ABSCHLUSSBERICHT

Entwicklung und Erprobung einer intelligenten Datenbank zur Optimierung des Life-Cycle-Managements von Windenergieanlagen unter Betrachtung von Zustandsinformationen extrem belasteter Komponenten und SCADA-Daten

Förderkennzeichen:8/12-16
Bewilligungszeitraum: 01.09.2015 – 31.08.2017 (Verlängerung bis 31.12.2017)

In Zusammenarbeit mit Fachhochschule Kiel
Projektleiter: Prof. Dr. Jens Lüssem

Bearbeiter: V. Vasundharan; D. C. Vallavaraj; S. Khan; J. K. Kröger; N. Loban; S. Y. Tan; N. Thomas; J. B. Gerth

Inhalt

1 Vorbemerkungen

In der Zusammenarbeit zwischen der CMC GmbH und der Fachhochschule Kiel wurden in diesem Projekt verschiedene Methoden betrachtet, wie Daten gespeichert sowie verarbeitet werden können. Dabei kann CMC stellvertretend für andere kleine und mittelständische Unternehmen betrachtet werden. Im Rahmen des Projektes sind sieben Abschlussarbeiten entstanden: zwei Bachelorthesen und fünf Masterthesen.

In dem Projekt wurden die Schwerpunkte Speicherung und Analyse von Daten für Windenergiesysteme bearbeitet mit dem Ziel der Optimierung des Lifecycles von Windenergiesystemen.

Der Fokus der Datenspeicherung liegt auf unterschiedlichen neuen Technologien aus dem NoSQL und *Big Data* Bereich, um eine intelligente Datenbank[1] zu entwickeln. Zur Analyse wurden verschiedene Methoden aus den Bereichen *Machine Learning* und *Deep Learning* angewendet.

Unsere Untersuchungen haben gezeigt, dass es sich für kleine und mittelständische Unternehmen rentieren würde, diese Methoden und Datenbanken einzusetzen. So können die eingesetzten Algorithmen (genauere und frühere) Vorhersagen über Trends liefern. Dabei verbessert sich die Vorhersagegüte mit der Anzahl der Daten und der aufgewandten Zeit, den entsprechenden Algorithmus für den Anwendungsfall anzupassen.

Adäquate Datenbanken vereinfachen das Auffinden von Daten deutlich. Mithilfe von neuen Datenbankkonzepten lassen sich Analysezeiten deutlich verbessern.

[1] Intelligente Datenbank = Eine Datenbank, die automatisch Analyseverfahren anwendet

2 Aufbau des Berichtes

Zunächst werden in den Kapiteln 3, 4 und 5 der Projektaufbau, die verschiedenen Projektziele sowie die angestrebten Meilensteine besprochen. (Basis: N. Loban)

Danach werden im Kapitel 6 und 7 die aktuelle Datenhaltung und ihre Verbesserung untersucht. (Basis: J. Gerth, S. Khan und N. Loban)

Im Kapitel 8 erörtern wir, wie Daten aus einer SQL-Datenbank zu einem Hadoop Cluster überführt werden können. (Basis: J. Gerth und S. Khan)

In den folgenden Kapiteln 9 – 11 werden die verschiedenen *Machine Learning* und *Deep Learning* Methoden erläutert. (Basis: V. Vasundharan, D. C. Vallavaraj, S. Y. Tan und N. Thomas)

Am Ende des Berichtes finden sich Schlussfolgerung aus dem Projekt im Kapitel 12 und Zukünftige Arbeiten im Kapitel 13. Ergänzt wird der Schluss mit einem Anhang, der unter anderem die Abstracts von den geschriebenen Thesen beinhaltet.

3 Projektziele

Im Projekt sollten folgende Ziele erreicht werden:

- Einarbeitung der Projektmitarbeiter in die Themengebiete der Windenergieanlagen, *Condition Monitoring*, Datenbanken und BI-Tools
- Erstellung bzw. Erweiterung der Softwareanforderungen in Form von User Stories
- Bestimmung der Windparks und Bereitstellung der Testdaten
- Analyse der User Stories und der vorhandenen Testdaten und Bestimmung der Datenqualität
- Analyse und Priorisierung der Datenbanksysteme für die zu entwickelnde Software
- Analyse und Priorisierung der *BI-Tools* für die zu entwickelnde Software
- Bestimmung der Datenanalyseverfahren
- Test der *BI-Tools* und die Umsetzung der ersten User Stories
- Implementierung von verschiedenen *Machine Learning* und *Deep Learning* Algorithmen für die *BI-Tools*
- Aufbau eines *Big Data* Systems zu Testzwecken

4 Projektplan

Dem Projekt liegt folgender Projektplan zugrunde. Jedes Arbeitspaket ist im Laufe der Arbeit einer feineren Planung unterzogen worden. Durch die klare Arbeitseinteilung konnten Arbeiten an manchen Arbeitspaketen vorgezogen werden. Im Weiteren sind der Gesamtplan und die Planung der einzelnen AP dargestellt.

4.1 Arbeitsmethoden

Für das Projekt wurde ein Projektplan nach dem Wasserfallmodell erstellt. Während der Durchführung wurde versucht, den Plan zu verfolgen. Durch die häufigen Wechsel von Studenten im Projektteam mussten einige Arbeitspakete wiederholt werden, um alle Beteiligten auf den aktuellen Projektstand zu bringen.

Zur Überwachung des Projektfortschritts wurden wöchentliche *Meetings* abgehalten. An den Meetings haben in der Regel die Studenten und Prof. Lüssem teilgenommen. Das Team aus Studenten hat zusammen in einem Projektraum in der Fachhochschule Kiel gearbeitet. Zusätzlich wurden regelmäßige *Meetings* bei CMC gehalten. Die Zeitabstände zwischen den *Meetings* waren dabei abhängig von der Verfügbarkeit von CMC und dem Projektteam sowie der Dringlichkeit der Fragen an CMC.

4.2 Meilenstein

MS-Nr.	**Meilenstein**
AP1-1	Workshop: Allgemeine Einführung in den Tätigkeitsbereich der CMC GmbH Behandelte Themen: • Vorstellung des Unternehmens • Strategien der Instandhaltung und Bereiche des CM • Überwachte Komponenten und CMS Hardware • CMS Software, Maschinendiagnose, Datenübertragung und Diagnosesicherheit • Praktische Beispiele • Life-Cycle-Management • Wirtschaftliche Betrachtung
AP1-2	Workshop: Datentransfer und Datenverwaltung Behandelte Themen: • CM Datenbanken • CM Datentransfer • MS SQL Dienste • CM Software
AP1-3	Workshop: Datentransfer und Datenverwaltung Behandelte Themen:

