TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

TUUKKA HARMAALA

GAS TURBINE POWER PLANT BENCHMARKING AND OPTIMIZATION WITH MACHINE LEARNING IN INDUSTRIAL INTERNET ENVIRONMENT

Master of Science Thesis

Examiner: prof. Risto Ritala
Examiner and topic approved on 29 November 2017

ABSTRACT

TUUKKA HARMAALA: Gas Turbine Power Plant Benchmarking and Optimization with Machine Learning in Industrial Internet environment
Tampere University of technology
Master of Science Thesis, 56 pages, 1 Appendix page
February 2018
Master's Degree Programme in Automation Technology
Major: Process Automation
Examiner: Professor Risto Ritala

Keywords: Gas turbine power plant, machine learning, linear regression, stepwise regression, Industrial Internet

For past five to ten years, the industry has been investing more and more in Industrial Internet. Industrial Internet is changing the whole industrial segment and it creates new opportunities for companies to grow their business. Industrial Internet allows users to combine multiple plants into one big ecosystem where the plants can exploit the information provided by the other plants.

This thesis combines gas turbine domain, machine learning and Industrial Internet together. Aim of this thesis was to develop a machine learning model and deploy it to Industrial Internet environment. The thesis is a proof of concept and it works as a base for developing the future applications.

The machine learning model predicts temperature corrected power output of a gas turbine. With the model, it is possible to point out a performance decrease in the turbine. The model was developed using stepwise regression method. The model was trained to work only on a base load.

The whole process from integrating data to the visualizations for the end user was implemented in this thesis. The work was implemented in Valmet Industrial Internet platform. In the thesis, there were data from two plants both having two gas turbines. All the turbines are the same model so benchmarking the turbines between each other is reasonable.

The created model calculates predictions of temperature corrected power output of the turbine and returns the predictions to the database. The data is visualized. As a result, a user can examine the performance of the turbines. The user interface provides a general view where the user can look overall performance figures of the day. User interface also provides more detailed data view where the user can look the data from a chosen hour.

TIIVISTELMÄ

TUUKKA HARMAALA: Kaasuturbiinilaitoksen vertailu ja optimointi käyttäen koneoppimista teollisen internetin ympäristössä
Tampereen teknillinen yliopisto
Diplomityö, 56 sivua, 1 liitesivu
Helmikuu 2018
Automaatiotekniikan diplomi-insinöörin tutkinto-ohjelma
Pääaine: Prosessien hallinta
Tarkastaja: professori Risto Ritala

Avainsanat: Kaasuturbiinilaitos, koneoppiminen, lineaarinen regressio, askeltava regressio, teollinen internet

Viimeisen viiden–kymmenen vuoden aikana teollisuus on sijoittanut kasvavissa määrin teolliseen internetiin. Teollinen internet muuttaa koko teollista segmenttiä ja luo yhtiöille uusia mahdollisuuksia kasvattaa heidän liiketoimintaansa. Teollinen internet mahdollistaa käyttäjiään yhdistämään monia laitoksia yhdeksi isoksi ekosysteemiksi, jossa laitokset voivat hyödyntää toisista laitoksista saatavaa informaatiota.

Tämä opinnäytetyö yhdistää kaasuturbiinit, koneoppimisen ja teollisen internetin yhteen. Opinnäytetyön tavoitteena on kehittää koneoppimismalli ja ottaa se käyttöön teollisen internetin ympäristössä. Opinnäytetyö toimii koetoteutuksena, jonka pohjalta tulevaisuuden sovelluksia voidaan kehittää.

Koneoppimismalli ennustaa lämpötilakorjattua kaasuturbiinin tehon ulostuloa. Mallin avulla on mahdollista osoittaa suorituskyvyn heikkeneminen turbiinissa. Malli on kehitetty käyttäen askeltavan regression metodia. Malli on koulutettu toimimaan ainoastaan pohjakuormalla.

Työssä on toteutettu koko prosessi datan yhdistämisestä loppukäyttäjälle tarkoitettuun visualisointiin. Työ on toteutettu Valmet Industrial Internet alustalla. Opinnäytetyössä käytettiin dataa kahdelta laitokselta, joissa molemmissa oli kaksi kaasuturbiinia. Kaikki turbiinit ovat malliltaan samoja, joten niiden vertaaminen keskenään on järkevää.

Luotu malli laskee ennusteita turbiinin lämpötilakorjatulle tehon ulostulolle ja palauttaa ennusteet tietokantaan. Data visualisoidaan. Tuloksena käyttäjä voi tarkastella turbiinien suorituskykyä. Käyttöliittymä tarjoaa yleisnäkymän, jossa käyttäjä voi katsoa yleisiä suorituskykyä kuvaavia lukuja päiväkohtaisesti. Käyttöliittymässä on myös yksityiskohtaisempaa dataa tarjoava näkymä, jossa käyttäjä voi tarkastella dataa valitulta tunnilta.

PREFACE

Learning new is an endless journey. I did not choose the subject for this thesis because it was something I was already good at. I chose it because I saw an opportunity to develop and challenge myself. Like many times in life, I found that even after a difficult start, it is possible to clear obstacles and cross the finishing line.

There are many people that deserve my acknowledgments. The whole Industrial Internet platform was so complex that I could not have managed to understand it without the help of Atte Nopanen, Pasi Virtanen and Antti Nissinen. Also, my mentors Jussi Lautala and Johan Musch deserve thanks for their support on my thesis. There were times when I had no idea how to proceed or in what direction I should take my thesis. Those times Risto Ritala offered invaluable help. With the help of my family, Maaret and Karri (quadruped furry friend), I could forget the stress caused by thesis home. Thanks to my parents for good upbringing.

Thesis work was a good intermediate stage between the studies and work life. There are scenarios when the thesis is passed into the background and the “real work” starts to dominate the workdays. For me, finishing the thesis work was important so I can end one chapter of my life and fully start a new one after that.

Tampere, 28.2.2018

Tuukka Harmaala

CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS

Term	Explanation	Unit
β	parameter	
γ	ratio of specific heats	
ε	error term	
η	efficiency	
$\dot{m}$	mass flow	kg/s
h	specific enthalpy	J/kg
LHV	lower heating value	J/kg
Q	heat	W
r	pressure ratio	
T	temperature	K
W	work	W
X	predictor variable	
Y	predicted value	

ADFS	Active Directory Federation Services
API	Application Programming Interface
BI	Business Intelligence
CPPS	Cyberphysical Production System
CSV	Comma Separated Values
DDL	Data Definition Language
DV	Data Vault
EFS	Amazon Elastic File System
ELM	Extreme Learning Machine
ETL	Extract, Transform, Load
GT	Gas Turbine
GUI	Graphical User Interface
HRSG	Heat Recovery Steam Generator
IaaS	Infrastructure as a Service
IGV	Inlet Guide Vane
IIC	Industrial Internet Consortium
IIoT	Industrial Internet of Things
IIS	Industrial Internet Systems
IoT	Internet of Things
IP	Internet Protocol
JDBC	Java Database Connectivity
KPI	Key Performance Indicator
M2M	Machine-to-Machine
ML	Machine Learning
ODBC	Open Database Connectivity
PaaS	Platform as a Service
RBAC	Role Based Access Control
RFID	Radio-Frequency Identification
S3	Amazon Simple Storage Service
SaaS	Software as a Service

SCADA	Supervisory Control and Data Acquisition
SFTP	SSH File Transfer Protocol
SNS	Amazon Simple Notification Service
SQS	Amazon Simple Queue Service
SSH	Secure Shell
SVM	Support Vector Machine
VII	Valmet Industrial Internet
VPC	Valmet Performance Center
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network

1. INTRODUCTION

In a modern world, the power production is in very important role. The electricity generation has almost quadrupled during last 40 years while fuel consumption has more than doubled [1].

To fulfill the growing demand for power, the power plants must be efficient and reliable. Not only mechanical design has to be excellent, but also an optimal use of machinery is essential.

The amount of data available is continuously increasing and devices are more and more often connected to Internet. Internet of Things (IoT) and Industrial Internet are concepts that many companies currently work on. As a concept, Industrial Internet means that devices or "things" are connected to Internet and can produce data about themselves or their environment. Efficient use of data is important now when there is more data available than ever before.

This thesis combines these two subjects: power production and Industrial Internet. The aim is to find a solution to utilize modern tools such as machine learning and Industrial Internet to improve gas turbine performance and operability.

1.1 Motivation

Power plants generate huge amount of data. There are hundreds or even thousands of measurements from each plant. Some of the measurements produce multiple observations or values in one second. That means that annually there are millions of rows of data. Usually large amount of data is called big data. Big data includes volume (amount of data), velocity (lots of data coming in in a short time) and variety (measurement data, emails, pictures, videos etc.). Often, the data is unstructured. That creates a need for better real-time analysis. The analysis gives an opportunity to find new hidden information from the data [2].

To be able to use available data efficiently, automated methods are needed for data analysis. Machine learning is a high-level term for a set of methods that allows users to predict or to make decisions based on automatically detected patterns in data [3].

In gas turbine domain, different methods of machine learning have been used to improve the performance of gas turbine engines. Regression models [4], [5] are used for predicting

load, gas flow and engine performance such as compressor efficiency. Bayesian Hierarchical Model [6] is used to predict the remaining lifetime of a turbine. For optimizing the turbine combustor performance, hill-climbing and downhill simplex algorithms were used to optimize the controller [7]. Neural networks [8], [9] have been utilized for engine degradation prediction, health monitoring and prognosis and for fault detection and isolation. For fault diagnosis, support vector machine (SVM) has worked well [10], [11]. To detect combustor anomalies, extreme learning machine (ELM) is used [12].

This very brief literary survey shows, that in a gas turbine domain there is a wide variety of machine learning applications. Machine learning combined with Industrial Internet upgrades turbine power management beyond traditional plant SCADA (Supervisory Control and Data Acquisition). The data can be connected to a cloud server and the analysis and the calculations can be done remotely. Benchmarking against other plants with similar engines can be done if all the plants are connected to the same system.

Utilizing data to better fulfill the needs of a company or its customer is an asset. Data accessibility must be easy but safe. Analysis and data discovery should be implemented so that they are easy for the end user. Industrial Internet of Things is one key element when trying to fulfill those requirements.

1.2 Objectives for thesis

The aim of this thesis is to find a solution how to apply machine learning techniques in Industrial Internet of Things environment. This solution is a proof of concept. Based on the solution, future machine learning applications can be developed to Industrial Internet of Things environment.

With machine learning, the purpose is to create a model that predicts produced power output of a gas turbine based on the inputs to the model. The model is trained with data that is selected from the time period when the turbine is performing well. The performance of the turbine decreases over time. Thus, with the model, it should be possible to point out the decreased performance. When the performance decreases, the prediction of the power output should be higher than the actual power output.

The state of the thesis work and future possibilities are also discussed. After the proof concept is ready, the development of further applications will be easier and the amount of work smaller because the developer does not have to do all the study and groundwork to get started. In this thesis, only one model is developed but the possibilities are unlimited to utilize machine learning in data analysis area.

2. GAS TURBINE

Gas turbine cycle converts thermal energy into mechanical energy to generate power. The thermal energy is generated by burning fuel in a combustor. Energy is transformed to mechanical energy by a turbine. Mechanical energy is furthermore transformed to electric power in a generator or into other energy forms with a mechanical drive.

A gas turbine cycle can be divided into three main stages. The first stage is compression in which ambient air is led to a compressor. In the compressor, the air pressure and temperature rises. In the next stage, the compressed air is mixed with fuel and combusted in a combustor. The last stage is a turbine where the combusted air-fuel mixture expands and rotates the turbine.

Section 2.1 is a short overview of gas turbine development. After that, the very basic turbine cycle is introduced in Section 2.2. Next, the components of a turbine process are introduced in Sections 2.3-2.5. As this work is not focusing on a mechanical design of a turbine it will not be discussed. At the end of this chapter, there is an overview of the type of turbine that is used on the sites studied in this work. The chapter concludes by presenting the key performance indicators (KPIs) of turbine performance.

2.1 Overview of Gas Turbine Development

The first invention that had the basics of the modern gas turbine was created in 1791 by John Barber. The components were the same as in the modern gas turbines: a compressor, a combustion chamber and a turbine. The main difference was the compressor type: Barber used a chain-driven reciprocating type compressor [13].

In 1930 Frank Whittle built a gas turbine that is esteemed as the father of modern gas turbines. It had a centrifugal compressor and a radial-inflow turbine. In 1941 General Electric modified Whittle engine for the first aero-engine.

Gas turbines have been developed considerably during the last 20 years. With new materials and technologies compressor pressure ratio has increased from 7:1 to 45:1 as its highest. Simple-cycle gas turbine thermal efficiency has increased from 15% to 45% [13].

2.2 The Brayton Cycle

The Brayton cycle consists of two isobaric (constant pressure) and two isentropic (constant entropy) processes. The process is ideal if we consider that there are no losses in the turbine or compressor and the gas is calorically and thermally perfect. That means that

gas specific heat at constant pressure and the specific heat at constant volume are constant. Thus the heat ratio γ is constant throughout the cycle [13].

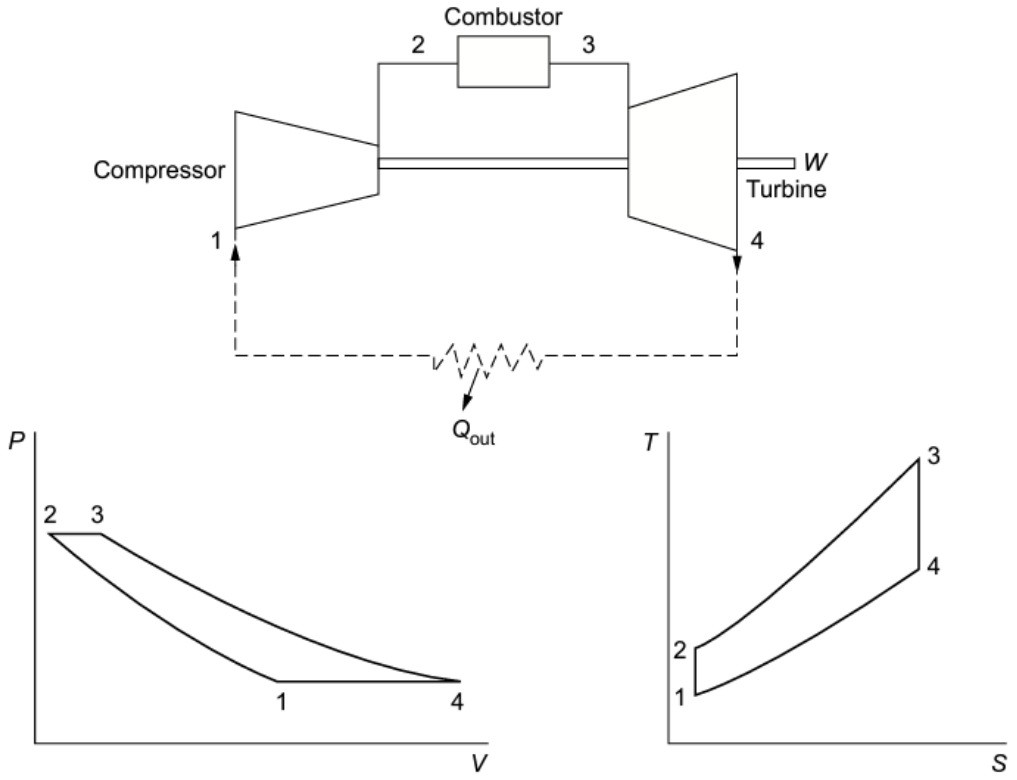

Figure 1. *The Brayton Cycle.* [13, p. 90]

The Brayton cycle is drawn in Figure 1. In the first stage (1→2) air is compressed in a compressor. Next, the air and fuel are combusted in a combustor (2→3). The work of a turbine is in stage 3→4. Finally, the heat recovery steam generator (HRSG) recovers heat from hot gas (4→1).

To make thermodynamic calculations, further assumptions are needed. Working fluid is assumed to be plain air and no chemical transformations happen during combustion. With that assumption, fuel combustion process is replaced by heat transfer process at a constant pressure. Also, exhaust and admission processes are replaced by a heat transfer process. Now the process is a closed cycle [14].

Assuming there are no changes in kinetic and potential energies and all components operate at 100% efficiency, the total work of the cycle can be explained as follows [13]:

Work of compressor:

$$W_C = \dot{m}_a(h_2 - h_1) \qquad (2\text{-}1),$$

where $\dot{m}_a$ is mass flow of air and h_1 is specific enthalpy before the compressor and h_2 is specific enthalpy after the compressor.

Work of turbine:

$$W_t = (\dot{m}_a + \dot{m}_f)(h_3 - h_4) \tag{2-2},$$

where $\dot{m}_f$ is mass flow of fuel, h_3 is specific enthalpy before the turbine and h_4 is specific enthalpy after the turbine.

