbine sensor validation through classification with artificial neural networks. *Appl. En., 88,* 3898–3904.

[68] Mattern, L. (1998). Using Neural Networks for Sensor Validation. *34th Joint Propolsion Conference*. Celeveland, Ohio.

[69] Bakir, T. and Boussaid, B. (2016). WANN fault detection and isolation approach design for a FAST wind turbine benchmark model. *3rd International Conference on Automation, Control, Engineering and Computer Science,* 657-681.

[70] Simani, S. and Turhan, C. (2018). Fault Diagnosis of a Wind Turbine Simulated Model via Neural Networks. *IFAC-PapersOnLine*, *51*(24), 381–388.

[71] Simani, S. and Paolo, C. (2019). Intelligent Fault Diagnosis Techniques Applied to an Offshore Wind Turbine System, *Appl. Sci, 9*(4), 783.

[72] Odgaard, P. F., Stoustrup, J. and Kinnaert, M. (2013). Fault-Tolerant Control of Wind Turbines : A Benchmark Model. *IEEE Trans. Control Syst. Technol., 21*(4), 1168–1182.

[73] Liu, H. and Motoda, H. (1998). *Feature Extraction, Construction and Selection: A Data Mining Perspective*. New York: Springer Science Business Media.

[74] Bellman, R. (1966). Dynamic Programming. *Management Science., 12,* 317-348.

[75] J. Tang, S. Alelyani, and H. Liu. (2014). Feature Selection for Classification: A Review. *Data Classif. Algorithms Appl.*, 37–64.

[76] Liu, H. and Motoda, H. (2007). Computational Methods of Feature Selection. Chapman&Hall.

[77] Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artif. Intell., 97*(1–2), 273–324.

[78] Dash, M. and Liu, H. (1997). Feature Selection for Classification. *Intell. Data Anal., 1*(97), 31–156.

[79] Sta, U. and Jain, L. C. (2015). *Feature Selection for Data and Pattern Recognition*, vol. 584, Springer.

[80] Bolón-Canedo, V. and Alonso-Betanzos, A. (2018). *Recent Advances in Ensembles for Feature Selection*, Springer.

[81] Peng, H., Long, F. and Ding, C. (2005). Feature selection based on mutual information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. *IEEE Trans. Pattern Anal. Mach. Intell., 27*(8), 1226–1238.

[82] Hall, M. (1999). *Correlation-based Feature Selection for Machine Learning*. (PhD Thesis). University of Sydney.

[83] Duda, R. O., Hart, P. E. and Stork, D. G. (2001). *Pattern Classification*. John Wiley & Sons, New York.

[84] Gu, Q., Li, Z and Han, J. (2011). Generalized Fisher Score for Feature Selection. *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence (UAI)*, 266-273, Barcelona, Spain.

[85] Robnik-Sikonja, M. and Kononenko, I. (2003). Theoretical and empirical analysis of ReliefF and RRiefF. *J. Mach. Learn. Res.*, *53*, 23–69.

[86] He, X., Cai, D. And Niyogi, P. (2005). Laplacian Score for Feature Selection. *Proceedings of the 18th International Conference on Neural Information Processing Systems (NIPS)*, 507-514, Vancouver, Canada.

[87] Smola, A., Gretton, A. and Bedo, J. (2007). Supervised Feature Selection via Dependence Estimation *Proceedings of the 24th international conference on Machine learning (ICML)*, 823-830, Oregon, USA.

[88] Nie, F., Xiang, S., Jia, Y., Zhang, C. and Yan, S. (2008). Trace Ratio Criterion for Feature Selection. *23rd AAAI Conference on Artificial Intelligence*, 671–676.

[89] Zhao, Z. and Liu, H. (2012). *Spectral Feature Selection for Data Mining.* CRC Press.

[90] Liu, H. and Motoda, H. (1998) *Feature Selection for Knowledge Discovery and Data Mining.* Kluwer Academic Publishers.

[91] Molina, L. C., Belanche, L. and Nebot, À. (2001). Feature Selection Algorithms: A Survey and Experimental Evaluation. *Proceedings. 2002 IEEE Int. Conf. Data Min. (ICDM)*, 306–313.

[92] Pudil, P., Novovičová, J. and Kittler, J. (1994). Floating search methods in feature selection. *Pattern Recognit. Lett.*, *15*(11), 1119–1125.

[93] Räsänen, O. and Pohjalainen, J. (2013). Random subset feature selection in automatic recognition of developmental disorders, affective states, and level of conflict from speech. *Interspeech*, 2013, August, 210–214.

[94] Siedlecki, W. and Sklansky, J. (1988). On Automatic Feature Selection. *Intertional J. Pattern Recognit. Artif. Intell.*, 2, 197–220.

[95] Chawla, N. V. (2009). Data Mining for Imbalanced Datasets: An Overview. *Data Min. Knowl. Discov. Handb.*, 875–886.

[96] Chawla, N. V., Bowyer, K. W., Hall, L. O. and Kegelmeyer, W. P. (2002). SMOTE : Synthetic Minority Over-sampling Technique. *Journal of Artificial Int. Research, 16*, 321-357.

CURRICULUM VITAE

Name Surname: Ayşe Gökçen Kavaz Taşkıner

E-Mail: agkavaz@gmail.com

EDUCATION:

B.Sc.: 2009, Yeditepe University, Systems Engineering (Control&Automation Engineering Branch)

M.Sc.: 2012, Istanbul Technical University, Energy Science and Technology Programme

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2010-2015 : Research Assistant in İstanbul Technical University, Energy Institute
- 2014-2015 : Visiting Research in Technical University of Denmark, Applied Mathematics and Computer Science
- 2015-Present : Senior Guidance, Navigation and Control Engineer in Roketsan.
- 2014 : Awarded a one-year scholarship grant (2214-A) from the The Scientific and Technological Research Council of Turkey (TUBITAK).

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Kavaz A.G.**, Barutcu B., "Fault Detection of Wind Turbine Sensors Using Artificial Neural Networks", Journal of Sensors, 2018.
- **Kavaz A.G.**, Barutcu B., "Fault Detection, Isolation and Prediction of Wind Turbines by Artificial Neural Networks Using SCADA Data" (Under Review)
- **Kavaz A.G**., Barutcu B., "A Hybrid Feature Selection and Construction Method for Detection of Wind Turbine Generator Heating Faults" (Under Review)
- **Kavaz A. G.**, Barutcu B., "Feature Selection and Extraction Methods for Fault Detection in Wind Turbines", The International Conference on Pattern Recognition on Artificial Intelligence, August 15-17, 2018, New Jersey, USA

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS :

- **Kavaz A. G.**, Tasan C., "IMU Error Compensation for Rotating Missiles by Artificial Intelligence Methods", The International Conference on Pattern Recognition on Artificial Intelligence, August 15-17, 2018, New Jersey, USA.

- **Kavaz A. G.**, Barutcu B., "Time Delay Algorithms for Wind Turbine Yaw Control System", 5th International Congress of Energy and Environment Engineering and Management (CIIEM), July 17-19, 2013, Lisbon, Portugal.