	• Zeitsignal • FFT-Analyse • Trends und Spektren • Diagnosen

Tabelle 4-1: Meilensteine

MS-Nr.	**Meilenstein**
AP2-1	Workshop: Stammdaten der WEA Behandelte Themen: • Bauformen der WEA • WEA mit Übersetzungsgetriebe • Baukonzept Turm • Rotor • Lagerungskonzepte • Getriebe • Generator • Energiegewinnung mit Synchrongenerator
AP2-2	Workshop: Stammdaten Getriebe Behandelte Themen: • Getriebekonzept • Überwachte Getriebekonzepte • Schwerpunkte der Überwachung • Warum Folgeschäden so gefährlich sind • Schadensverlauf Lager
AP2-3	Workshop: Kinematische Daten Behandelte Themen: • Lager • Getriebeverzahnung • Getriebeübersetzung
AP1-4	Workshop: Installation des CMS Behandelte Themen: • CMS-Auslieferungspaket • Konfiguration des Observers und der CMS-DB • Vorbereitung der Kommunikationseinheit • Konfiguration des Routers • Konfiguration des CMS • Verbindungscheck • Sensoren
AP2-5	Workshop: Condition Monitoring Behandelte Themen: • Prozess der Datenauswertung • Logbuch • Erstellung der Zwischenberichte • Erstellung der Monatsberichte
AP2-6	Workshop: Störungen am CMS Behandelte Themen:

	• Arten der CMS-Störungen • Kommunikationsstörungen • Störungsliste • Störungsberichte
AP2-7	Workshop: Aufbereitung der detektierten Schäden Behandelte Themen: • Schadensdokumentation

Tabelle 4-2: Meilensteine 2

5 User Stories

5.1 User Stories BI-Tool

US 1 Verbindungskontrolle

Ein CMI möchte automatisch benachrichtigt werden, wenn eine Kommunikationsstörung vorliegt, um die Kommunikationsverbindung zu kontrollieren und ggf. eine Störungsmeldung an den Kunden zu senden.

Auslösendes Ereignis: *keine Kommunikation zwischen dem CMS in der WEA und dem MSQL Server der CMC GmbH*

US 1-1: Jeden Morgen führt ein Diagnoseingenieur die Verbindungskontrolle zur Datenübertragung von den verbauten CMS zum MSQL Server der CMC GmbH durch. CMI öffnet den Front-End der Software und sieht in der

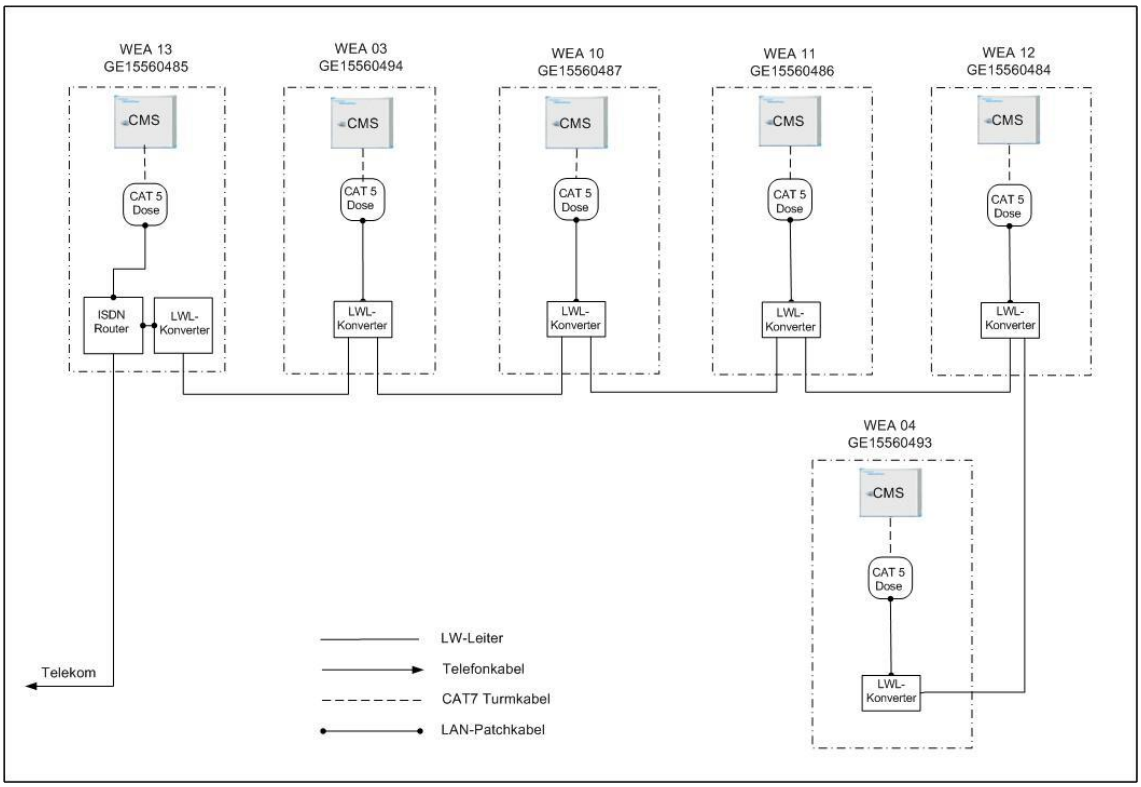

Abbildung 5-1: Kommunikationsanbundeung bzw. Parkvernetzung Test- WP Hanstedt II

allgemeinen Übersicht eine Alarmmeldung für Verbindungsstörung.

US 1-2: CMI öffnet die Alarmmeldung. In dem gleichen Fenster wird eine automatisch erstellte Liste der WEA ersichtlich, die sich seit mehr als 12 Stunden nicht gemeldet haben. Weiterhin ist aus der Liste zu erkennen wann das letzte Mal sich das CMS und mit welcher IP-Adresse gemeldet hat. Dabei muss berücksichtigt werden, dass die GSM-Verbindung permanent besteht, die Systeme mit ISDN-Anschluss melden sich am Server alle 12 Stunden. Weiteres

Kriterium für die angezeigte Reihenfolge der angezeigten Kommunikationsstörungen ist der Grad der detektierten Auffälligkeiten am Antriebstrang der betroffenen WEA. Somit sollen die vorgeschädigten WEA höhere Priorität erhalten.

US 1-3: CMI öffnet mit einem Doppelklick eine Kommunikationsstörung. In dem neuen Fenster werden die gewählte Kommunikationsstörung und die dazugehörigen Kommunikations- und Netzwerkdaten wie IP-Adressen und Parkvernetzung (Abbildung 5-2) der betroffenen WEA und weiteren WEA im WP (können nicht unbedingt von der Störung betroffen sein) angezeigt. Der CMI sieht alle nötigen Daten um das betroffene CMS manuell über cmd.exe zu

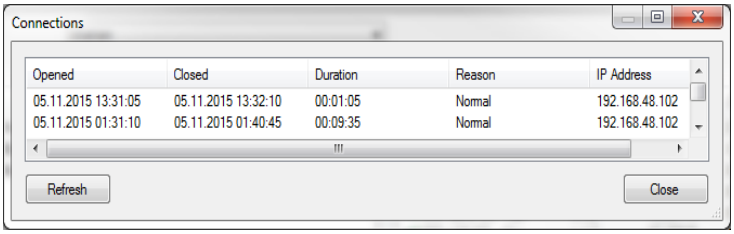

Abbildung 5-2: Manuelle Kommunikationskontrolle über @ptitude Observer

pingen.

Hinweis: Verbindungskontrolle erfolgt zurzeit in folgenden Schritten. Prüfung der Verbindung jeder WEA im @ptitude Observer 8.5; keine Verbindung ab 48 Stunden wird als potentielle Verbindungsstörung eingestuft; Überprüfung im LAN-Monitor ob einer der zwei verfügbaren ISDN Kanäle frei sind; mittels der zugehörigen IP Adressen die CMS oder die Router pingen; kann keine Kommunikation aufgebaut werden, Störungsmeldung per Email an den Kunden und die autorisierte Institution senden.