Total output work:

$$W_{cyc} = W_t - W_c \tag{2-3}$$

Heat added to the system:

$$Q_{2,3} = \dot{m}_f LHV_{fuel} = (\dot{m}_a + \dot{m}_f)(h_3) - \dot{m}_a h_2 \tag{2-4},$$

where LHV_{fuel} is lower heating value of the fuel. Overall adiabatic thermal cycle efficiency is:

$$\eta_{cyc} = \frac{W_{cyc}}{Q_{2,3}} \tag{2-5}$$

Brayton cycle's adiabatic thermal efficiency can be increased by increasing the pressure ratio and the turbine firing temperature. With the assumptions made above, the relationship between the ideal adiabatic thermal cycle efficiency and pressure ratio for the ideal Brayton cycle can be written as [13]:

$$\eta_{ideal} = \left(1 - \frac{1}{r_p^{\frac{\gamma-1}{\gamma}}}\right) \tag{2-6},$$

where r_p is the pressure ratio and γ is the ratio of the specific heats. With the assumption that pressure ratio is the same in the compressor and the turbine, the ideal efficiency can be expressed with following relationships:

With pressure ratio in the compressor:

$$\eta_{ideal} = 1 - \frac{T_1}{T_2} \tag{2-7}$$

With pressure ratio in the turbine:

$$\eta_{ideal} = \frac{T_4}{T_3} \tag{2-8}$$

In the actual cycle, the losses must be considered. If efficiencies of compressor (η_c) and turbine (η_t) and difference between firing temperature T_f and the ambient temperature T_{amb} are considered, the efficiency of the cycle can be expressed as [13]:

$$\eta_{cycle} = \left(\frac{\eta_t T_f - \frac{T_{amb} r_p^{\left(\frac{\gamma-1}{\gamma}\right)}}{\eta_c}}{T_f - T_{amb} - T_{amb}\left(\frac{r_p^{\left(\frac{\gamma-1}{\gamma}\right)} - 1}{\eta_c}\right)} \right) \left(1 - \frac{1}{r_p^{\left(\frac{\gamma-1}{\gamma}\right)}} \right) \tag{2-9}$$

Figure 2 shows the effect of firing temperature and pressure ratio to the cycle overall efficiency. The efficiencies were calculated with 92% turbine efficiency and 87% compressor efficiency. The ratio of specific heats is 1.4 and ambient temperature is 0°C.

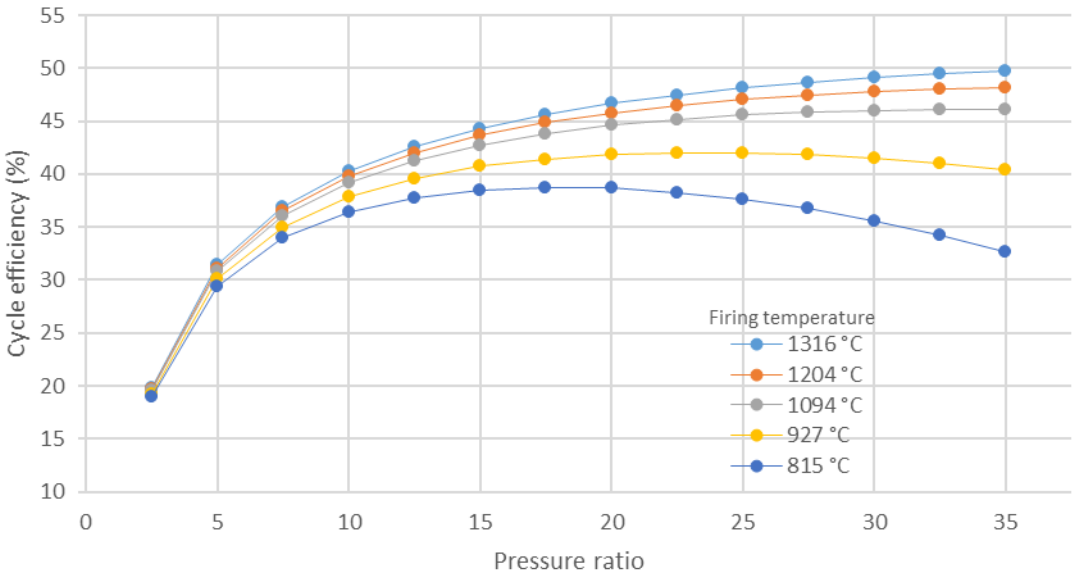

Figure 2. *An overall cycle efficiency as a function of pressure ratio and firing temperature. Adapted* from [13, p. 91].

After a certain value, the increase of the pressure ratio causes descending in overall cycle efficiency, regardless of firing temperature. High pressure ratios also diminish the operating range of the compressor. They can also induce dirt retention in the inlet air filter and on the compressor blades. At worst cases, it can also result to compressor surge [13].

There are many other cycles and methods to improve the overall efficiency such as simple cycle with heat exchange, reheat cycle with heat exchange, intercooled compression and combined cycle [13]–[15]. As they are not essential with respect to this thesis, they are not reviewed.

2.3 Compressor

A compressor is an essential part of a gas turbine. The compressor pressurizes a working fluid, typically atmospheric air. Compressed air enables to increase the injected fuel in a combustor and furthermore increase produced mechanical energy in a turbine.

The compressors can be divided into three categories: the positive displacement compressors, centrifugal-flow compressors and axial-flow compressors. Separation between the categories can be done by flow and pressure the compressors are used for. The positive displacement compressors are used for high pressure and low flow. Centrifugal-flow compressors are used for medium flow and medium pressure. Axial-flow compressors are for high flow and low pressure [13].

The centrifugal-flow and axial-flow compressors are continuous flow compressors and are used for compressing the air in gas turbines. Their pressure ratio per stage varies between 1.05-1.3 (axial) and 1.2-1.9 (centrifugal) and efficiency varies between 80-91% (axial) and 75-87% (centrifugal). The compressor consumes 55-60% of all power generated by the gas turbine so the efficiency of the compressor is essential [13].

2.3.1 Centrifugal Compressors

In a centrifugal compressor, there is a stationary casing. The casing contains a rotating impeller. The impeller creates a high velocity to the air. The air is imparted to diverging passages where velocity decelerates and the pressure rises [15].

First, the air enters the impeller eye and gets accelerated by the vanes of impeller disc. The static pressure rises from the eye to the tip of the impeller because of centripetal acceleration. Rest of the pressure rise is obtained in the diffuser. In the normal design of centrifugal compressors half of the pressure rise takes place in the impeller and the other half in the diffuser [15].

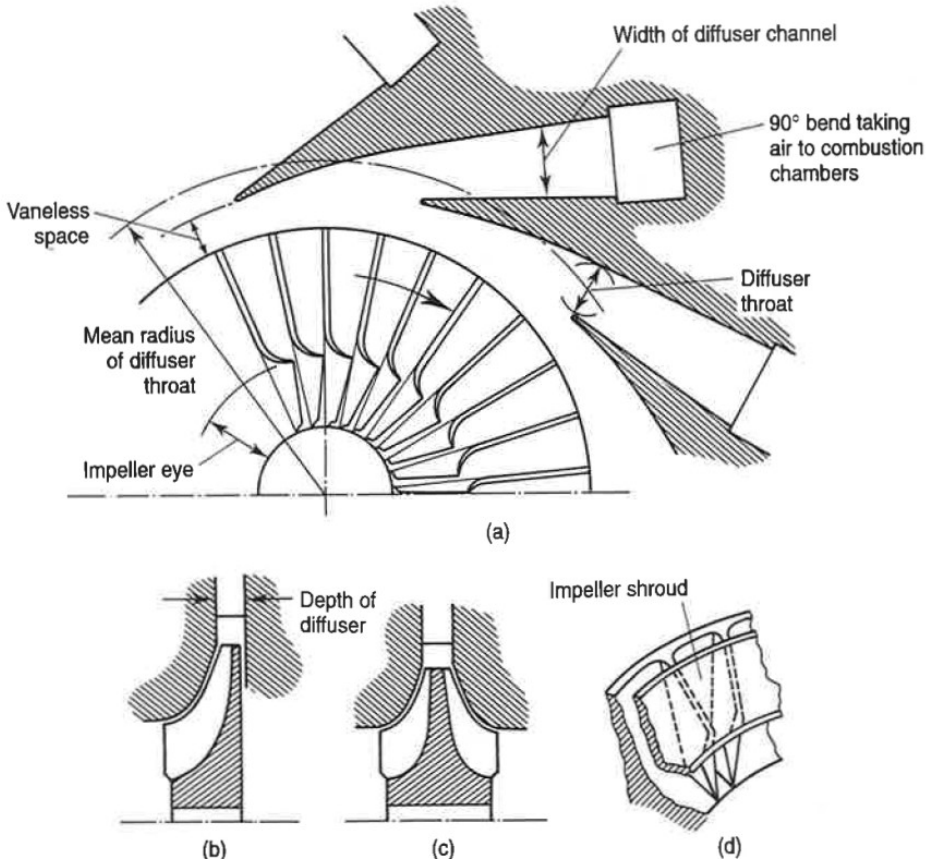

Figure 3. *Sketches of centrifugal compressor.* [15, p. 128]

The mean radius, number of vanes and vane angle affect the compressor characteristics. Air flow temperature, corrosion and stress on compressor must be considered when evaluating the compressor efficiency. Fouling of the compressor decreases the efficiency of the compressor.

2.3.2 Axial Compressors

In the axial compressors, the working fluid moves in axis direction. There are multiple stages, each having a row of rotor blades followed by a row of stator blades. The working fluid accelerates in the rotor blades and decelerates in the stator blades. In the stator blades the kinetic energy transfers into static pressure. Required pressure ratio is achieved by placing enough stages in the compressor [15].

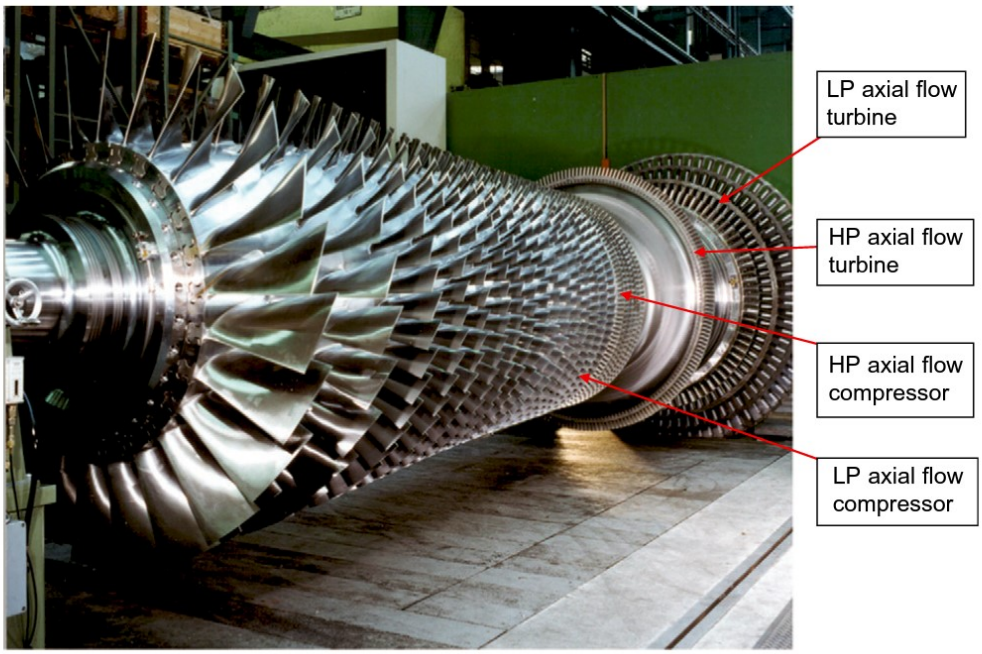

Figure 4. *A multi stage axial-flow turbine rotor. The compressor part can be seen in the forefront of the picture.* [13, p. 55]

The flow to the first stage of the axial-flow compressor is controlled by inlet guide vanes (IGVs). With the air flow angle, it is possible to adjust the air throughput and attack angle [15].

2.4 Combustion Systems

In the normal open–cycle process, combustion is a continuous process. The fuel is mixed with the air supplied by the compressor and combusted in the combustor. Ignition with an electric spark is needed only in the beginning of the combustion. It is important to maintain steady combustion and reliability in hot temperatures [15].

One of the basic combustor types is a can combustor which consists of individual combustion cans. The air stream from the compressor is split into separate streams. Each can has its own fuel supply from the common supply line.

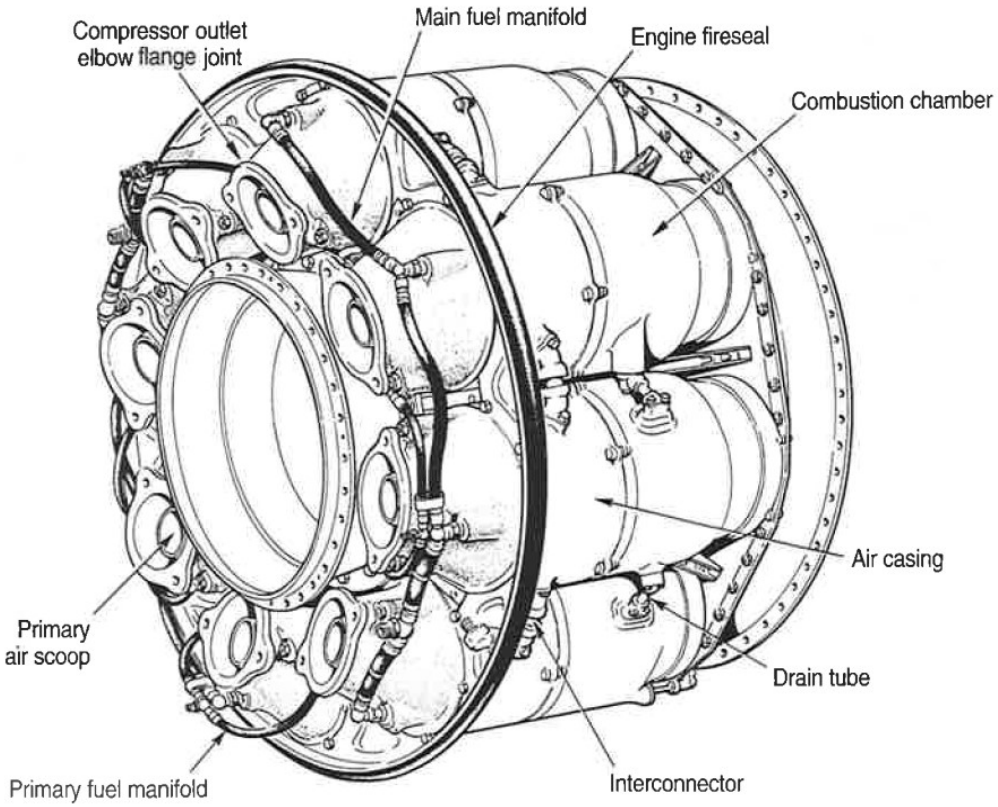

Figure 5. *Can type combustor.* [15, p. 236]

In industrial engines, can combustors are widely used. In Figure 5 the combustion cans are separate but there are other designs that use a cannular system, where individual flame tubes are spread evenly around annular casing [15].

In the combustion process, the air is supplied in three stages. In the primary zone, 15-20% of the air is mixed with the fuel to provide high temperature and fast combustion. In the secondary zone, 30% of the air is inserted to the process. Combustion is completed in the secondary zone. In the tertiary or dilution zone, the remaining air is mixed with products of combustion. Air cools the temperature of the product from the combustion for the turbine. Sufficient turbulence is needed to achieve constant temperature distribution [15].

In order to achieve a self-piloting flame in the air stream, recirculating flow pattern is needed. Some of the burning mixture in the primary zone is directed with swirl vanes back to the incoming fuel and air. Example of this arrangement is presented in Figure 6 [15].

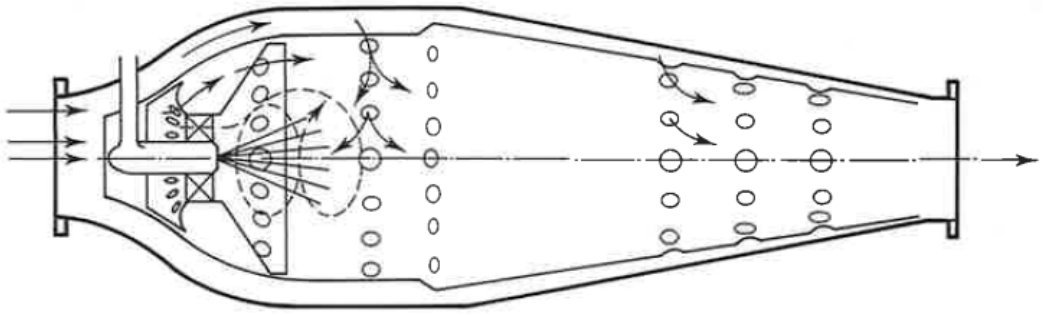

Figure 6. *Combustion chamber with swirl vanes.* [15, p. 240]

The most essential characteristics of combustion performance are: pressure loss, combustion efficiency, outlet temperature distribution, stability limits and combustion intensity [15]. Performance related subjects are looked closer at Section 2.7.

2.5 Turbine

As compressors, turbines come in many types. Two main types of turbines are radial-flow and axial-flow turbines. As this thesis studies axial-flow turbines, which is the most common type, this section is focusing on axial-flow turbines.

Radial-flow turbines are more efficient with low mass flows and they are used in cryogenic industries. Apart from the lowest powers, the axial-flow turbine is usually the more efficient solution [15].

Axial-flow turbine can be considered as a counterpart for axial-flow compressor. Figure 7 illustrates the difference between the turbine and compressor blades. The flow is entering and exiting from the turbine in axial direction. Axial turbines appear in two types: impulse and reaction turbines. In the impulse turbine, the enthalpy drop happens completely between the nozzles. Therefore, flow velocity is high when entering the motor. In the reaction turbine enthalpy drop is divided between the nozzle and the rotor [13].

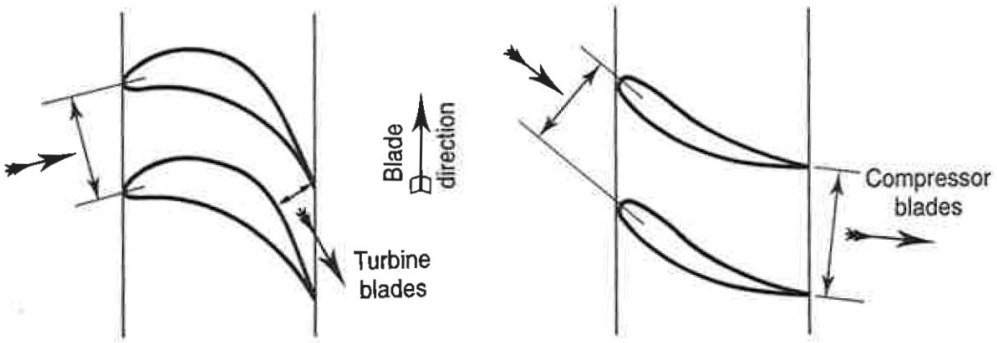

Figure 7. *The turbine and compressor rotor blades.* [15, p. 156]

Usually, there is more than one stage in the axial-flow turbine. Normally the front stages are impulse (zero reaction) whereas the later stages have about 50% reaction. There are differences in the outputs and efficiencies. Impulse stages produce about twice the output compared to the stages where 50% is reaction. As a drawback, the efficiency of impulse stage is inferior to the 50% reaction stage [13]. Characteristics of axial-flow turbine are shown in Figure 8. Because the enthalpy drop happens completely between the nozzles, the turbine is impulse turbine.

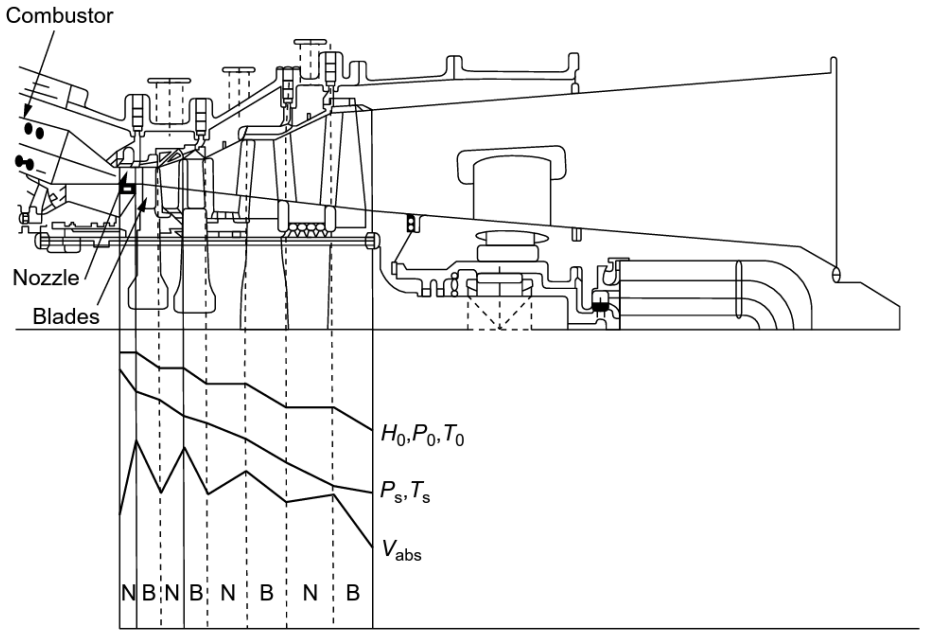

Figure 8. *Axial-flow turbine flow characteristics.* [13, p. 386]

As mentioned in Section 2.2, increased firing temperature improves the overall efficiency of the cycle. Because of high pressured air coming from the compressor combined with very hot combustion products, the first stages of turbine need air cooling. There are several concepts for air cooling: convection cooling, impingement cooling, film cooling and transpiration cooling. The most recent turbines in combined-cycle plants are cooled with steam [13]. Depending on the cooling technology, there are grooves, holes and other structures in the blade for cooling. The steam cooling is the most efficient technology and allows the highest firing temperatures.

In the turbine part, the mechanical condition of the turbine affects the most the performance. Fouling and erosion of turbine blades, foreign object damage to turbine or turbine nozzles corrosion can decrease the performance of the turbine.

Fouling of compressor blades is one important mechanism that decreases gas turbine performance over time. Particles, typically between 2 to 10 μm are adhering to turbine blades and are causing decrease of efficiency. Common particles are smoke, oil mists, carbon and sea salt [16].

2.6 Solar Taurus 60

The power plants analyzed in this thesis, have Solar Taurus 60 turbines. As all the plants analyzed are of the same turbine model the turbines can be benchmarked with each other: with the same inputs, the outputs of the turbines should be the same as well.

Solar Turbines Corporation owned by Caterpillar Inc has more than 40 years of experience from industrial gas turbines. Solar Taurus 60 produces 5670 kWe power and its heat rate is 11 430 kJ/kW-h [17].

The turbine is a single-shaft turbine with 12-stage axial compressor. The compressor's pressure ratio is 12.2:1 and inlet airflow is 21.3 kg/s. The turbine has an annular-type combustion chamber with 12 fuel injectors. The combustion chamber has a single torch ignitor system [17].

2.7 Turbine Key Performance Indicators

Measures that describe well the performance of a process or otherwise explain the quality or availability of the process are called key performance indicators (KPIs). KPIs guide the operator to make better process control decisions, and thus make the process more efficient and profitable. Some KPIs can help the operator to detect if there are some faults or abnormalities in the process.

One thing that limits the possibilities for KPI calculations is the number of available measurements. There can be several KPIs that could be beneficial for evaluating the performance of the system, but their values cannot be calculated from the existing measurements.

The first KPIs that are chosen for this study are the produced active power and temperature corrected power. Temperature of inlet air is affecting strongly the output of a turbine. At low temperatures, it is possible to get higher power out from the turbine whereas at higher temperatures the power decreases. With the inlet air temperature taken into account, the output power is normalized to respond the power that would be generated at 15 °C. Normalized output powers can then be benchmarked between different sites that operate at different ambient temperature. The power map of the Solar Taurus 60 turbine is presented in Figure 9.

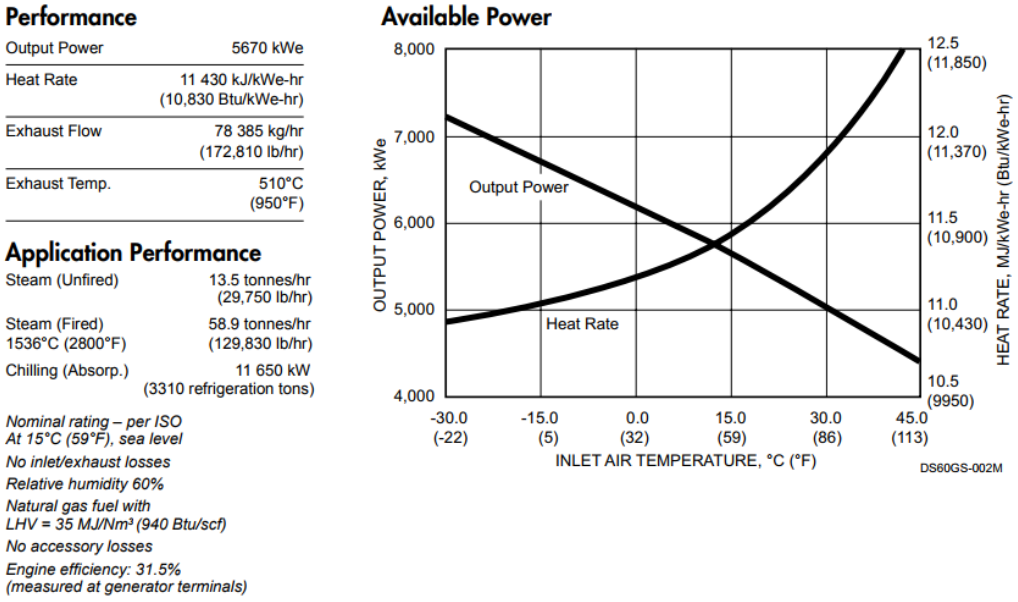

Figure 9. *The power map of Solar Taurus 60* [17].

In addition to inlet air temperature, the output power should be corrected with respect to the inlet air pressure and humidity. The pressure is affecting the output power because it affects air density. However, there are no measurements of inlet air pressure or humidity available at the sites studied, thus corrections based on them are difficult to implement. However, the output power correction due to the inlet temperature is the strongest.

Other potential KPIs are differences in the T5 temperatures. T5 temperatures are the turbine inlet temperature measurements and there are six of them. When the gas turbine is operating normally, the T5 temperatures should be quite even and constant in time. Wide spread of temperatures indicates that there may be problems in combustion or fuel supply streams can be blocked.

Turbine startup time could be one KPI. There is a sequence that an automation system follows during a startup. To use a turbine cost-effectively, the starts should succeed every time without problems. Delayed startup times indicate that there may be a need for maintenance. It is also possible to benchmark site operations with turbine startup. From that information, the operator can find if other turbine's actuators such as valves are well trimmed.

3. MACHINE LEARNING

Machine learning is an effective tool for analysing large datasets. Currently, the volume of available data is enormous and instead of getting data, efficient use of data is more important. Machine learning makes possible to find patterns and do predictions from data automatically.

Bell's book (2015) about machine learning provides definitions for ML (machine learning): Arthur Samuel defined ML as: "*[A] Field of study that gives computers the ability to learn without being explicitly programmed.*". Other definition for ML is provided by Tom M. Mitchell: "*A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.*". As a conclusion, ML is artificial intelligence where systems learn from data and improve with experience using computing [18].

There are two main types of machine learning. The first type is called supervised or predictive learning. From the inputs **x**, the purpose is to find mapping to outputs y from a labelled set of input-output pairs. Learning is done with a dataset called training set. The quality of the created model is validated with a test set [3].

The other type of machine learning is unsupervised learning. In unsupervised learning, there is only output data without predefined input data. Murphy (2012) describes that "*The goal is to discover "interesting structure" in the data; this is sometimes called knowledge discovery.*". Unsupervised learning is somewhat of what human beings and animals do: they learn from experience without right answers (e.g. learn to see without knowledge of what should be the right output) [3].

3.1 Machine Learning Process

Machine learning process starts with collecting data. Once data is collected, relevant attributes are defined. In this thesis, there are two plants with two turbines each so in total there is data from four turbines. To make the learning process and comparison between turbines "fair", the attributes are selected based on a machine that has the least measurements available.

After data is available and attributes are chosen, the objective for the machine learning algorithm is defined. A pseudo-code should be created to explain the objective and functionality of the application. Definitions for the machine learning algorithm are given in Section 3.2.

After defining the system, we can answer the questions: what we want to do with the data, how it is done and what are the results we are expecting? Based on the definitions, appropriate methods can be chosen to fulfil the requirements.

When the planning and definition phase is done, the actual work can start. The data is transferred to the development environment and coding the machine learning algorithm starts. If the planning phase is done well, the implementation phase is easier.

Finally, the developed ML application is tested with test data. Testing shows the accuracy and validity of the model. Based on the testing, the user can decide if the developed model is accurate enough or if it needs further development.

The test data can be collected separately from the training data. It is also possible to sample points from the training data to be used only for testing. When using that procedure, the size of training data diminishes. That is not a problem in a case where the amount of available data is large.

3.2 Specification of the ML Algorithm

The objective of the machine learning algorithm in this thesis is to predict temperature corrected power output of the turbine. The idea is to teach the model with training data where the turbine is performing well. If the turbine performance decreases because of fouling of compressor or corrosion of turbine blades for instance, the model prediction for power output should then be higher than the actual measurement.

The model uses several predictors in training. There are attributes such as inlet gas pressure, main gas valve position, air pressure after compressor, exhaust gas temperature, inlet air temperature and pilot gas valve position, 16 predictors total. The predictors are listed in Table 1.

Table 1. *Used predictors in the model.*

PREDICTOR
MAIN GAS VALVE POSITION
AIR PRESSURE AFTER THE COMPRESSOR
EXHAUST GAS TEMPERATURE
PILOT GAS VALVE POSITION
T5-1 TEMPERATURE
T5-2 TEMPERATURE
T5-3 TEMPERATURE
T5-4 TEMPERATURE
T5-5 TEMPERATURE
T5-6 TEMPERATURE
INLET AIR TEMPERATURE
REACTIVE POWER
NOX CONTROLLER POSITION
INLET GAS PRESSURE 1
INLET GAS PRESSURE 2
INLET GAS PRESSURE 3

It is assumed that the turbine performs well after overhaul. Based on the assumption, data from a relatively short time period (five days) after overhaul is chosen for training. Only one turbine is overhauled during the time on which there was data available.

The measurements from the system are available at one-second intervals. It would be possible to take for example ten seconds or one-minute averages. The data analyst must make a decision of used sampling period.

The power plants that are analysed in this thesis, have the same turbines (Solar Taurus 60). Because of that, generated prediction model should work on every turbine. In both plant locations, there is similar weather and climate conditions. That supports the expectation that the model works with every turbine.

The gas turbines are mainly operating with baseload. That means that the operator is all the time trying to maximize the gained power output from the turbine. Because of that, the model is trained to work only on the baseload.

The requirement that the model is trained to work only with the baseload means that there is a need for filtering of the data. If there are periods in the training data when the turbine is shut down, those periods must be filtered out from the data. In addition, the periods when the turbine is running with a partial load must be filtered out from the training data.

Additional filtering can be done for the data. There can be noise in the data and the machine learning algorithm may try to reproduce the noise and implement it in the model. That is an unwanted feature so filtering the data with for example moving average can improve the final result of the model.

3.3 Selected Methods

Testing the ML algorithms should be started from simple methods. If they do not work, the user can then move to more advanced methods. Advanced models tare slower to compute, they are heavier and in worst cases, they do not explain the phenomenon any better than the simpler methods.

Regression models are widely used statistical methods. Regression analysis utilizes relation between two or more quantitative variables. The response is predicted from one or more variables called predictors.

Linear regression model is expressed as:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{i,p-1} + \varepsilon_i \quad (3\text{-}1),$$

where Y_i is predicted value, β_0, $\beta_1, \cdots, \beta_{p-1}$ are parameters, $X_{i1}, \cdots, X_{i,p-1}$ are predictor variables and ε_i is Gaussian error term $\varepsilon \sim N(0, \sigma^2)$ at time $i = 1, \ldots, n$ [19]. The model is first-order model so there are no interaction effects between predictor variables.

When there is a large number of possible predictors, the number of possible models grows quickly. Some of the chosen predictors are describing the predicted value better than the others and some of the predictors may not be describing the wanted value at all. Therefore, those non-describing predictors should be dropped out from the model. There are automatic search procedures developed to simplify the selection of the model variables that describe the system.

In this thesis, a method called stepwise regression is used. This automatic search method either drops or adds a predictorvariable to the model. The method uses error sum of squares reduction, coefficient of partial correlation, t* statistic, or F* statistic as the criteria for adding or removing a predictor variable. The t* statistic can be stated as

$$t^* = \frac{b_l}{s\{b_l\}} \quad (3\text{-}2),$$

where

$$b_l = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sum (X_i - \bar{X})^2} \quad (3\text{-}3)$$

and

$$s\{b_l\} = \sqrt{\frac{MSE}{\sum (X_i - \bar{X})^2}} \quad (3\text{-}4),$$

where MSE is mean squared error.

The stepwise regression method chooses only one regression model as the “best”. “Best” subsets algorithm selects multiple “good” models for final consideration and the user can choose which model suits best for the use. In that sense, the stepwise regression has its own vulnerabilities. The quality of the model must therefore evaluate using different diagnostics [19].

There are different approaches to stepwise regression. One is forward stepwise selection where there are no variables in the starting situation. The algorithm uses chosen model fit criterion and adds (or later deletes) the variables based on which variable gives the best fit for the model. Backward elimination method can be considered as the opposite method. It starts with all the candidate variables and drops the variables based on a similar criterion as in forward stepwise selection. There is also a method called bidirectional elimination that is a combination of the forward stepwise selection and the backward elimination.

The forward selection method is described more accurately below [19]:

1. The method fits a simple linear regression model (only one variable in the model) with each of the possible variables. Each simple linear regression model is tested with t* statistic. The variable that has the largest t* value is the candidate for first addition. There is a predetermined value that the t* value must exceed.
2. The regression routine makes all regression models with two variables, where the variable that was chosen in step 1 is the other one. The variable with the largest t* value or respectively the smallest P-value is the candidate for next addition. Again, the t* value must exceed the predefined level or the program terminates.
3. In the third step, the stepwise regression model examines if one of the variables that are already in the model should be dropped. Again, the t* statistic is done for every variable in the model. The variable with the smallest t* value is a candidate for dropping. If the value is below the predefined level, the variable is dropped from the model. Otherwise, the variable is kept in the model.
4. The stepwise regression routine continues the examination and examines if new variables should be added. Then it examines if any of the existing variables should

be dropped. The routine stops when there are no new variables that could be added or dropped or the number of predefined steps of the routine is exceeded.