US 1-4: Nach dem Pingen ist die WEA immer noch nicht erreichbar. Der CMI geht auf die Schaltfläche „Störung melden“. Es wird eine automatische Email mit den Angaben für WP, WEA und dem Datum der Verbindungsstörung an den in Stammdaten hinterlegten Ansprechpartner konfiguriert. Der CMI bestätigt

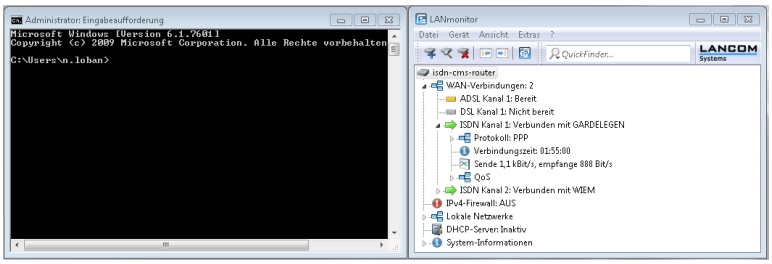

Abbildung 5-3: Rechts - Kontrolle der Verfügbarkeit der ISDN-Kanellen; Links - Eingabeaufforderung (cmd.exe) für das Pingen Systeme

die Angaben und die E-Mail wird versendet.

US 1-5: CMI geht auf „Störung registrieren“. Die Kommunikationsstörung wird in der Störungsliste registriert.

US 1-6: Das Pingen ist erfolgreich. Der CMI entscheidet sich für die weitere Beobachtung der Anlage und stuft diese als „weiter beobachten“ ein. Die WEA steht jetzt am Anfang der Liste und hebt sich durch ein Sonderzeichen von anderen ab.

US 1-7: Das Pingen ist erfolgreich. Der CMI entfernt die WEA aus der Liste.

US 2 Trendrückgang

Der CMI möchte automatisch benachrichtigt werden, wenn bei der Zustandsüberwachung ein Trendrückgang festzustellen ist, um die Warn- und Alarmgrenzen an das neue Trendniveau rechtzeitig anpassen zu können (Abbildung 5-4). Ein Trendrückgang kann durch abschwächende Auffälligkeit, Komponententausch oder Wartung und durch einen abgeschlagenen Sensor verursacht werden.

Auslösendes Ereignis: *Trendrückgang eines Signals*

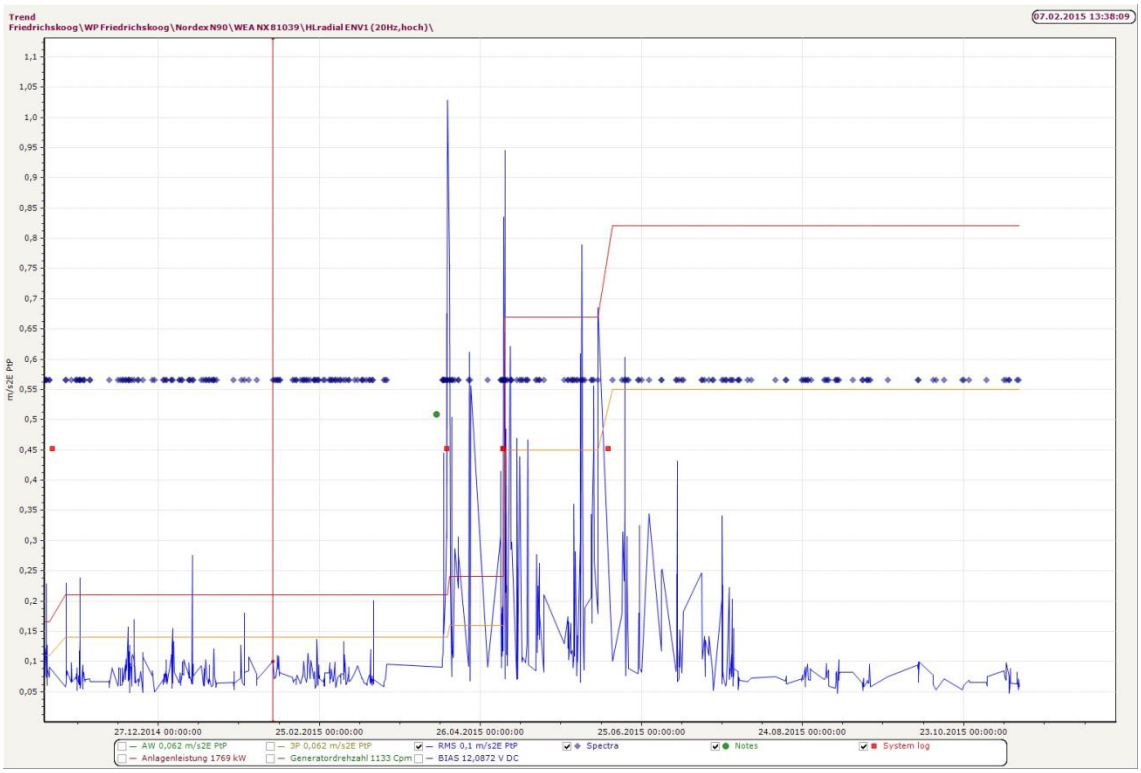

Abbildung 5-4: RMS-Trendrückgang, Anpassung der Grenzen erforderlich

US 2-1: Jeden Morgen führt ein Diagnoseingenieur die Alarmkontrolle durch. Die Trendanstiege werden im @ptitude Observer durch Warn- und Alarmgrenzen überwacht. Die Trendrückgänge werden von der neuen Software erkannt. CMI öffnet den Front-End der Software und sieht Meldungen für den Trendrückgang.

US 2-2: Der CMI öffnet die Meldung. Es wird eine Liste der betroffenen WEA mit folgenden Angaben angezeigt: Windpark, WEA, Messpunkt, Trend.

US 2-3: Zu jedem Trend macht die Software automatisch ein Vorschlag für das neue Niveau der Warn- und Alarmgrenzen.

US 2-4: Der CMI orientiert sich an den Angaben der neuen Software. Öffnet @ptitude Observer und passt die Warn- und Alarmgrenzen für jeden angegebenen Messpunkt an und markiert die Meldung als bearbeitet. Dabei entscheidet der CMI selbst, ob er den Empfehlungen für neue Grenzen folgt oder nicht.

Hinweis: Die Alarmmeldungen des @ptitude Observers weisen nur auf die Trendanstiege zu. Die Trendrückgänge werden nur durch einen Zufall entdeckt. Auf untere Warn- und Alarmgrenzen im @ptitude Observer wird verzichtet, da sonst zu viele Meldungen erzeugt werden.

Bei einem abgeschlagenen bzw. abgefallenen Sensor sind die Trendrückgänge oder sprungartige Trends an allen Messpunkten eines Sensors zu beobachten. Die Hierarchie ist in der Abbildung 5-5 dargestellt.