As earlier mentioned, the quality of the model should be tested. There are many different methods that can be used. One method that describes the coverage of the model is coefficient of determination. That measure is denoted as R^2 and given by [19]:

$$R^2 = \frac{SSR}{SSTO} = 1 - \frac{SSE}{SSTO} \tag{3-5},$$

where *SSR* is regression sum of squares, *SSTO* is the total sum of squares and *SSE* is error sum of squares. The higher the R^2 is, the better the model is. It is always true that $0 \le SSE \le SSTO$, hence

$$0 \le R^2 \le 1 \tag{3-6}$$

Even if the R^2 is high, the model might not work properly if most of the predictions require extrapolation outside the area where the observations are done in the test data [19]. If this statement is exemplified in this thesis, the training data the turbine is operating only on specified load range. There are no guarantees that the model would work as well outside of the training data's operating range.

The model can be further analyzed by calculating the prediction uncertainty. For given values of $X_1,\ldots,X_{p-1}$, denoted by $X_{h1},\ldots,X_{h,p-1}$, the mean response is denoted by $E\{Y_h\}$. Vector $\mathbf{X}_h$ is defined as [19]:

$$\underset{p\times 1}{\mathbf{X}_h} = \begin{bmatrix} 1 \\ X_{h1} \\ \vdots \\ X_{h,p-1} \end{bmatrix} \tag{3-7}$$

so that the mean response to be estimated is:

$$E\{Y_h\} = \mathbf{X}'_h \beta \tag{3-8}$$

The estimated mean response corresponding to $\mathbf{X}_h$, denoted by $\hat{Y}_h$, is

$$\hat{Y}_h = \mathbf{X}'_h \mathbf{b} \tag{3-9}$$

The variance for mean response (residual standard error) is

$$\sigma^2\{\hat{Y}_h\} = \hat{\sigma}^2 \mathbf{X}'_h (\mathbf{X}'\mathbf{X})^{-1} \mathbf{X}_h \tag{3-10}$$

The $1-\alpha$ confidence limits for $E\{Y_h\}$ are [19]:

$$\hat{Y}_h \pm t(1-\alpha/2;n-p)s\{\hat{Y}_h\} \quad (3\text{-}11)$$

With the confidence limits, it is possible to not only to inspect how close the prediction is to actual value but also how well the actual value stays inside the confidence interval. Quite often 95% confidence interval is used in the examination of the model.

3.4 Software

In this thesis, R was used for developing the model. R is a statistical computing software/programming language that is widely used by engineers and scientists. It suits well for large-scale computing, statistics and machine learning for instance.

The data that was used to train the model, is read with R. The data is then filtered and after filtering, the data is ready to use for training. The training is done with stepwise regression function. Libraries for many statistical functions for R are available in Internet. In total, there are more than 12 000 packages available in CRAN (Comprehensive R Archive Network) [20].

The selected stepwise regression learner package works in either forward, backward or both directions (bidirectional elimination). The user can define the model that the algorithm starts with. In addition, the user defines the simplest model that the algorithm should produce and the most complex model (maximum number of variables) that the algorithm should produce. The user can define some other modelling parameters as well, such as a maximum number of steps [21].

For end use the predictive models can be implemented with Python, which allows applications in more versatile environments, such as cloud services. The parameters of the model that is created with R, are saved in the configuration file. The Python code reads the parameters from the configuration file and uses them to make predictions.

4. VALMET INDUSTRIAL INTERNET (VII)

Internet of Things (IoT) is a relatively new concept. Its basic idea is that all the devices are connected to the Internet either wireless or wired. Applications can be developed for the devices and value created for consumers. Also in industry, the companies are pursuing their own Industrial Internet of Things products and services. This thesis is focusing on Valmet Industrial Internet, but there are many other implementations also: for example ABB [22], Andritz [23], GE [24], Honeywell [25] have their own implementations of the IoT platforms in power production.

Industrial Internet has many names: Industrial Internet of Things (IIoT), Internet 4.0 and Industry 4.0. However, the basic idea remains the same: instead of consumer markets, the concept of Internet of Things is used in an industrial environment. IIoT can be applied in energy production, manufacturing, transportation, logistics and many other businesses [26]. In this thesis, the concept is called IIoT.

In this chapter, the basic concept of Industrial Internet of Things is introduced first. Valmet's own Industrial Internet solution is introduced after the general overview. At the end of the chapter, security issues regarding IIoT are discussed.

4.1 Industrial Internet of Things in General

The main idea of Internet of Things is that all the devices are connected to Internet and therefore interacting with each other to reach common goals [27]. In the industrial world, the devices can be sensors, actuators, control systems and RFID (Radio-Frequency Identification) tags for instance.

IIoT combines various technologies. For example, Big Data, cloud computing, networking and artificial intelligence are used [28]. The strength of the IIoT is that various communication technologies are connected, such as WSN (Wireless Sensor Network), RFID, WLAN (Wireless Local Area Network), M2M (machine-to-machine) and traditional IP (Internet Protocol) technologies [29].

IIoT relies on the structure of M2M technology. Especially in factory automation, M2M communication has been in use for a long time; the machines are communicating with each other and co-operating to manufacture products. IIoT architecture is slightly different for there is an Internet layer between things and services [26]. The devices transmit data to central server or cloud., where data is integrated and analyzed. Therefore, the user's computer needs less computing capacity.

Typical Industrial Internet architecture can be explained with three-tier topology illustrated in Figure 10. Three-tier topology has the main components that Industrial Internet Systems (IIS) network needs [26].

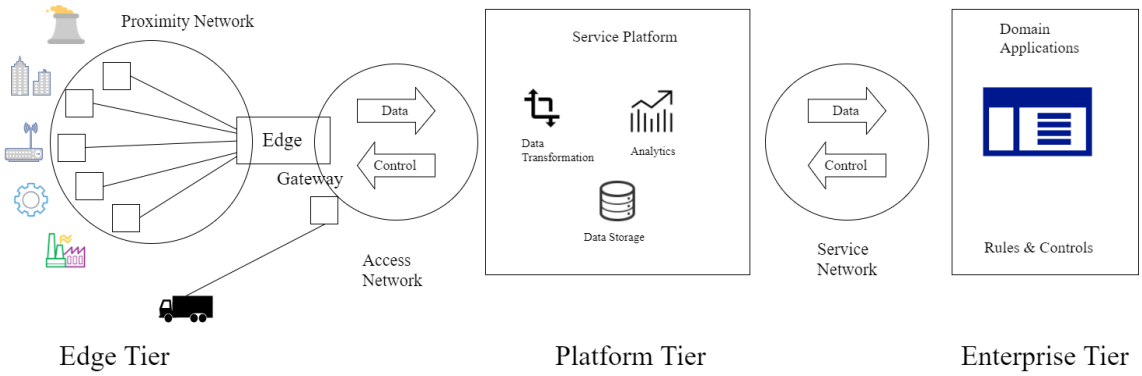

Figure 10. *Three-tier Architecture. Adapted from* [26, p. 77].

Data is collected and transmitted over the proximity network in edge tier. Data comprises all measurements such as power plant process data, weather data services, production management data and logistical data just to mention few. In Figure 10, there are control signals coming from platform tier to edge tier. Often this connection is not used especially in power plants or other critical targets where external control signals could be a security risk. The control is done locally on-site.

In the middle, there is platform tier. It receives data from the edge tier using the access network. The platform tier is responsible for transferring and processing the data. Data flow management and data storage are also in the platform tier. The calculations and analytics are executed in the platform tier as well.

The last tier is the enterprise tier. Applications, business intelligence tools and end-user interfaces are there. Normally the customers see only things that are in the enterprise tier. Rights and access to different applications and data are controlled with access management systems. Access control and security issues are discussed more in Section 4.8.

A variety of IIoT use cases of IIoT have already been implemented. Healthcare service industry, food supply chain, mining industry, and transportation and logistics have examples of applications of IIoT. Artificial intelligence and cloud computing are employed more and more in IIoT and those research trends are likely to grow in the future [29].

4.2 Valmet Industrial Internet in General

Valmet has launched its Industrial internet offering where it combines company's long-term expertise of process automation and control systems to a modern Industrial Internet

ecosystem. The VII platform serves energy, pulp, paper and board and tissue customers [30].

Valmet Industrial Internet applications provide users data visualization, reporting and guidance, asset reliability optimization and operations performance optimization. For the customers, the applications are accessible from Valmet Customer Portal. The concept includes also Valmet Performance Center (VPC) where experts help customers to optimize their processes. In VPC the customer can do continuous remote monitoring, controls and fine tuning optimization, on-demand expert remote support and data discovery and big data analysis [31].

4.3 Valmet Industrial Internet System Architecture

Valmet Industrial Internet can be divided into sections based on the architecture illustrated in Figure 11. The platform is mainly using the Amazon Web Services (AWS) cloud components: Platform as a Service (PaaS), Infrastructure as a Service (IaaS) and Software as a service (SaaS). PaaS means that the whole platform is provided for the customer. IaaS means that there is a set of platforms where the customer can create his/her own applications and services. SaaS means that the service provider provides software to the customer. The technologies and services are described more accurately later in Sections 4.4 – 4.8.

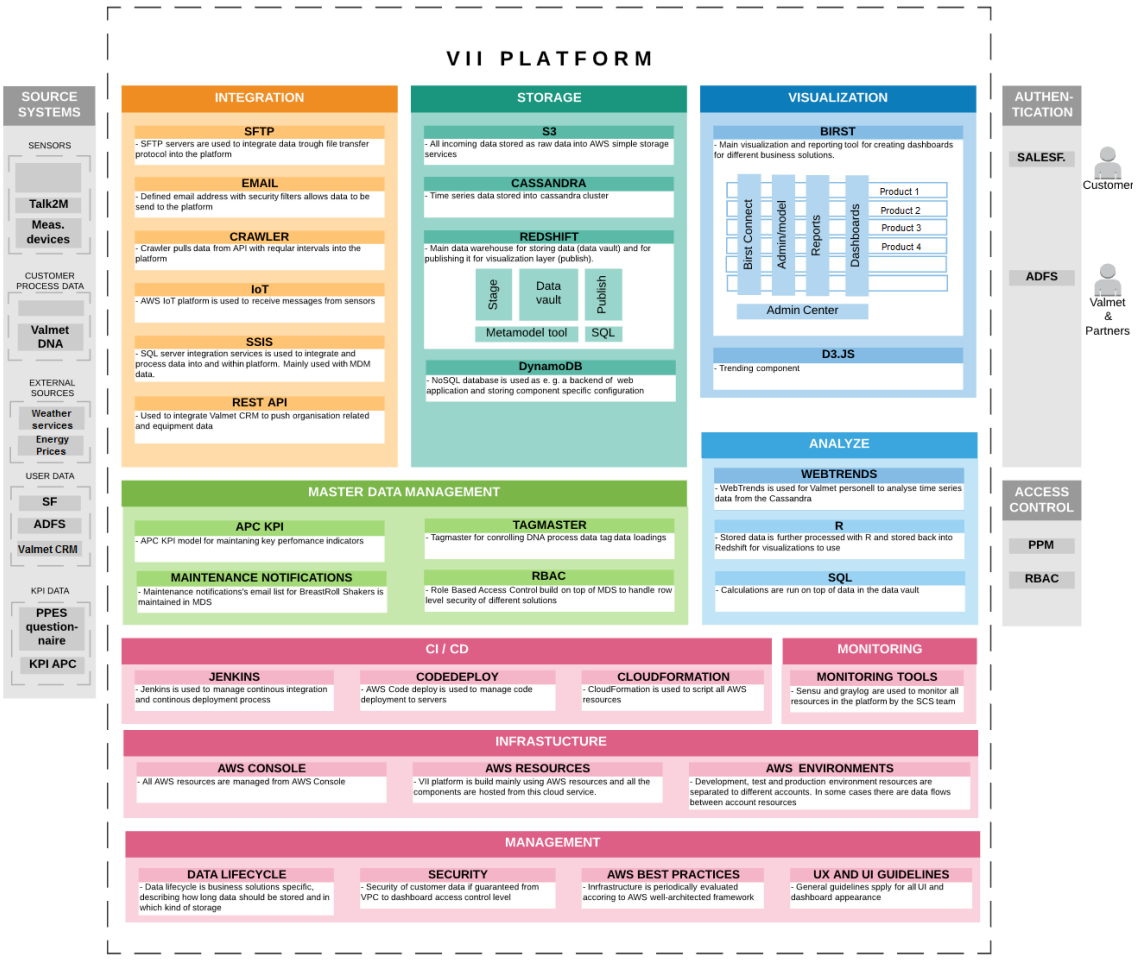

Figure 11. *Valmet Industrial Internet platform architecture (Courtesy of Valmet Inc).*

In Figure 11, the source systems are on the left-hand side. The source systems provide all data that is coming to VII platform. Data covers all raw measurement data, metadata, external sources such as weather information, user data, forms and KPI data for example. Available data is integrated to the platform via technologies such as SFTP (SSH File Transfer Protocol). Normally system level integrations (e.g. Valmet DNA) are done with SFTP and when connecting single devices to the system, AWS IoT interface is used.

There are different data storages in VII platform depending on a use case. S3 (AWS Simple Storage Service), Cassandra, RedShift and DynamoDB are the storage solutions of AWS [32]. For example, the raw data is normally stored into S3.

The data is visualized with business intelligence tools, mainly with Birst software. Customers see only the visualization part of the platform. To get access to visualizations, users need to authenticate themselves via ADFS (Active Directory Federation Services) (internal users) or SalesForce.com (external users).

User access is controlled with master data services tool and on the top of that, data visibility is controlled with Role Based Access Control (RBAC). Analytics can be done by

using SQL directly on top of the storage system (RedShift) or by loading the data into analysis tools such as R.

Continuous integration and continuous deployment (CI/CD) are done mainly by Jenkins. CodeDeploy is used, too. Jenkins is hosted on EC2 server on its own AWS account. The source code is hosted in Valmet Bitbucket.

In the VII platform, there are three different environments: developer, test and production. The developer profile is used for developing new applications to the platform. After the development, the application is tested with a test profile. When the product has passed testing, it can be taken to production. The ready products are deployed to production environment. The customers access to the products in that environment.

4.4 Data Integration

In VII, there are several technologies to integrate the data from source systems to VII platform as can be seen from Figure 11. In this section, one example of data integration is described. Integration from Valmet DNA to VII platform is chosen as an example because it will be probably the most common data source for the platform. Valmet DNA is Valmet's own automation system.

From Valmet DNA, the data is sent to the SFTP servers in packages that include data and metadata that describes the content of the file. File name is chosen so that it describes the data: the name can include the mill, line, source system and timestamp.

Between the SFTP server and the mill, there is AWS Route 53 service. Route 53 is routing the data traffic from source systems to SFTP servers. It accepts only the data from trusted sources. With Route 53, it is also possible to distribute the load for SFTP servers evenly [33].

SFTP is a secure file transfer protocol that runs over the SSH protocol. SSH (Secure Shell) enables computer to remote login to another computer securely. SFTP is using data encryption and cryptographic hash functions. The server and user are authenticated so the transfer is secured [34], [35].

SFTP servers have common Elastic File System (EFS) that is Amazon's file storage that is used with Amazon EC2 instances. The files are stored there from all SFTP servers. Another virtual machine is running a script that polls for new files that are sent from SFTP servers to EFS. The virtual machine copies the files to AWS S3 bucket and the original files on EFS are moved to an archive folder.

4.5 Data Vault Modeling

To better understand the data and its structure, the concept of data vault (DV) modeling is introduced. DV modeling is a warehouse architecture that is used in this project in RedShift. Data vault suits well for the data warehouses that integrate data from multiple operational systems. It also makes possible to trace data and find the origin for it.

Data Vault 2.0 is a system that includes best practices of modeling, methodology, architecture and implementation. Modeling provides patterns to integrate raw data from different sources together. Methodology means the use of best practices. The most usual practice is that a development team is focusing on sprints (two–three weeks) where it tries to optimize the repeatable data warehousing tasks. Architecture defines the structure of the system and how data systems are integrated together. Implementation means the actual automated data flow and error reduction [36].

The tables that are used are normalized. The database then consists of unified tables where the data is traceable [36]. The data vault model consists of hubs, links and satellites. In the hub, the unique business keys are listed. The business keys are drivers of the business. With them it is possible to tie the data to different business processes. The relations between those business keys are listed in links. The actual measurement data is in satellites. Data vault conceptual model is in Figure 12.

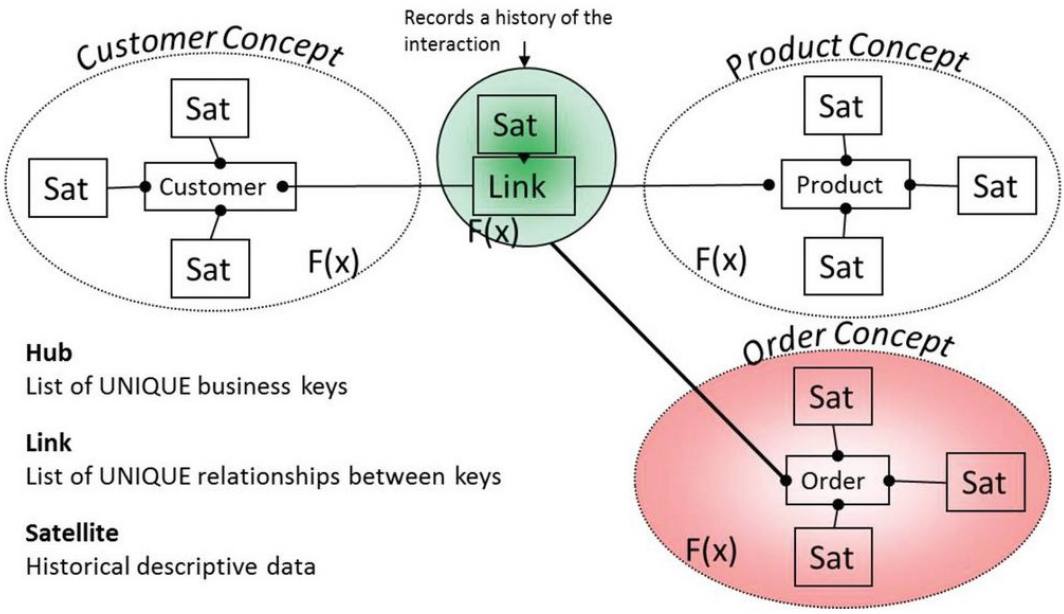

Figure 12. *Data Vault Conceptual Model* [36, p. 139].