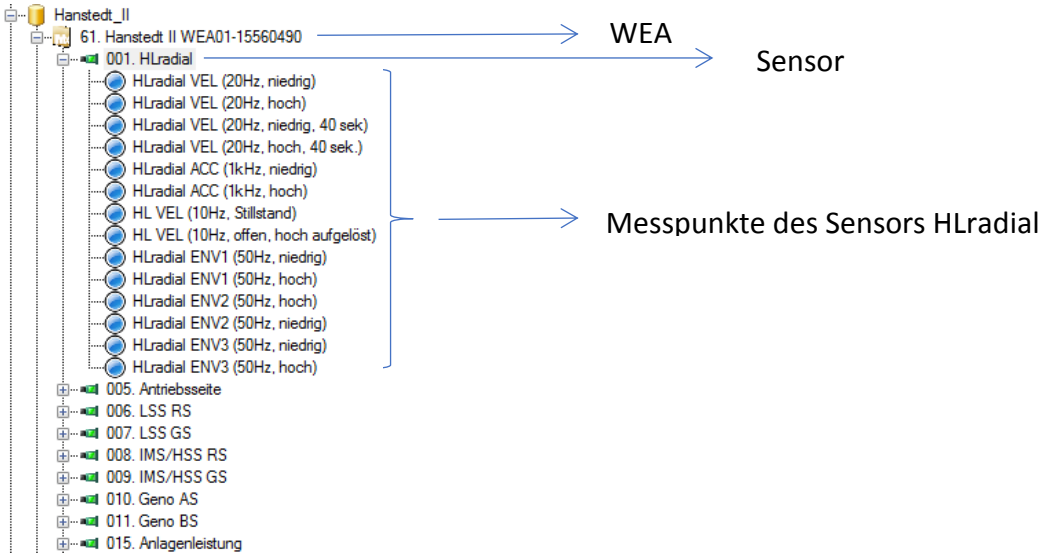

Abbildung 5-5: Hierarchie einer WEA in @ptitude Observer

US 3 Priorisierte Kontrolle der Warn- und Alarmmeldungen

Ein CMI möchte eine Prioritätenliste der vorliegenden Warn- und Alarmgrenzen haben, um die WEA mit den meisten und schwerwiegenden Meldungen als erstes bearbeiten zu können.

Auslösendes Ereignis: *tägliche Kontrolle der Warn- und Alarmgrenzen*

US 3-1: Zum größten Teil des Tagesgeschäfts des CMI gehört die Kontrolle der Warn- und Alarmmeldungen in der Zustandsüberwachungssoftware @ptitude Observer. Um die WEA mit den größten und schwerwiegenden Alarmen nach Priorität kontrollieren zu können, öffnet der CMI die neue Software. Im Front-End wählt er den Bereich „Kontrolle Warnmeldungen“.

US 3-2: In dem gleichen Fenster erscheint die Liste der WEA. Die WEA mit der höchsten Priorität werden oben angezeigt.

Priorität	WEA	WP	Alarm	Beobachtung
1	GE15560485	Hanstedt II	46	
2	V40887	Niebüll	40	x
n	n	n	n	

Tabelle 5-1: mögiche Angaben in der Prioritätenliste

US 3-3: CMI orientiert sich an den Vorgaben der neuen Software und bearbeitet die Warn- und Alarmmeldungen in @ptitude observer. Die kontrollierte WEA stuft der CMI als „bearbeitet“ ab. Diese wird an das Ende der Liste verschoben.

US 3-4: Bei der nächsten WEA sind weitere Beobachtungen bei der Trendentwicklung (oder weiteren Signalen) erforderlich. CMI stuft die WEA als „weiterhin beobachten“. Im neuen Fenster kann der CMI die WEA Daten mit den Daten des Messpunkts und des Signals ergänzen. CMI speichert die Meldung ab, Fenster geht zu (Tabelle 5-1).

WEA	WP	Messpunkt	Signal	Bemerkung
V40887	Niebüll	PS01-PIN ENV2 (50Hz, hoch)	Trend RMS	Freitext

Tabelle 5-2: mögiche Angaben für weitere Beobachtungen

US 3-5: Bei der Alarmkontrolle am nächsten Tag kann der CMI sich die gemachten Bemerkungen sowohl in der Prioritätenliste als auch in einer Sonderliste Anzeigenlassen. Der Doppelklick auf eine Meldung öffnet ein Fenster mit Details (Tabelle 5-2).

Hinweis: mögliche Priorisierung der Alarme

Priorität 1: langsam drehender Bereich

Priorität 2: mittelschnell drehender Bereich

Priorität 3: schnell drehender Bereich

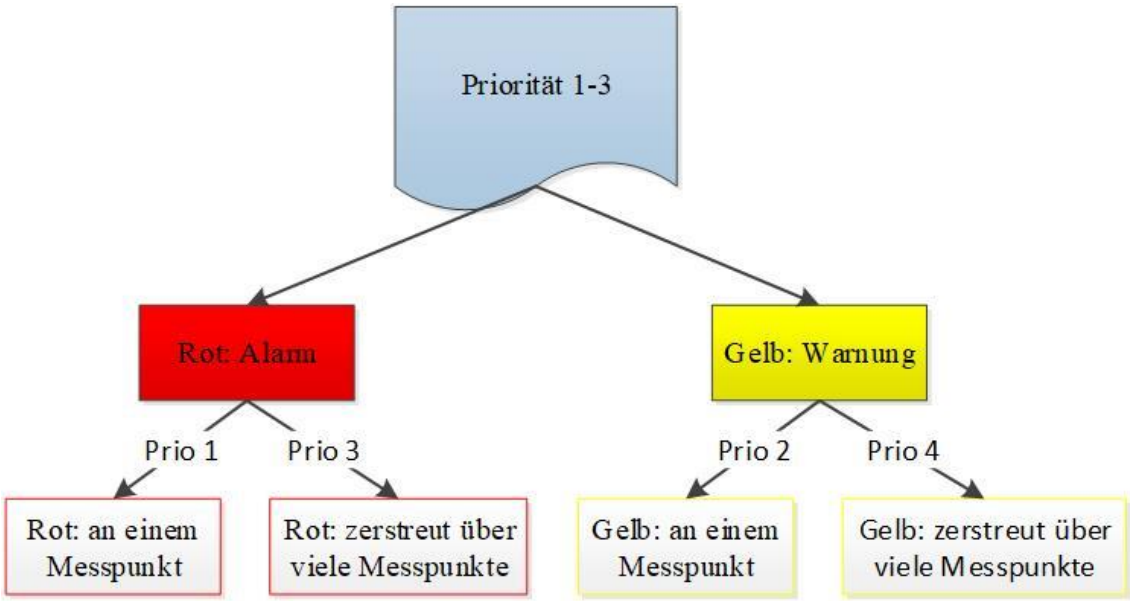

Abbildung 5-6: Prioritäten bei der Kontrolle der Warn- und Alarmmeldungen

Weiterhin werden die Alarme gesondert betrachtet. Die an einem Messpunkt gehäuften Alarme (rot) haben die höchste Priorität. Danach folgen die gehäuften Warnungen (gelb). Abbildung 5-6 erläutert das Prinzip.

US 4 Drehzahlsignalstörung

Der CMI möchte über die Störung des Drehzahlsignals automatisch informiert werden, um rechtzeitig Gegenmaßnahmen anleiten zu können.