In Section 4.1, three-tier architecture was introduced. Also, DV 2.0 architecture is based on three-tier data warehouse architecture. In the data warehouse, the tiers are divided to

landing zone, data warehouse and information delivery layer. Architecture is illustrated in Figure 13.

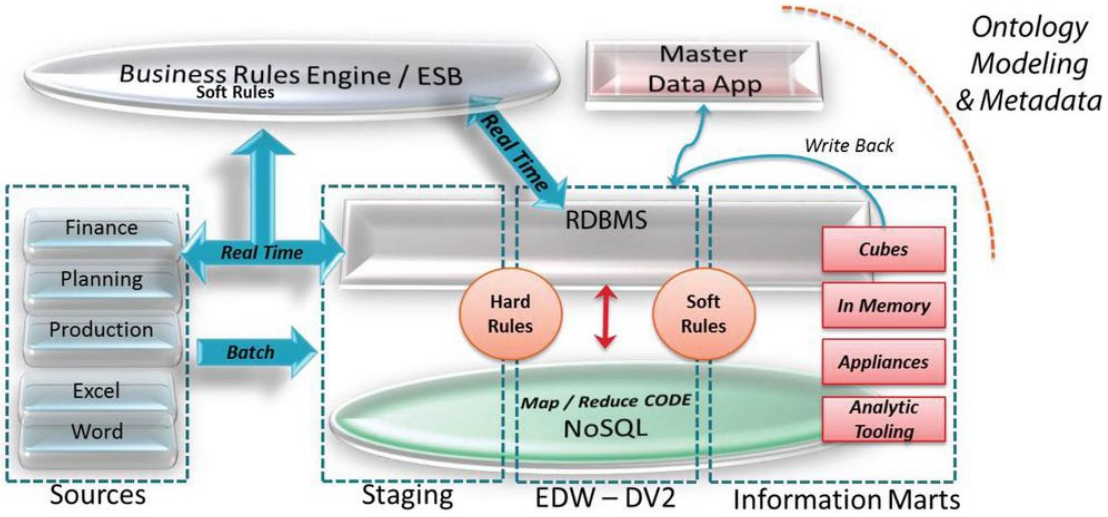

Figure 13. *Data Vault Architecture* [36, p. 149].

The data arrives at in first tier (staging) as real-time data or as batches. Staging is a temporary storage. From there, the data is transferred to correct data warehouse based on hard rules. The second tier is enterprise data warehouse (EDW). The data is structured based on a data vault model.

The last tier is information marts. There the EDW data is packaged to snapshots of data (for example hourly or daily data), and it is available for visualization or calculations. The data that is available in the third tier is decided by soft rules.

Outside of these three tiers, there is also a master data application that can handle for example missing data. The master data application is a set of manually managed data, such as metadata or customer information that is not often changeable. There can be also defaults that are used for filling the blank values in measurements.

4.6 Data Storage

All incoming data is stored to Amazon S3. The data is stored as objects in buckets. The buckets can contain prefixes to organize the data. S3 a is simple and cheap storage for storing large amounts of data. Thus it suits well for storing raw data from the mills and sites [37].

The stored data must be in right form and the files must be named correctly. Usually, the measurement data from the sites is sent to VII as .csv (comma separated values) files where each row has timestamp, tag name and value. The file naming includes mill id,

process area, line id, location, data type and timestamp. Example of the file naming could be "*123456_energy_001_rusko_data032_20171109000000.csv*".

From the data, the database tables are created with DynamoDB. DynamoDB is a NoSQL database service. In DynamoDB, the user adds a table that includes information about which files it reads (based on a file naming) and to which table it saves the files. It is also possible to define the timespan of data in database table. With timespan, it is possible to reduce the storage use and make the database more efficient [38]. In VII platform, the DynamoDB points that to which table the incoming data is directed.

From S3, the data is transferred to RedShift or Cassandra. Two lambda functions either store data to RedShift loader bucket from where the data is transferred into RedShift staging area or load the data into Cassandra. AWS Lambda is a compute service that runs a code without user's need for a server. It executes the code only when needed (new data comes in) [39]. Lambda function creates a message and sends it to SQS (Simple Queue Service) queue.

The lambda is attached to SNS (Simple Notification Service). Users, devices and applications can send and receive notifications from the cloud with SNS. The client can either publish or subscribe the messages [40]. S3 publishes when new data is available. Lambda subscribes the message and pushes it forward. The messages are in JSON-format.

In VII system, RedShift is the main data warehouse where the data is stored. For RedShift, the data vault model configuration is created with a metamodel tool. It is a tool where the user can define hubs, links and satellites. Keys and columns are also defined there. Based on configuration the tool creates DDLs (Data Definition Language). With them, the user can create tables in RedShift.

From SQS, the ETL (extract, transform, load) process reads messages and forwards the data based on the information that the object has. After the ETL process has dumped the JSON objects, it retrieves a workflow file from S3. The workflow file has a workflow sequence list that indicates in what order and where (data link, hub or satellite of data vault model) the data is pushed from staging area table.

In RedShift, the data is furthermore pushed to publish layer. For example, Birst that is used for visualizing the data is using RedShift's publish layer as a source for its dashboards.

The other database that is used is Cassandra by Apache. It is highly scalable and available, distributed database. The data is replicated to multiple nodes so the data is fault tolerant [41]. Cassandra is not an AWS resource but it is running on AWS EC2 cluster. Cassandra is used for storing time series data on a Cassandra cluster.

4.7 Data Processing, Analysis and Visualization

The visualization of the data for customers in this thesis is done with Birst. The data analysis and calculations are done mainly with SQL or R-software. In addition, other software products can be connected to VII platform. These parts are described in this section as independent components of the system.

Industrial Internet Consortium (IIC) defines analysis done in IIoT systems as Industrial Analytics. The use of IIoT enables users to apply also other data sources than business data in analysis, such as weather forecast data. The business leaders have recognized the importance of Industrial Analytics. A survey made by IIC shows the key areas that business leaders see the best opportunities [42].

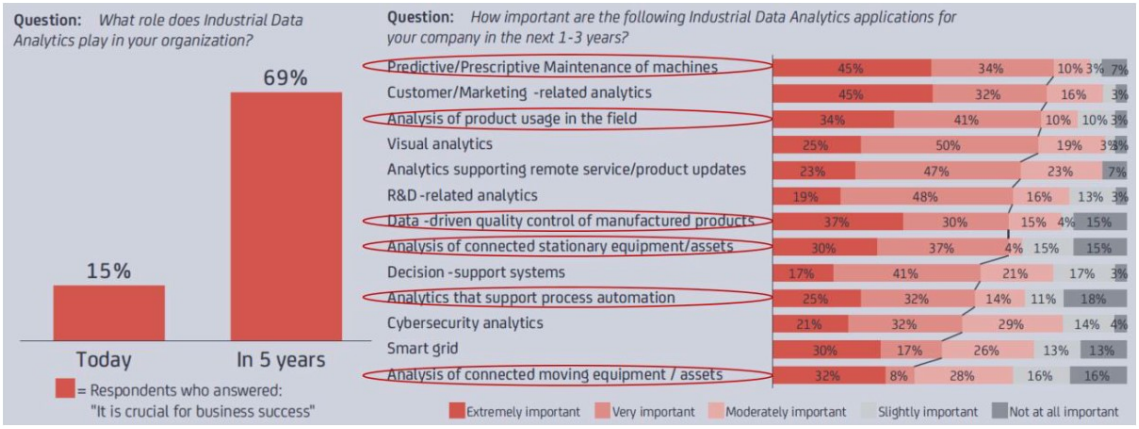

Figure 14. *Survey for business leaders about the importance of Industrial Analytics* [42, p. 13].

Figure 14 shows that the business leaders understand the potential of industrial analytics and the need for it in the future. Predictive maintenance of machines has been considered the most important goal of analysis. In industry, there are good possibilities for developing predictive maintenance applications. Instead of doing maintenance work based on schedules, it is more efficient to do it based on the analysis of the condition of machine parts. Then the maintenance would be done at the time it is really needed.

The challenge regarding to analysis is that seldom the relevant variables can be measured directly. For example, when planning the predictive maintenance for gas turbine, there are no measurements for fouling of compressor or corrosion of turbine blades. However, it is believed that the fouling of compressor and the corrosion of turbine blades can be deduced from other measurements indirectly with models and calculations.

Analytics can be divided into three main categories: descriptive, predictive and prescriptive analysis. The descriptive analysis makes analysis from history or current data and it

can be used for example anomaly detection or model building and training. In the predictive analysis, outcomes or behaviors are predicted with statistical or machine learning techniques. With the prescriptive analysis, it is possible to do guidance or recommendations based on results from the predictive analysis [42].

4.7.1 Birst

Birst is a business intelligence (BI) tool that enables the users to visualize data easily. The company was founded in 2004 [43]. The users can create charts, histograms, trends or other illustrations and combine them into views, called dashboards. A part of the dashboard, e.g. trend is called a dashlet.

In Birst, the user applications are called spaces. For a space, the user can define access rights (viewer, admin), and the data sources. Birst is connected directly to RedShift with Birst Connect. The sources can consist of multiple tables. Furthermore, local sources are possible to add into Birst.

The Birst user interface, Figure 15, is simple. A dashlet is created with drag-and-drop. The attribute of interest can be selected and dropped to the “canvas”. With menus the user can define colors and styles for the dashlet.

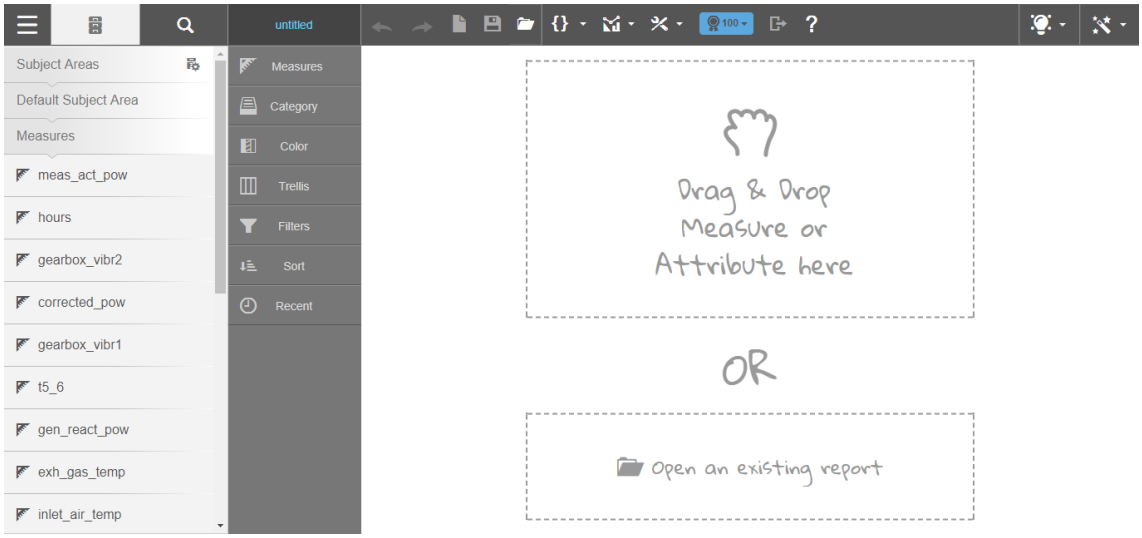

Figure 15. *User interface of Birst Visualizer.*

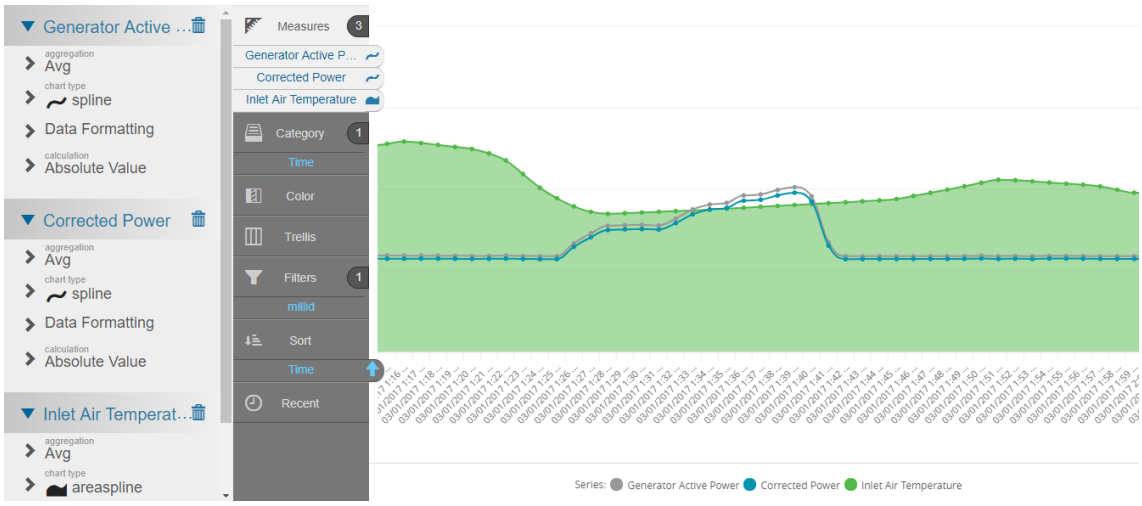

Figure 16. *A ready dashlet.*

An example of a dashlet is shown in Figure 16. There are two objects that are splines and one object that is areaspline type. Dashlets can then combined to a dashboard. One space can consist of multiple dashboards. The dashboards are created to illustrate entities logical from domain knowledge perspective. There can be a collection of KPI values as one dashboard and then there could be one dashboard for emissions, for example.

A dashboard may have filters. The user can filter data by dates, mill or line, for example. Scrolling of data is therefore easy and benchmarking different lines of one mill is made straightforward. Values of trends or other visual components can be seen by hovering the cursor on the trend.

The dashboards can be stacked. The main view can show the values as a histogram in one-hour averages. It is possible to create functionalities where by clicking one hour-wise value in the dashboard, a minute-wise view opens. This is a convenient functionality when the user wants to scroll data from wide time span and then zoom to a shorter period.

The dashboards should be so universal that the same dashboards can be used on many sites. If a customer has many sites, there can be a landing page with a map and all customer's sites are marked on the map. From the landing page, the customer can select the site of interest and view the data of that plant.

4.7.2 SQL Calculations and Aginity Workbench

The data can be read from the Redshift where it is stored according to the data vault model. With SQL, it is possible to read values from tables and use them in the calculations. Furthermore, the user can save the calculated values to a new table.

Aginity has created an application called Aginity Workbench that provides GUI-based (Graphical User Interface) tools that make the development easier. The application is SQL

database development tool where it is possible to do queries, create and manage tables and schemas [44].

As mentioned earlier in Chapter 4.6, raw data coming from S3 is pushed to the staging area. Based on the data vault model, the data is moved to data vault layer. When the calculations are added to the existing data, the data can be moved to publish layer. All the data that is presented in Birst is usually read from publish layer. The data is stored in the tables that are called views. For example, one view-table could be *turbine_kpi_view*.

4.7.3 R

R is an open source software that is made for statistical computing. It runs on multiple platforms. The software is an open source project so it has a wide community that provides a large number of different libraries that can be used in the analysis [45].

R runs on its own server and it is set up with cloud formation stack that can be found from the Valmet Bitbucket. The R server connects to RedShift with JDBC (Java Database Connectivity). It is an application programming interface (API) that defines how the client is connected to a database.

When the connection is opened with JDBC, it is possible to query data from the RedShift with SQL queries. Data is stored to local variables or tables in R and after that, the computing is done in R. In R, the user can create for-loops, and/or structures, models, statistical tests and other analysis. R can be scheduled to return values for example once a day.

4.7.4 AWS Lambda

AWS Lambda is a serverless computing service. Lambda is a function that does computing only when it is triggered by an event. The trigger can be a change in the AWS other resources such as DynamoDB or it can be scheduled with AWS CloudWatch service.

AWS Lambda is scalable and the user pays only for the compute time that is used. This makes the use of AWS Lambda effective especially in the use cases where the need for computing is not continuous or the computing is done only for example once a day. AWS Lambda supports various languages (Node.js, Java, C# and Python) [39].

4.7.5 Other tools

It is possible to access VII resources with other tools. For example, Matlab can be connected to VII. There is JDBC driver for Matlab available and by adding the driver to Matlab, the connection can be created to RedShift.

Amazon websites [46] provide information of JDBC drivers that are needed to connect to Amazon RedShift. That means that there is almost unlimited number of possible software products and tools that can be connected to VII platform.

Software licenses must be considered when adding new software products to the Industrial Internet environment. The licenses can be very expensive. Often the problem can be solved by using license-free implementations such as Python.

4.8 Data and Application Access

The data access and permissions are well controlled. The user needs own permissions both to the environment (AWS) and to the application (Birst) separately. The accounts cannot be shared with anyone but they are personal. This section describes the structure of access rights and permissions in the environment and application level.