Auslösendes Ereignis: *Störung des Drehzahlsignals*

Hinweis: Die Drehzahlsignalstörung ist in diesem Fall als 0 Umdrehungen/min 24 Stunden lang oder als durchschnittlich viel geringere Drehzahl als die der Nachbaranlagen definiert.

US 4-1: Der CMI startet die neue Software, die mit einer Periode von 24 Stunden die Drehzahlkontrolle durchführt, und bekommt eine Meldung für die Drehzahlsignalstörung.

US 4-2: Der CMI öffnet die Meldung und bekommt eine Liste der betroffenen WEA. Im @ptitude Observer kontrolliert er die Drehzahl. In einem Störungsfall geht CMI auf „Störung melden“. Es wird eine automatische Email mit den Angaben für WP, WEA und dem Datum der Drehzahlsignalstörung an den in Stammdaten hinterlegten Ansprechpartner konfiguriert. Der CMI bestätigt die

Abbildung 5-7: Beispiel eines Drehzahlsignals

Angaben und die E-Mail wird versendet.

US 4-3: Der CMI geht auf „Störung registrieren“. Die Drehzahlsignalstörung

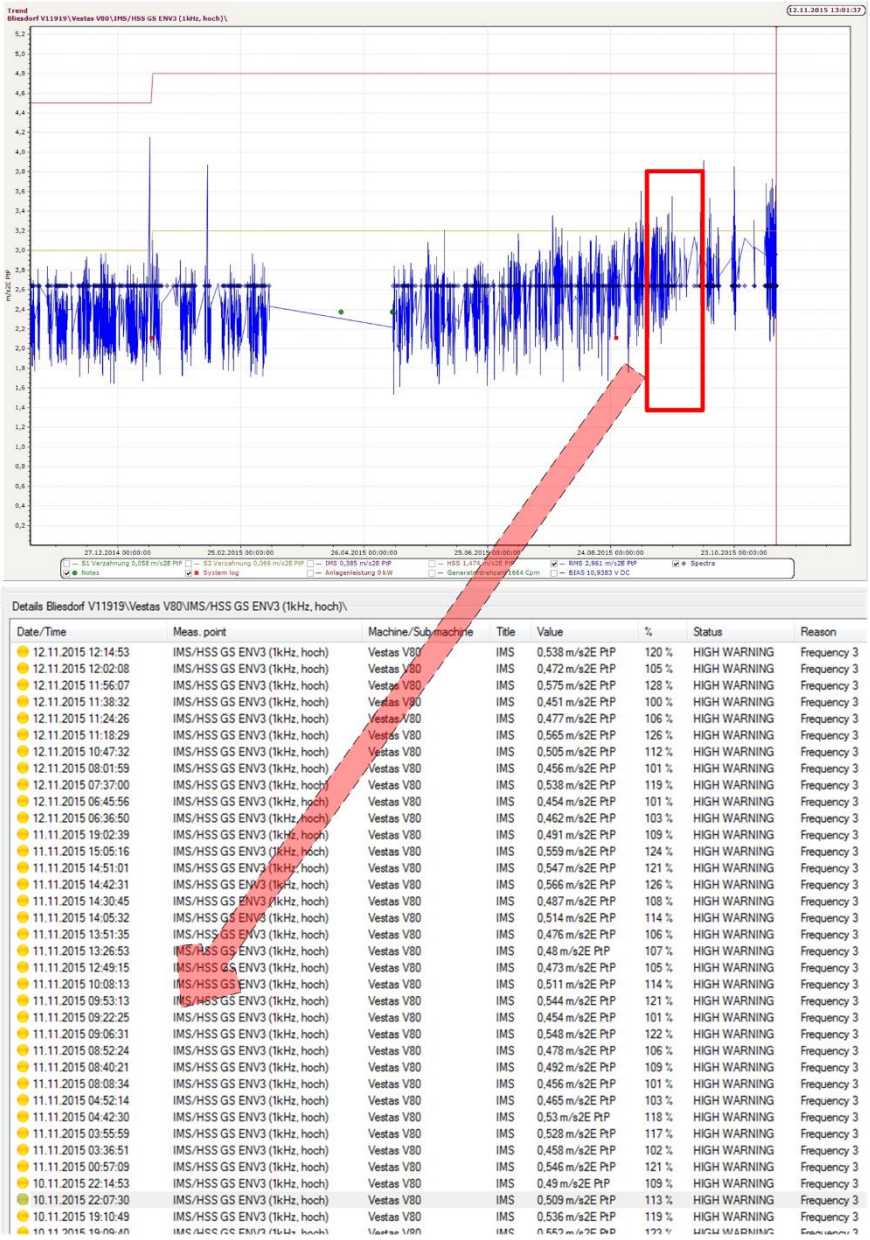

Abbildung 5-8: Mehrfache Überschreitung der Warngrenze eunes Trends und die entsprechende Warnmeldungen – Enstufung als Prio 1

wird in der Störungsliste registriert.

US 4-4: Die kontrollierte WEA stuft der CMI als „bearbeitet“ ab. Diese wird an das Ende der Liste verschoben.

US 5 nicht ausreichende Datenmenge an einem Messpunkt

Der CMI möchte über die Messpunkte informiert werden, die über eine unzureichende Menge an Daten verfügen, um rechtzeitig neue Messpunktkonfiguration vornehmen zu können.

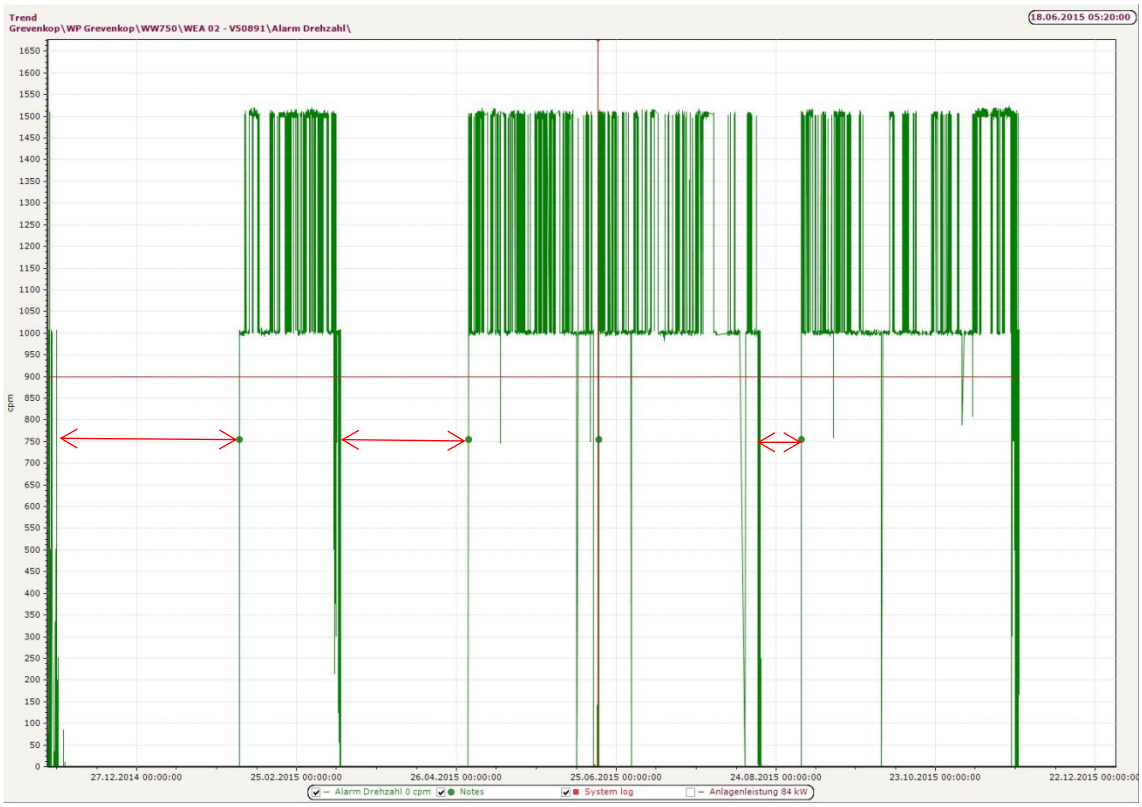

Abbildung 5-9: Drehzahlsignal mit drei Signalausfällen wegen verunreinigter Drehzahlerfassung

Auslösendes Ereignis: *nicht ausreichende Datenmenge*

Hinweis: Die Signale eines Messpunkts mit einer nicht ausreichenden Datenmenge sind in der Abbildung 5-9 dargestellt. Für die meisten WEA werden beim CM zwei Überwachungsbereiche auf der WEA-Leistungskennlinie angelegt, s.g. hohe und niedrige Klasse (vgl. Abbildung 5-10). Das heißt, die Daten werden nur bei einer bestimmten Drehzahl und der Anlagenleistung aufgenommen und ausgewertet. Verschiebt sich die Kennlinie der WEA, so werden nicht genug Daten für die Auswertung vorhanden sein. Die Klassen müssen angepasst werden.

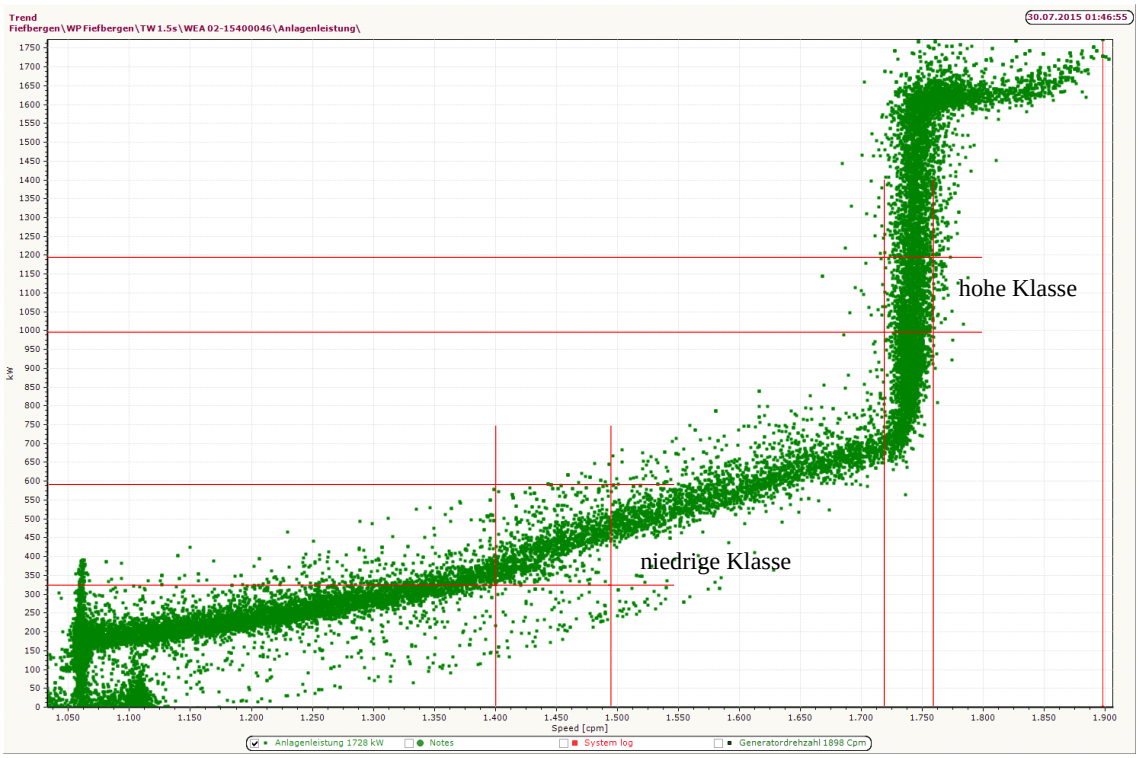

Abbildung 5-10: Leistungskurve der WEA TW 1.5s mit den definierten Klassen

US 5-1: Der CMI startet die neue Software, die mit einer Periode von 7 Tagen (muss besprochen werden) die Trendkontrolle durchführt, und bekommt eine

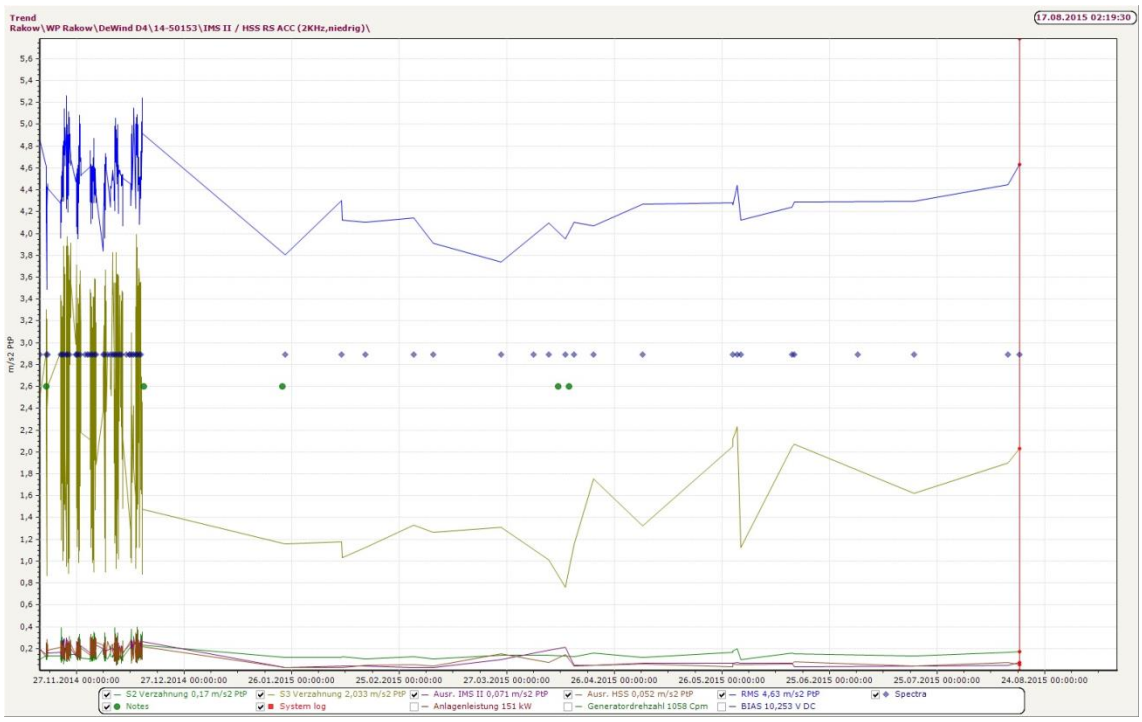

Abbildung 5-11: Messpunkt mit unausreichender Datenmenge

Meldung für die nicht ausreichende Datenmenge.