In AWS, there are three environments: development, test and production environment. The names of the environments are quite descriptive. The actual development is done in development environment. Next, the piece of work is tested in test environment and a finalized product is then moved to production environment.

In each environment, a user can have different level of rights. The user with admin role has full permissions everywhere. The developer role restricts some of the rights but the developer can develop applications by creating new things in the environment. The most restricted role is viewer. With the viewer role, the user cannot modify the AWS components, but only view objects.

In Birst there is a three-level structure for permissions and data access. The first level is space access and group mappings. This level gives the user Birst access to the spaces. Internal users are using Active Directory Federation Services (ADFS) and the customers are logging into the spaces via SalesForce.com service.

The second level of Birst permissions is capability and content security. The capabilities and content are secured with user groups. The separation between the groups is lightly different from the one in the AWS environment. The internal users can be either in developers group or viewers and reviewers group. The developers have all the access rights to Birst. With viewers' and reviewers' access rights the user can explore created dashboards and investigate dashboard visualizer. For the customers, there is SalesForce.com viewers group. It has the same rights as internal viewers group but the customer cannot explore dashboard visualizer.

The last level of Birst permissions and data access is row level security. It handles the data security. Accesses are based on the user accounts (email addresses). Solution owners

can add row level rights through Master Data Services (MDS). The rights can be determined to each customer separately so they can see data only from their own mills. By default, users will not see any data.

This three-level structure serves well the purpose of Birst. Customers from the same business area can have access to same spaces but the content in the space, i.e. what data is available, is defined in the third level. The same dashboards can be then used with many customers having different data access rights.

4.9 Security

According to Bain IoT customer survey in 2016, the customers are most concerned about security issues in IoT [47]. After that, there are concerns about high prices or unclear economic benefits, concerns about IT, technology challenges and many others [47]. Because the security issues are the biggest concern, they must be considered in the IIoT systems. The data access and permissions take care of the visibility of system so that right people have access to right things. However, there are also many other threats to the system.

In IIoT systems, there are several attack surfaces. Systems are smart factories that consist of several cyberphysical production systems (CPPS). There are electronics, software, networking, machines and humans. In those surfaces, different threats are endangering the system. Some of the possible threats are listed in Table 2.

Table 2. *Attack surfaces of cyberphysical production system (CPPS)* [48].

ATTACK SURFACE	EXAMPLE TARGET	THREAT
ELECTRONICS	CPUs, microcontrollers, actuators, sensors	Invasive hardware attacks, side-channel attacks, reverse-engineering attacks
SOFTWARE	OS, applications	Trojans, viruses, runtime attacks
NETWORKING	Ethernet, WiFi	Man-in-the-middle, denial of service, eaves dropping
HUMANS	Engineers, operators	Phishing, social engineering

One important goal is to secure the availability of the data. That prevents the delays that could result in loss of productivity. Another vital objective is to prevent system failures. To achieve that, protection against sabotage, malware and networking attacks must be secured [48].

Another concern regarding data is ownership issues. The customer and the service provider must agree on the ownership of the data. There can be multiple questions about the data ownership. For example, the customer can have the ownership of the raw data. When the service provider does analysis and calculations with raw data, who owns the results of calculations and analysis? There must be also strict rules regarding access of the third parties.

There are no general rules of the ownership of the information. There are regulations and laws about data but they are not universal. Some of the regulations are valid only in the U.S. or in Europe [49]. There are also neutral organizations that customers can trust. One example is Open Trust Alliance that gives guidelines on multiple security issues such as device security and privacy disclosure [50].

In addition to the laws and regulations, also the agreements between service providers and customers are needed. The regulations and guidelines can be used to help to create the agreements. As a result, all parties should have a common understanding about ownership of the data, privacy and responsibilities concerning to security.

5. IMPLEMENTATION

The implementation phase consists of several stages. First, the data is imported from the info server to one-day packages. The data is then transferred to the cloud. From the data, the training data for machine learning algorithm is selected. The data is then filtered and analyzed. The machine learning model is created. The predictions are done with the created model. Finally, the results are visualized in presentation layer (Birst). Each stage is described in its own section.

5.1 Data Integration

The data is imported from Valmet DNA. The data is gathered and saved into csv -files (Comma Separated Values) into one-day packages. The connection to the server is created with ODBC (Open Database Connectivity). The data import was done using R software. The data packages were named as described in Section 4.6.

In this thesis work, a live connection to the sites was not permitted. Thus, different from usual VII projects, the data was imported manually from info server and then transferred to AWS manually. This procedure is sufficient to illustrate the proof of concept of utilizing machine learning in Industrial Internet environment.

Usually, the imported data files have only one value of a tag on a row. In the case studied, the sampling period is one second and the number of imported tags is more than 20, so the size of the data packages would be too large. Therefore, the data is stacked so that there is one timestamp at each row but there are multiple columns each having one value of a specified tag at that timestamp. This decreases the size of the data packages significantly.

For the data, a table was created in RedShift to the staging area. In the table, the columns were defined before the actual data was transferred to RedShift. In the table, there are columns for datetime, mill id, line id and one column for each measurement.

The data was first transferred temporarily to S3 bucket. After the data is in S3, the data can be copied to RedShift table by running a command in RedShift browser (Aginity Workbench). The command in SQL language is the following:

copy staging.energyturbine_kpi from 's3://datasourcebucket/' iam_role 'arn:aws:iam::123456123456:role/redshift-copy-role' delimiter ',' csv ignoreheader as 1;

The arguments of copy command are: the destination where the data is copied, the data source, the definitions for AWS role, delimiter, format of source files, and finally command to ignore the header row in the source file (as headers were already defined when the table was created).

After the copying is completed, the files can be deleted from S3 bucket. This procedure for transferring data to RedShift is recommended only for large datasets. For smaller datasets, data flow described in Section 264.6 is used. If there is no live connection available, the transfer can be done by adding correctly formed csv files to S3 source bucket manually. The workflow is otherwise the same as the one described in Section 4.6.

5.2 Machine Learning Model Development

The machine learning model was created in R environment. For the model development, the data was read with R and the model was trained by using stepwise regression method. As a result, the R program generated a regression model with 16 predictors and the intercept (β_0). The model is predicting temperature corrected power output of the turbine.

First, the data was read from the cloud database, RedShift. For the training, only five-day period after gas turbine maintenance were selected.

After the data collection, the data was filtered. If the temperature corrected power is below 4500 kW or if the NOx control is above 20%, the corresponding data was excluded from the training data. The periods of low power output were left out as the model were to describe only the base load situation. The periods of high NOx control were left out because high NOx control indicates an abnormal turbine operation that the NOx control system is trying to compensate.

Before the model is, the training data is filtered with a non-causal moving average that computes the average of three past, the current and three future values. The moving average filtering was done to filter unwanted “spikes” or other abnormalities that can be caused by disturbances in measurement devices or other errors.

After the data was filtered, the model was trained with a stepwise regression algorithm applying bidirectional elimination. The algorithm needs a starting point (which variables to start with), and the lower and higher limits for the model complexity.

The starting point was the three variables: main gas valve position, air pressure after compressor, and pilot gas valve position. No predictors was chosen as the lower limit of complexity, and all the possible variables were chosen as the higher level of complexity.

The stepwise regression algorithm starts to either add or drop the variables from the model. As a result, the model ends up with a model that has 16 variables in it. The algorithm chose all the available variables for the model. That means that either all the variables are describing the predicted value well or the limits for accepting the variable in the model are too soft. The goodness of the model can be analyzed with the methods that were described in Section 3.3.

Based on the model, the program inserts the identified model parameters to file and saves the file to a table. In the table, there are columns for the model variables (X) and for estimates of parameters (β). The idea is that for making predictions, the user does not need the model in R (R-file) but he can do the prediction based on the configuration table and calculating the prediction for each new data instant by using Equation (3-1).

Another advantage of using the configuration file for making the predictions is that the data analyst can easily modify the current model without interrupting the predictions if they are running at the same time. The predictions are running based on the configuration file and in the background, the user can for example add more variables to the model or modify the training data. When the new model is trained, the user can just update the configuration file and after that, the predictions with the new model can be done.

The computation in the cloud starts to work with the new model instantly when the configuration is uploaded to the cloud. Existence of available tags in the systems must be ensured before adding new variables in the model. Otherwise the computation doesn't work. If the new model is put into service instantly, there is likely a step in the prediction. To avoid the step, the model should be deployed during a time when the machine is not running.

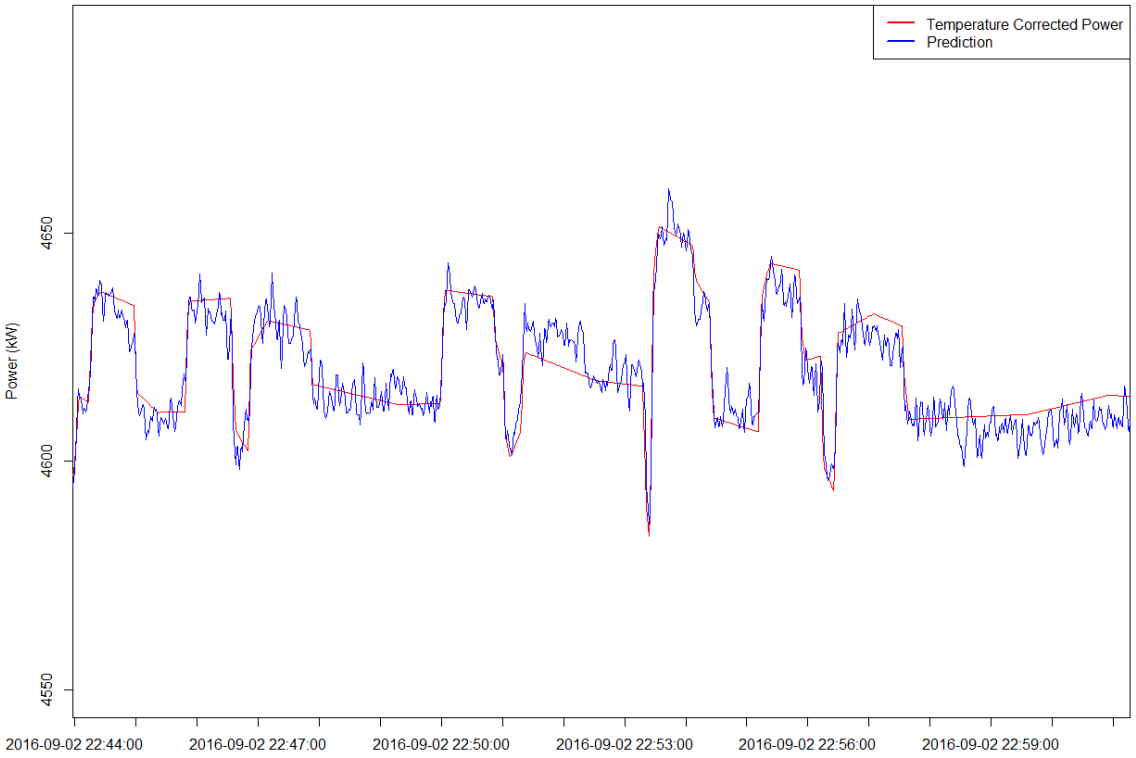

Figure 17. *Actual temperature corrected power and prediction. Data is validation data.*

Figure 17 shows an example of prediction made with the developed regression model. Figure 17 is zoomed from the plot. Actual time range for the validation data was from September 2, 2016 to September 7, 2016. The prediction follows the actual value well. With the validation data, median residual error of the prediction is -0,281 kW and R^2 value is 0,9988. Minimum residual is -28,590 kW and maximum residual is 40,525 kW in the range of validation data. Even the maximum residual is less than 1% of the total power output. Residual standard error is 6,715 kW.

For the predictions, prediction uncertainty can be calculated. By using Equation 3-11 it is possible to calculate certain confidence intervals for each prediction. Short time period of the predictions with 95% confidence interval are in Figure 18.

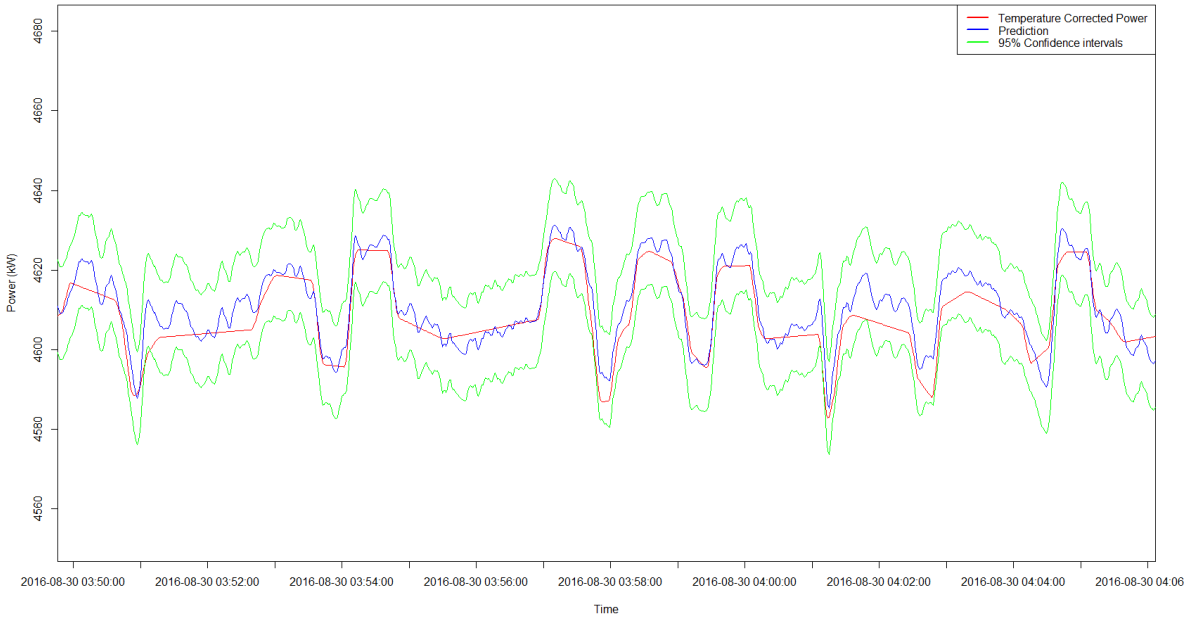

Figure 18. *Prediction with 95% confidence intervals and actual measurement.*

From Figure 18, it is possible to see that the actual temperature corrected power stays inside the confidence intervals most of the time. Based on the confidence interval analysis, it can be concluded that the model is describing the temperature corrected power output well enough and it can be used for detecting the decreased performance of the turbine. Reasons for a decrease in performance can be fouling of compressor, corrosion in the compressor or turbine blades and blocked gas pipes in compressor for example, see Chapter 2.

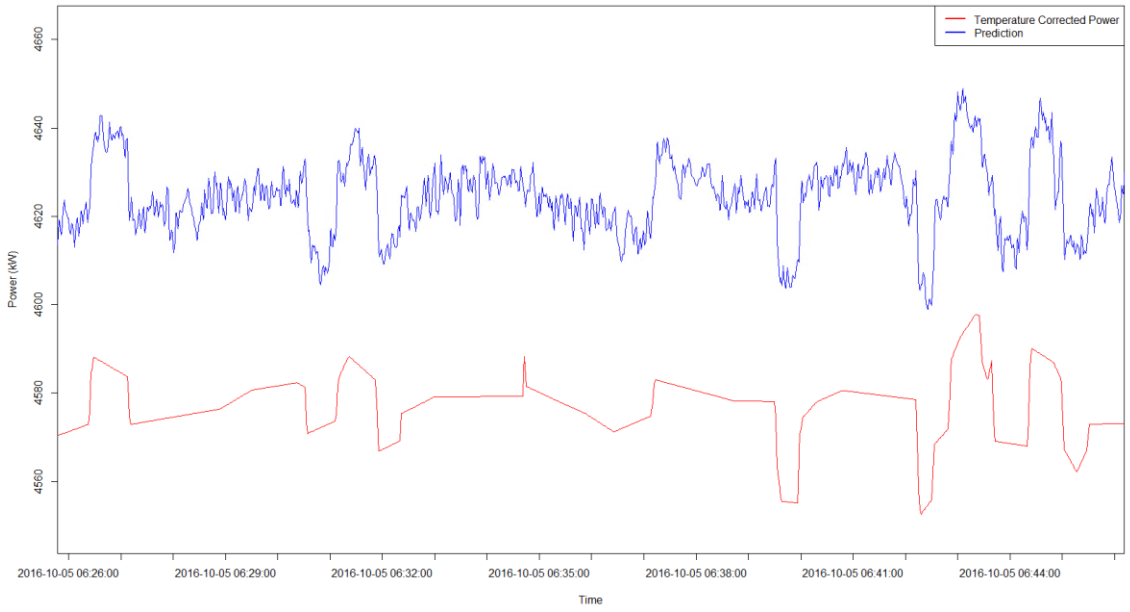

Figure 19. *Decreased performance.*

In Figure 19, the prediction is 40 to 50 kW higher than the actual temperature corrected power. The shape of temporal variation of the temperature corrected power and the prediction are similar so it can be assumed that the model is working but the performance of the turbine has decreased. From the data, it is not possible to recognize any certain point when the performance decreases significantly. The performance decreases slowly as time passes.

The model validity should be considered. The model has been made to work at the baseload and it might not work as well at lower loads because the model needs to extrapolate the prediction outside the area of the training data. The turbines that are included in this thesis are operating mainly at the baseload so it is not a major problem.

The training data was chosen from certain load range but there are also other measures that may affect to model accuracy. In the training data, the inlet air temperature was in a certain range and it varies with seasons. The predicted variable is temperature corrected but the correction is not perfect and thus extrapolation of inlet air temperature may cause some errors to the prediction.

One way to evaluate the model is to study the residual of the prediction. The error that is caused by a decrease of the performance can be extracted from the residual and it can be presented as trend. The residual is extracted by taking ten-minute moving average of the residual and it is removed from actual residual. Moving average is considered as decreased performance (offset). After that, the dispersion of the residual can be examined. Dispersion increasing over the time, i.e. its variation growing indicates that the model accuracy is decreasing.

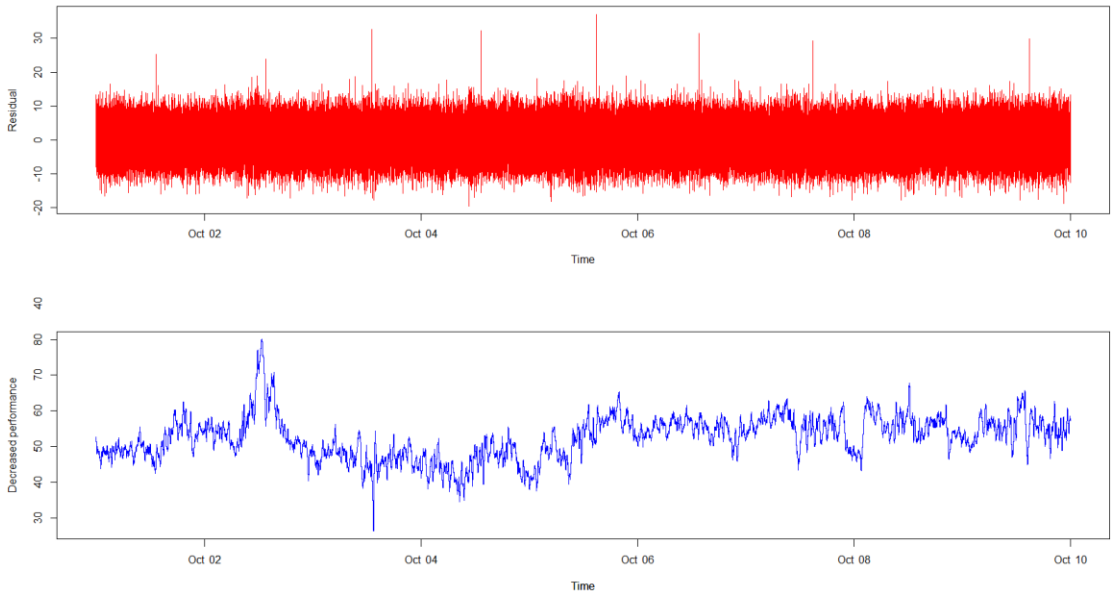

Figure 20. *Residual dispersion and decreased performance. Vertical scale in kW.*

In Figure 20 the upper graph presents the residual from which the moving average is subtracted and the lower graph presents the moving average itself, i.e. the decrease of the performance. The residual dispersion is stationary but there is variation in the curve of decreased performance. The variation in the curves is relatively small and it can be caused by errors in the model or disturbances in the process: there can be a temporary disruption in the fuel feed or impurities in the inlet air for example.

The long-term performance decrease can be judged by how the offset develops. Right after the overhaul the prediction and the actual value are in practice the same. The situation in Figure 19 and Figure 20 is two months after the overhaul. Already at the time, the decrease in the performance is more than one percent.

The performance can be improved with small maintenance breaks where compressor can be washed and turbine blades can be checked. The fouling of compressor is the most likely reason to decrease the performance and its effect can be mostly removed by washing the compressor.

With decreased performance analysis, it is possible to benchmark the turbines. The turbines are of the same type. Thus, when the turbines are working optimally, the output should be the same if the turbines have the same inputs.

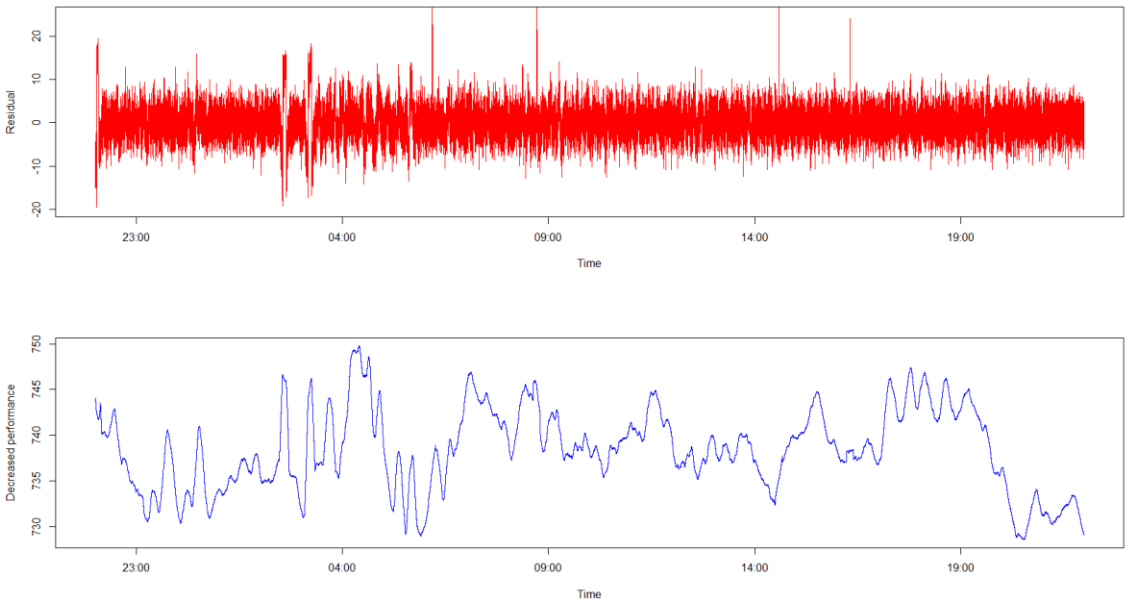

Figure 21. *Residual dispersion and decreased performance analysis done with another turbine. Vertical scale in kW. The data is from 29.10.2016*

Figure 21 shows residuals when the model identified for one turbine is applied to another. The decreased performance is over 700 kilowatts. The second turbine is from the same site as the one for which the model was trained. For this turbine, overhauls have not been done.

The dispersion width for the two turbines is the same in Figure 21 and Figure 20. The residual dispersion is about ±10 kilowatts. Because the dispersion is similar, the prediction seems reliable. The loss of 700 kilowatts in power is remarkable and to achieve the original level of produced power, thorough maintenance for the turbine is needed.

5.3 Data Arrangement in RedShift

The data that was dumped from S3 to RedShift is in staging area. As the efficient use of visualizations in Birst, the tables must be formed so that it is fast to read the values from the tables and in Birst, there is no need for tasks requiring major computing capacity.

The tables are moved to publish area in a form used in Birst. In Birst, the purpose is to get an overall picture of the process. There is no need to present the data in one-second intervals. For Birst, tables with one-minute interval and one-hour interval (averages) can be created.

Also, temperature corrected power output is calculated in RedShift. The calculation is done with SQL commands with actual power output and inlet air temperature as its inputs.

date_time	millid	lineid	date	hours	month	year	temperature_corrected_pow
2016-08-27 21:16:00	399410	2	2016-08-27	21	8	2016	4635,2408
2016-08-27 21:29:00	399410	2	2016-08-27	21	8	2016	4630,4395
2016-08-27 21:33:00	399410	2	2016-08-27	21	8	2016	4646,9550
2016-08-27 21:38:00	399410	2	2016-08-27	21	8	2016	4625,4905
2016-08-27 21:41:00	399410	2	2016-08-27	21	8	2016	4627,2424
2016-08-27 21:49:00	399410	2	2016-08-27	21	8	2016	4622,0302
2016-08-27 21:55:00	399410	2	2016-08-27	21	8	2016	4624,2746
2016-08-27 21:58:00	399410	2	2016-08-27	21	8	2016	4620,2328
2016-08-27 22:02:00	399410	2	2016-08-27	22	8	2016	4622,9810
2016-08-27 22:07:00	399410	2	2016-08-27	22	8	2016	4625,9807
2016-08-27 22:20:00	399410	2	2016-08-27	22	8	2016	4626,3144

Figure 22. *Example of a publish table in RedShift.*

In publish area, for each row there are columns for datetime, mill id, line id, date, hours, month, year and measurement data, such as temperature corrected power. In Birst, data can be filtered according to, for example, column data on mill id or date. A normal data view in Birst could with one-hour averages, and by clicking a certain hour, data from that hour could be opened. With month or year columns it is possible to calculate KPI values such as one year-average power output.

5.4 Model Deployment to Valmet Industrial Internet Environment

The predictions in VII environments are done with AWS Lambda function. The function reads the parameters of the regression model, queries the turbine data from RedShift and makes the prediction based on the data. Finally, the function returns the calculated predictions to RedShift to another table.

The parameters of the model are saved in a DynamoDB table. Another table is created to DynamoDB for the date when the prediction is done. AWS Lambda is triggered by AWS CloudWatch at user specified intervals. The Lambda function reads the parameters from the DynamoDB tables and makes the queries to the RedShift. The Lambda function computes the predictions for one day period at the time.

After the calculations are done and returned to the RedShift, the Lambda function updates the date in the DynamoDB table. When the CloudWatch triggers the Lambda again, the predictions are done for the next day compared to previous Lambda call.

The use of Lambda function with CloudWatch brings flexibility to the user. The user can do the calculations as frequently as needed. The calculations can be executed only once a day but it is possible to execute the calculations also every five minutes if needed. The system is scalable regardless of the number of the power plants connected to the system.

If there are many power plants integrated to the system that are using the same machine learning model, it is possible to create individual Lambda functions for each power plant. A large number of calculations in the same Lambda function lengthens the execution time of the function. The problem can be avoided by separating the calculations to several Lambda functions that are running parallel. The server costs of the Lambda functions are based on the calculation capacity that the functions are using so separating the calculations to several Lambda functions should not raise the price considerably.

5.5 Visualization

Visualization is implemented in Birst. The purpose of the visualization is to point out the user the overall condition of the selected turbine. Birst reads the data from RedShift and based on the data, KPI values and trends are presented as curves on the dashboards.

The visualization consists of two dashboards. The first dashboard is a general view that shows KPI values that are one day averages. The trends in the general view are one hour averages. The second dashboard with a more detailed view shows KPIs that are one hour averages and trends as one minute averages.

In the first dashboard, called “Turbine Performance”, there are three KPI values of the top of the dashboard: the average temperature corrected power, the lost power and the annual money loss. The average temperature corrected power shown is the one the turbine has produced during the selected day.

Lost power shows the average difference between the prediction and actual temperature corrected power output during the day. The lost power KPI has an indicator color: if the lost power is less than 50 kW, the color is green; if the lost power is between 50 kW and 200 kW, the color is yellow; above 200 kW, the color is red.

Annual money loss is calculated based on the lost power. The lost power is multiplied by the average electricity price, 30€/MWh, 24 (hours in a day) and 365 (days in one year). As a result, the KPI shows the annual earnings loss that is caused by decreased performance of the turbine.

Both dashboards can be filtered with date, mill id and line id. In addition, the detailed data dashboard can be filtered with a selected hour of the day.

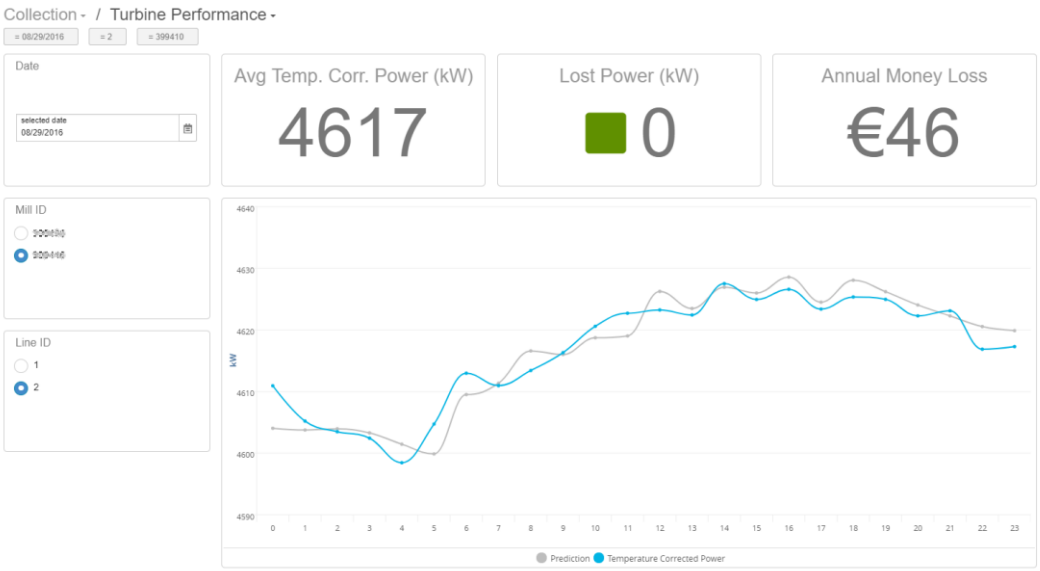

Figure 23. *"Turbine Performance" -dashboard.*

The "Turbine Performance"-dashboard with the general view of the turbine performance is shown in Figure 23. The dashboard is easy to read and at the top of the dashboard, there are the most important KPIs with large and clear texts. At the left-hand-side of the dashboard, the user can choose values for the filters. In Figure 23, the selected turbine is the one on which the overhaul was done and the model identified. The time instant is shortly after the overhaul so the prediction and the temperature corrected power are quite the same.

The other dashboard is called "Detailed Data". The dashboard has some of the same measures as the "Turbine Performance" dashboard but it has also some measurements that are describing the process. The dashboard is opened by clicking the trend values from the first dashboard. Birst transfers the filter values to the "Detailed Data" -dashboard automatically.

The structure of "Detailed Data" -dashboard is similar to the first dashboard. There are KPIs at the top of the dashboard. Below the KPIs there are the trends. From the filters, the user chooses the hour of which data is displayed on the dashboard. There is also secondary data in the database but it was chosen not to be shown in Birst.

From the detailed data, the user can examine if there are any abnormal values or odd behavior in the process. If needed, new filters can be added. With new filters, the user can filter the process data based on chosen conditions. For example, the user can choose only time periods when the inlet air temperature is above 20 °C or when the exhaust gas temperature is more than 450 °C.

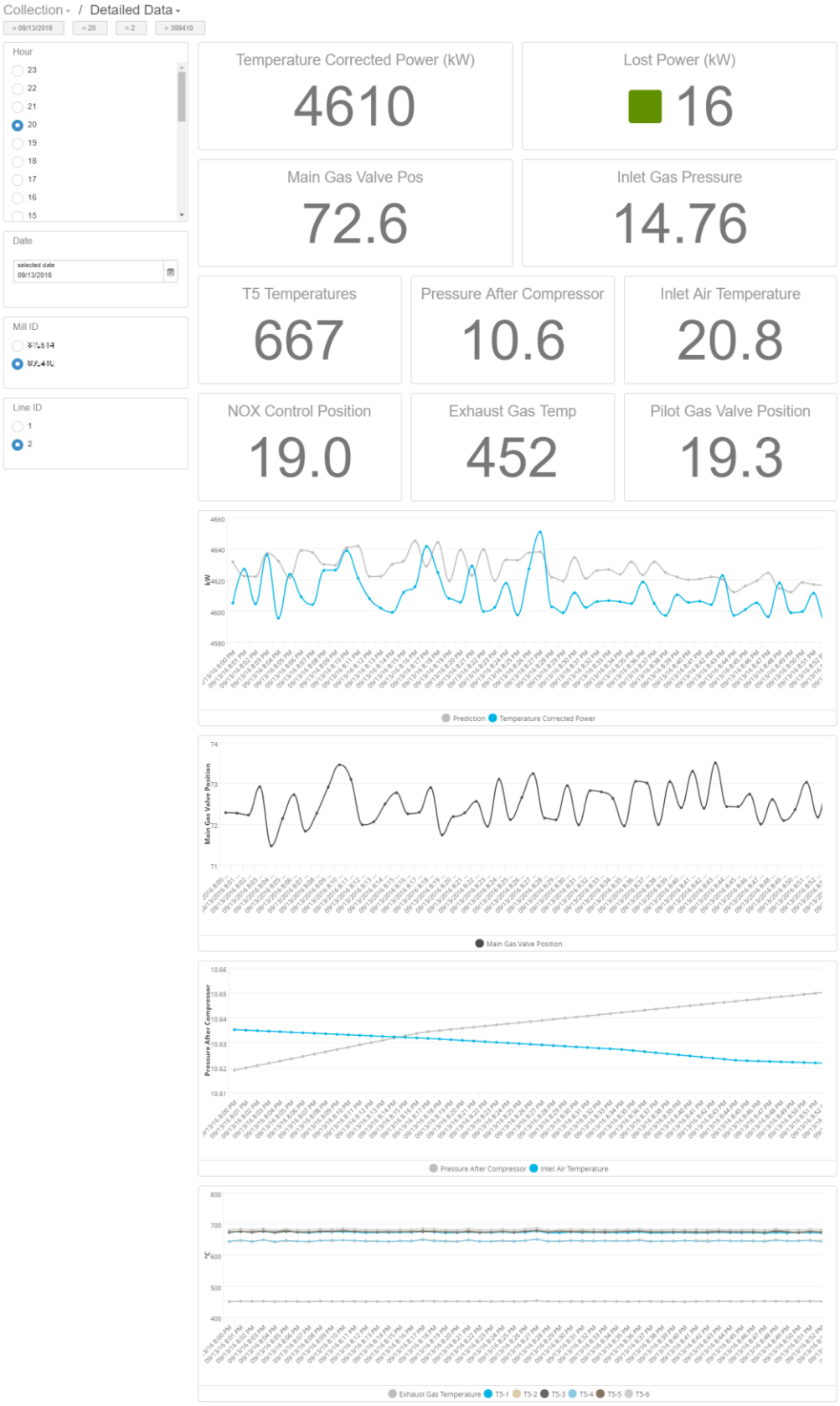

Figure 24. *"Detailed Data" -dashboard.*

"Detailed Data" -dashboard is shown in Figure 24. In addition to the one-hour averages that are as KPIs in the dashboard, the same values are also on the trends. Some of the values are combined in the same dashlet. For instance, T5 temperatures and exhaust gas temperature are wise to put in the same dashlet because they correlate between each other.

With Birst, benchmarking turbines is easy. All the turbines are connected to the same system. The user can easily benchmark the turbines at different times just by changing the filter values. If the user changes the line id from 2 to 1 in the dashboard, the values of the other turbine at the same mill at the same time are shown in the dashboard.

The system also shows the performance decrease of the turbines over the time. Based on the information provided by the system, the user (site manager) can do decisions when the maintenance breaks should be done for the turbines.

5.6 Implementing a Practical Decision Support System

In the system, there are two roles that are clearly different from each other. The first role is data analyst or developer that is responsible for everything inside the IIoT system. The second role is a user whose access rights are strongly limited.

With the permission of the customer, the data analyst creates a data pipeline from the site to the IIoT platform. The data analyst creates the data model and machine learning model. He controls the data inside the IIoT system. The data-analyst administrates the accesses regarding to data and the applications.

With the help of the customer, the data-analyst creates the user interface for the application. The customer (user) either approves the user interface or gives proposals for improvements. The customer is not permitted to make modifications to the user interface by him/herself. All the changes are done by the developer.

The end user has access to the raw data (the end user is working at the site) and the user interface that is showing the predictions and compares them to the actual measurement data. Everything between that is excluded. The data access was described in Section 4.8. Simplified, the user logs in to the service through the Salesforce.com. In the service, he has access to the user interface but he is not able to do any changes there.

There can be third role in the system: the domain experts. The domain experts can do deeper analysis for the data based on their domain expertise. Their expertise can be provided for the customer as an extra service. The domain experts help the data analyst to detect if the model needs to be upgraded. The data analyst upgrades the model using new training data or new specifications and deploys the upgraded model to the system.

The communication between the customer and the service provider is in important role. Especially in the developing phase, it is essential that the customer gives the service provider exact specifications about the system. The service provider is in response for guiding the customer to understand the system and its restrictions.

After the product (application) is deployed to production, the data-analyst or developer role is supportive. The data-analyst can make changes to the model or to the user interface if the user requests improvements. In that sense, the project resembles a normal software project: the product is developed and handed out for the customer. After that, the product provider continues the support of the product as long as needed or agreed.

At the developing phase, the data analyst is working with the product on daily basis. The data analyst may have scheduled meetings with the end-user where they follow the proceeding of the development. At the developing phase, the end-user is not in contact with the product on daily basis. After the production is deployed to production, the end-user can start the use of the product on daily basis.

6. CONCLUSIONS AND FUTURE WORK

The aim of this thesis was to develop a machine learning model in the gas turbine domain. The developed model predicts temperature corrected power output of a gas turbine. The model was deployed to Industrial Internet of Things (IIoT) environment. The model is a multiple linear regression model that was developed using stepwise regression -method.

The model and its implementation is a proof-of-concept. The implementation works as it was planned and during the project, the whole process from integrating the data from source system to visualizing the data in for the end-user was implemented. The implementation uses comprehensively components of Valmet Industrial Internet.

With the model, it is possible to point out decreased performance in a gas turbine. Compressor fouling, wear of machine parts and corrosion are decreasing the performance of the turbine. When the decreased performance is estimated, it is easier to decide on maintenance breaks.

The model was trained to work on the baseload. Most of the time, the operators are trying to achieve the maximum output from the turbine at the specific sites. The model is not therefore working on all loads. On other load areas, the model needs to extrapolate outside of the range from where the training data was from.

The data in this project was from two gas turbine power plant sites with two turbines each site. All the turbines are of the same the type (Solar Taurus 60) and therefore the model should work on each turbine. The model was trained from the data when the turbine is performing well. A live connection to the power plants was not permitted. Thus, data transfer from the site to the system was manual. For an actual customer project, the live connection will be built.

The machine learning model could have been analyzed further. The validity of the model was analyzed by taking the offset between the prediction and temperature corrected power output. After that, the residual dispersion was examined. The dispersion was stationary. Based on the dispersion analysis, the model appeared to be accurate enough. In addition to dispersion analysis, the user can examine the power plant data in Birst. In Birst, the user can look if there are any abnormal values in the data. In the future, the analysis of what variables are causing the performance decrease is needed.

From only one turbine overhaul data was available. The training data for the model was taken only from it. For verifying the model, testing the model with another turbine on which overhaul has been done would be a good way to validate the model.

6.1 Customer Benefits

The general idea of the developed machine learning model is to allow the user to monitor the decrease in the turbine performance. When the performance of the turbine decreases, the turbine is not producing as much electric power as it was producing earlier with the same inputs. The lost performance is lost money.

Based on a market data provided by Nord Pool Group [51], the day-ahead prices in the year 2017 were approximately 30 euros per MWh. If the turbine performance has decreased 200 kilowatts, in the course of one year the lost earnings, compared to the optimal performance of the turbine, would have been more than 50 000 euros.

The knowledge of the amount of lost earning helps the site manager to make decisions when to do the maintenance work. If the turbine is producing 4500 kilowatts on an average, the lost production earnings for one-day maintenance break would be 3240 euros. If the one-day maintenance break increases one year average performance by 13 kilowatts, the maintenance break has paid for its lost production time. The cost of maintenance work (employees) and spare parts were not included in this estimation.

6.2 Future Development

For future development, there are multiple possibilities. The model could be trained to work on much wider range of loads. Individual models for, for example 80-100%, 60-80% etc. from full load level could be identified. The application could then switch between the models depending on what load area the machine is running at the time. Alternatively, a nonlinear single model could be constructed to cover the whole load range.

There were also other possible machine learning applications for gas turbines that were discussed during this thesis, for example analysis of the start-up of the turbines. The application could compare the start-up times and start-up curves with selected measurements. The application could then recognize good start-ups and bad start-ups. The start-ups could then be benchmarked and the application could be used for finding if there are turbines which actuators are trimmed poorly or if there are other problems.

Another useful application for future development is compressor fouling analysis. If the compressor fouling and decrease of the compressor efficiency are known, the decisions of when the compressor is washed could be optimized.

REFERENCES

[1] International Energy Agency, “Key world energy statistics,” 2016.

[2] M. Chen, S. Mao, and Y. Liu, “Big Data: A Survey,” *Mob. Networks Appl.*, vol. 19, no. 2, pp. 171–209, Apr. 2014.

[3] K. P. Murphy, *Machine Learning : A Probabilistic Perspective*. MIT Press, 2012.

[4] Fei Chu, F. Wang, Xiaogang Wang, and S. Zhang, “A kernel partial least squares method for gas turbine power plant performance prediction,” in *2012 24th Chinese Control and Decision Conference (CCDC)*, 2012, pp. 3170–3174.

[5] Y. G. Li and P. Nilkitsaranont, “Gas turbine performance prognostic for condition-based maintenance,” *Appl. Energy*, vol. 86, no. 10, pp. 2152–2161, Oct. 2009.

[6] M. A. Zaidan, R. Relan, A. R. Mills, and R. F. Harrison, “Prognostics of gas turbine engine: An integrated approach,” *Expert Syst. Appl.*, vol. 42, no. 22, 2015.

[7] N. T. Davis and G. S. Samuelsen, “Optimization of gas turbine combustor performance throughout the duty cycle,” *Symp. Combust.*, vol. 26, no. 2, pp. 2819–2825, Jan. 1996.

[8] S. Kiakojoori and K. Khorasani, “Dynamic neural networks for gas turbine engine degradation prediction, health monitoring and prognosis,” *Neural Comput. Appl.*, vol. 27, no. 8, pp. 2157–2192, Nov. 2016.

[9] M. Amozegar and K. Khorasani, “An ensemble of dynamic neural network identifiers for fault detection and isolation of gas turbine engines,” *Neural Networks*, vol. 76, pp. 106–121, Apr. 2016.

[10] D. Zhou, H. Zhang, and S. Weng, “A New Gas Path Fault Diagnostic Method of Gas Turbine Based on Support Vector Machine,” *J. Eng. Gas Turbines Power*, vol. 137, no. 10, p. 102605, Oct. 2015.

[11] S.-M. Lee, W.-J. Choi, T.-S. Roh, and D.-W. Choi, “A study on separate learning algorithm using support vector machine for defect diagnostics of gas turbine engine,” *J. Mech. Sci. Technol.*, vol. 22, no. 12, pp. 2489–2497, Dec. 2008.

[12] W. Yan, “One-class extreme learning machines for gas turbine combustor anomaly detection,” in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 2909–2914.

[13] M. P. Boyce, *Gas Turbine Engineering Handbook*, 4th ed. Butterworth-Heinemann, 2012.

[14] G. F. M. de Souza, *Thermal Power Plant Performance Analysis*. London: Springer London, 2012.

[15] H. Cohen, G. F. C. Rogers, and H. I. H. Saravanamuttoo, *Gas turbine theory*, 4th ed. Longman, 1996.

[16] R. Kurz and K. Brun, "Fouling Mechanisms in Axial Compressors," *J. Eng. Gas Turbines Power*, vol. 134, no. 3, p. 32401, Mar. 2012.

[17] "Solar Turbines - Taurus 60." [Online]. Available: https://mysolar.cat.com/en_US/products/power-generation/gas-turbine-packages/taurus-60.html. [Accessed: 26-Sep-2017].

[18] J. Bell, *Machine Learning Hands for Developers and Technical Professionals*, vol. 7, no. 1. Wiley, 2015.

[19] M. H. Kutner, C. Nachtsheim, J. Neter, and W. Li, *Applied linear statistical models*, 5th ed. McGraw-Hill Irwin, 2005.

[20] "CRAN - Contributed Packages." [Online]. Available: https://cran.r-project.org/web/packages/. [Accessed: 29-Dec-2017].

[21] "R: Choose a model by AIC in a Stepwise Algorithm." [Online]. Available: https://stat.ethz.ch/R-manual/R-devel/library/MASS/html/stepAIC.html. [Accessed: 29-Dec-2017].

[22] ABB Group, "Internet of Things, Services and People - IoTSP." [Online]. Available: http://new.abb.com/about/technology/iotsp. [Accessed: 28-Sep-2017].

[23] "ANDRITZ presents 'Metris – Industrial IoT Solutions.'" [Online]. Available: https://www.andritz.com/group-en/news-media/news/2017-05-31-andritz-presents-metris-group. [Accessed: 28-Sep-2017].

[24] "The Industrial Internet | IIoT Insights | Industry 4.0 | GE Digital." [Online]. Available: https://www.ge.com/digital/industrial-internet. [Accessed: 28-Sep-2017].

[25] "Industrial Internet of Things by Honeywell." [Online]. Available: https://www.honeywellprocess.com/en-US/online_campaigns/IIOT/Pages/index1.html. [Accessed: 28-Sep-2017].

[26] A. Gilchrist, *Industry 4.0 : the industrial internet of things*. Apress, 2016.

[27] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[28] J.-Q. Li, F. R. Yu, G. Deng, C. Luo, Z. Ming, and Q. Yan, "Industrial Internet: A Survey on the Enabling Technologies, Applications, and Challenges," *IEEE Commun. Surv. Tutorials*, vol. 19, no. 3, pp. 1504–1526, 2017.

[29] L. Da Xu, W. He, and S. Li, "Internet of Things in Industries: A Survey," *IEEE Trans. Ind. Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.

[30] Valmet Corporation, "Valmet Industrial Internet offering launch." [Online]. Available: http://www.valmet.com/media/news/press-releases/2017/valmet-

launches-new-industrial-internet-offering-and-starts-partnership-with-tieto/. [Accessed: 11-Oct-2017].

[31] Valmet Corporation, “Valmet Industrial Internet.” [Online]. Available: http://www.valmet.com/about-us/industrial-internet/. [Accessed: 11-Oct-2017].

[32] Amazon Web Services, “Overview of Amazon Web Services,” 2017. [Online]. Available: https://aws.amazon.com/whitepapers/overview-of-amazon-web-services/. [Accessed: 11-Oct-2017].

[33] Amazon Web Services, “Amazon Route 53 Documentation.” [Online]. Available: https://aws.amazon.com/documentation/route53/. [Accessed: 30-Oct-2017].

[34] “SFTP File Transfer Protocol | SSH.COM.” [Online]. Available: https://www.ssh.com/ssh/sftp/. [Accessed: 30-Oct-2017].

[35] “SSH Protocol – Secure Remote Login and File Transfer | SSH.COM.” [Online]. Available: https://www.ssh.com/ssh/protocol/. [Accessed: 30-Oct-2017].

[36] W. H. Inmon and D. Linstedt, *Data architecture : a primer for the data scientist : big data, data warehouse and data vault*. Morgan Kaufmann Publishers, 2015.

[37] Amazon Web Services, “Amazon Simple Storage Service (S3) — Cloud Storage.” [Online]. Available: https://aws.amazon.com/s3/. [Accessed: 09-Nov-2017].

[38] Amazon Web Services, “What Is Amazon DynamoDB? - Amazon DynamoDB.” [Online]. Available: http://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html. [Accessed: 09-Nov-2017].

[39] Amazon Web Services, “What Is AWS Lambda? - AWS Lambda.” [Online]. Available: http://docs.aws.amazon.com/lambda/latest/dg/welcome.html. [Accessed: 09-Nov-2017].

[40] Amazon Web Services, “Amazon Simple Notification Service (SNS) Documentation.” [Online]. Available: https://aws.amazon.com/documentation/sns/. [Accessed: 09-Nov-2017].

[41] The Apache Software Foundation, “Apache Cassandra.” [Online]. Available: http://cassandra.apache.org/. [Accessed: 09-Nov-2017].

[42] W. W. Diab, K. E. Harper, S.-W. Lin, D. Nair, and W. Sobel, “The Industrial Internet of Things Volume T3: Analytics Framework,” *Ind. Internet Consort.*, 2017.

[43] Birst, “Business Intelligence Companies - Why Birst Overview.” [Online]. Available: https://www.birst.com/company/. [Accessed: 04-Dec-2017].

[44] Aginity, “Aginity — big data analytic solutions. | Aginity Workbench.” [Online]. Available: http://www.aginity.com/workbench/. [Accessed: 04-Dec-2017].

[45] The R Foundation, “R: The R Project for Statistical Computing.” [Online].

Available: https://www.r-project.org/. [Accessed: 04-Dec-2017].

[46] Amazon Web Services, “Configure a JDBC Connection - Amazon Redshift.” [Online]. Available: http://docs.aws.amazon.com/redshift/latest/mgmt/configure-jdbc-connection.html. [Accessed: 04-Dec-2017].

[47] A. Bosche, D. Crawford, D. Jackson, M. Schallehn, and P. Smith, “How Providers Can Succeed in the Internet of Things,” *Bain Br.*, 2016.

[48] A.-R. Sadeghi, C. Wachsmann, and M. Waidner, “Security and privacy challenges in industrial internet of things,” in *Proceedings of the 52nd Annual Design Automation Conference on - DAC ’15*, 2015, pp. 1–6.

[49] E. Harison, “Who owns enterprise information? Data ownership rights in Europe and the U.S.,” *Inf. Manag.*, vol. 47, no. 2, pp. 102–108, Mar. 2010.

[50] A. Gilchrist, *IoT security issues*, 1st ed. De/G Press, 2017.

[51] Nord Pool Group, “Electricity day-ahead prices.” [Online]. Available: https://www.nordpoolgroup.com/Market-data1/Dayahead/Area-Prices/ALL1/Yearly/?view=table. [Accessed: 31-Jan-2018].

APPENDIX A: MACHINE LEARNING MODEL PARAMETERS

VARIABLE	COEFFICIENT
INTERCEPT	-1590.78973303647
MAIN GAS VALVE POSITION	3.58507824297635
GAS PRESSURE AFTER COMPRESSOR	347.894926074804
EXHAUST GAS TEMPERATURE	0.655253916768893
PILOT GAS VALVE POSITION	-65.6484745163643
T5-1 TEMPERATURE	0.456418669395561
T5-2 TEMPERATURE	2.00299113511974
T5-3 TEMPERATURE	1.17260801113399
T5-4 TEMPERATURE	-0.943012344783304
T5-6 TEMPERATURE	3.29513975514825
INLET AIR TEMPERATURE	15.1629597840949
REACTIVE POWER	-0.0092938526370424
NOX CONTROL POSITION	-162.761608545487
INLET GAS PRESSURE 1	54.5933207064595
INLET GAS PRESSURE 2	-43.0806694156413
INLET GAS PRESSURE 3	5.98625453603192