US 5-2: CMI öffnet die Meldung und bekommt eine Liste der betroffenen WEA mit dem Namen der Messpunkte.

Fall 1 – Die Leistung wird vom CMS erfasst: Der CMI öffnet eine Meldung. Die aktuelle Leistungskennlinie mit den definierten Klassen (ähnlich Abbildung 5-10) werden angezeigt. Gleichzeitig wird ein Vorschlag für die neuen Klassen gemacht.

Fall 2 – Die Leistung wird nicht vom CMS erfasst: Der CMI öffnet eine Meldung. Dem CMI wird vorgeschlagen die Kennlinienwerte (Leistung und Drehzahl) hochzuladen. Der CMI fordert von der TB die entsprechenden Daten im Excel Format an. Der CMI wählt die auf dem File Server gespeicherten Kennwerte aus und lädt diese hoch. Die aktuelle Leistungskennlinie mit den definierten Klassen (ähnlich Abbildung 5-10) werden angezeigt. Gleichzeitig wird ein Vorschlag für die neuen Klassen gemacht.

US 5-3: Die kontrollierte WEA stuft der CMI als „bearbeitet“ ab. Diese wird an das Ende der Liste verschoben.

US 6 Frequenzüberlagerung

CMI möchte einzelne Getriebemesspunkte auf die Überlagerung der Frequenzen

der Getriebekomponenten im Spektrum überprüfen lassen.

Abbildung 5-12: Überlagerung der Frequen IMS II und der vierten Harmonischen der IMS I Frequenz

US 7 Graphische Darstellung der WEA auf der Weltkarte

CMI möchte eine visuelle Übersicht der WEA auf der Weltkarte haben, um

- *Standort der WEA schnell ermitteln zu können*
- *den CM-Status der WEA schnell erkennen zu könne*
- *über im WP herrschende Wetterbedingungen vor Ort schnell informiert zu werden*

US 7-1: CMI öffnet die neue Software. Auf einer Karte sind alle im System registrierte Windparks bzw. WEA angezeigt.

US 7-2: Über z.B. ein Ampelsystem wird der Status der sich in der online Messung befindenden WEA angezeigt.

US 7-3: Der CMI kann wählen, welche WEA angezeigt werden sollen: online CM, offline CM, VE, SB usw.

US 7-4: Der CMI fährt mit der Maus über das Symbol einer WEA. In einem Pop-Up-fenster werden die wichtigsten Informationen der WEA angezeigt.

US 7-5: Mit einem Doppelklick auf die WEA wird zu den Stammdaten der WEA gewechselt.

US 8 Erkennung der Auffälligkeiten nach Muster

Ein CMI möchte die Überwachungsdaten nach den detektierten Schadensmustern von der neuen Software analysieren lassen, um Hinweis auf ähnliche Schadensentwicklungen bei anderen WEA zu bekommen.

Auslösendes Ereignis:

US 8-1: Ein CMI führt die tägliche Kontrolle der Alarm- und Warnmeldungen im @ptitude Observer. Dabei wird eine eindeutige Auffälligkeit an einer Komponente des Antriebstrangs festgestellt. CMI startet die neue Software und lädt/eingibt die Parameter der detektierten Auffälligkeit ins System. Daraufhin wird die Analyse der Überwachungsdaten auf ähnliche Schadensvorgänge analysiert.

US 8-2: Der CMI kann entscheiden, ob die WEA mit dem gleichen Designe des Antriebstrangs und den identischen kinematischen Daten analysiert werden, oder nur WEA mit der gleichen Komponente.

US 8-3: Der CMI bekommt eine Trefferliste der WEA als Ergebnis der Analyse.

US 8-4: Der CMI speichert die detektierte Auffälligkeit als ein Analysekriterium für die Zukunft in einem Katalog.

Hinweis: Eigentlich bei einer entsprechenden Aufbereitung der detektierten und bestätigten Auffälligkeiten bzw. Schäden in einer Schadensdatenbank kann diese Information in der Zukunft für die automatische Analyse eingesetzt werden. Somit bekommt der CMI die Möglichkeit das Analyse- bzw. Suchkriterium gezielt zu bestimmen.

US 8-5: CMI sucht in der Schadensdatenbank ein Lagerschaden des bestimmten Getriebes. Dieser Schaden wird als ein Analysekriterium definiert. System führt eine Analyse der aktuellen Überwachungsdaten, um gleiche Schadensentwicklung zu finden.

US 9 Vorhersage der Grenzüberschreitung im langsam drehenden Bereich des Antriebstrangs

Ein CMI möchte eine Vorhersage für den Zeitpunkt der Überschreitung der Warn- und Alarmgrenze durch einen Trend im langsam drehenden Bereich um schwerwiegende Auffälligkeiten rechtzeitig zu detektieren.

Hinweis: Der Trendanstieg im langsam drehenden Bereich ereignet sich langsam über längere Zeitperiode und ist wegen der Betrachtung bei der Datenauswertung eines Zeitraums von 12 Monaten, meistens nicht sofort zu erkennen. Ein rascher Anstieg des Trends im langsam drehenden Bereich deutet auf viel gravierende Auffälligkeiten, als im schnelldrehenden Bereich.

US 10 Berechnung der Frequenzbänder über Dreh- und Lagerfrequenzen

CMI möchte automatisch aus den hinterlegten kinematischen Daten Frequenzbänder für die Erstellung der neuen Messpunkte berechnen lasse, um den Prozess der Neukonfiguration der Messpunkte zu beschleunigen und zusätzliche Fehlerquelle auszuschließen. B1 – Frequenzband über Drehfrequenzen, die am jeweiligen Messpunkt überwacht werden sollen, B2 – Frequenzband über Lagerfrequenzen, die am jeweiligen Messpunkt überwacht werden sollen.

Hinweis: Die Berechnung basiert auf den höchsten und niedrigsten Frequenzen der Lager bzw. der Wellen. Aus diesen Frequenzen wird der Multiplikator (Mittelfrequenz) berechnet:

$f_h - höchste\ Frequenz$

$f_n - niedrigste\ Frequenz$

$f_m - Mittelfrequenz\ oder\ Multiplikator$

$$f_m = \left(\frac{f_h - f_n}{2}\right) + f_n$$

Höchste Frequenz bei den Lagern ist immer Innenringfrequenz. Der Suchbereich bzw. Spannbreite oder *range* ist die doppelte Differenz zwischen der höchsten und niedrigsten Frequenz plus 5%:

$$(f_h - f_n) * 1{,}05$$

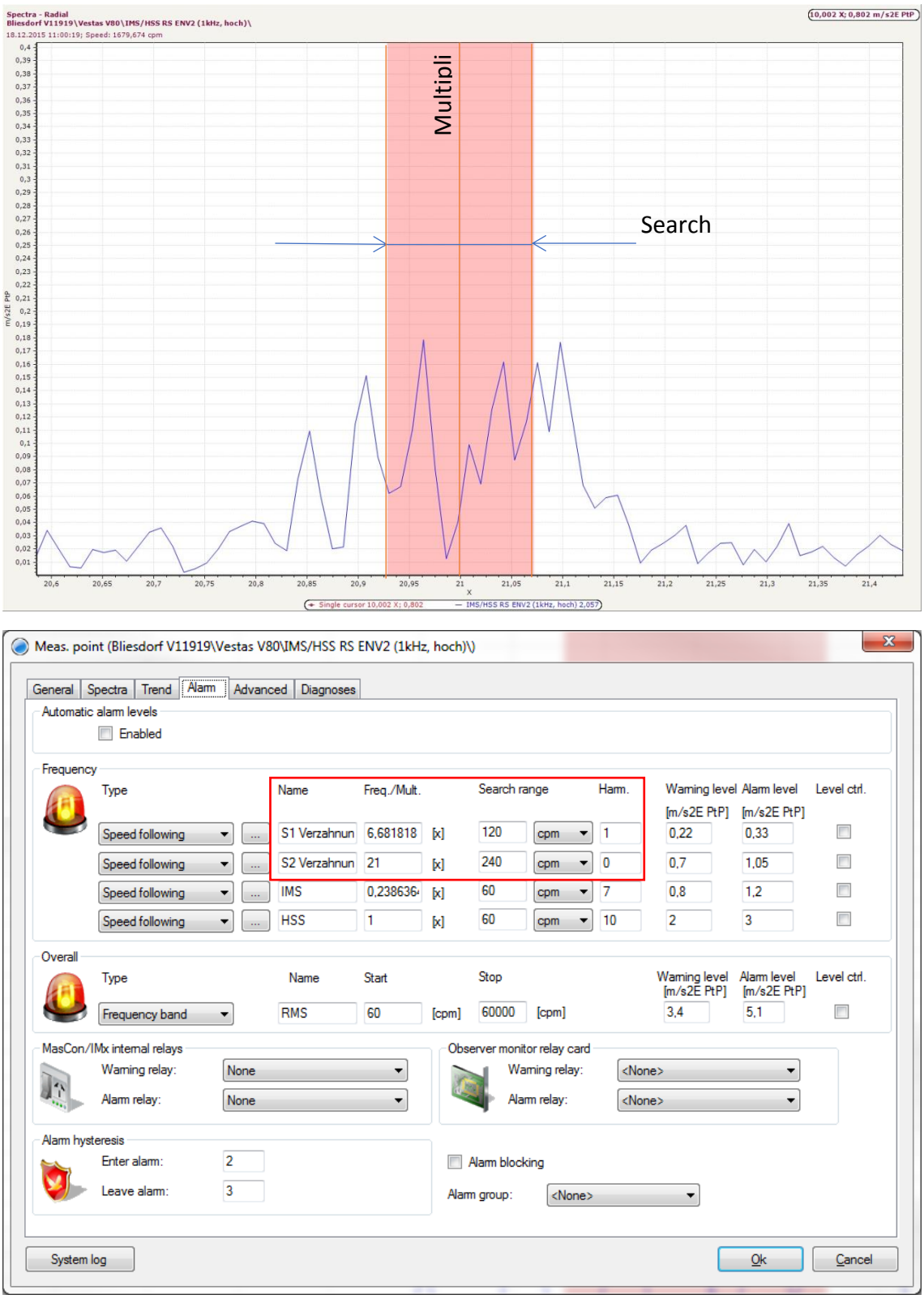

Abbildung 5-13: Konfiguration des Frequenzsuchbereichs im @ptitude Observer

US 10-1: CMI öffnet die Kinematischen Daten in der Stammdatenbank geht auf Frequenzband über Lagerfrequenzen berechnen. Es öffnet sich das Berechnungstool in dem CMI die relevanten Lager ankreuzt und auf berechnen klickt. Als Ergebnis werden der Multiplikator des Bandes (mittlere Frequenz aus den höchsten und

niedrigsten Frequenzen der Lagerkomponenten) und Spannbreite des Suchbereichs (*Search range*) ausgegeben, die CMI Messpunktkonfiguration überträgt.

CMI öffnet die Kinematischen Daten in der Stammdatenbank geht auf Frequenzband über Drehfrequenzen berechnen. Es öffnet sich das Berechnungstool in dem CMI die relevanten Getriebewellen oder Drehfrequenzen ankreuzt und auf berechnen klickt. Als Ergebnis werden der Multiplikator des Bandes (mittlere Frequenz aus den höchsten und niedrigsten Frequenzen) und Spannbreite des Suchbereichs (*Search range*) angezeigt, die CMI Messpunktkonfiguration überträgt.

5.2 User Stories Stammdatenbank

US 1 Stammdaten der WEA

Ein CMI möchte die Stammdaten der WEA in einer Datenbank speichern, um ein einheitliches Datenmanagement zu pflegen und die Verfügbarkeit der Fehlerfreien Daten im Alltagsgeschäft zu garantieren.

Hinweis: Zu den Stammdaten der WEA werden folgende Datengruppen gezählt:

- Grundcharakteristik der WEA – Maschinendaten, Sicherheit, Erstkontakt, Standort, Vertragspartner, Technische Betriebsführung, Berichtsberechtigte Personen
- Technische und kinematische Daten der Antriebstrangkomponenten – Lagerung des Antriebstrangs, Hauptlager I und II, Getriebe, Generatoren

US 1-1: MC bekommt einen neuen WP in die Überwachung. CMI öffnet die neue Software und geht auf „WEA im System Registrieren“. Ein neues Fenster öffnet sich. CMI gibt Daten für eine WEA ein und speichert diese ab.

US 1-2: Der neue Windpark besteht aus neun WEA vom gleichen Typ. Um die mehrfache Eingabe der gleichen Daten bei jeder WEA zu vermeiden, möchte ein CMI die Möglichkeit haben zu entscheiden, welche Daten aus der vorhergehender Eingabe in die neue Eingabe übernommen werden sollen.

US 1-3: Die wesentlichen Attribute der Stammdaten werden dem CMI von der Software vorgegeben, z.B. durch die Kombinationsfelder. Somit kann CMI alle Daten nach einem einheitlichen Muster in die DB einpflegen.

US 1-4: CMI bekommt zu einem späteren Zeitpunkt die kinematischen Daten des Getriebes, Hauptlagers und Generators. CMI öffnet die neue Software und sucht die bereits registrierte WEA mit der Seriennummer in der Datenbank um die fehlenden Daten ergänzen zu können.

US 1-5: CMI nimmt die kinematischen Daten in die DB auf. Der vorliegende Getriebetyp wird von der Software nicht vorgegeben. Da der CMI über entsprechende Rechte verfügt, geht er z.B. Charakteristik/Stammdaten/Getriebetyp

und trägt den neuen Getriebetyp ein. Ab sofort kann der neue Getriebetyp im entsprechenden Kombinationsfeld bei der Eingabe der kinematischen Daten gewählt werden.

US 1-6: CMI möchte kinematische Daten des Getriebes eingeben. Dafür muss dieses entsprechend der Anzahl der Planeten-, Stirnrad- und Differentialstufen nachgebildet werden.

US 1-7: Für jede Stufe wird die Anzahl der Zähne eingetragen und die Übersetzung der einzelnen Stufen und des ganzen Getriebes automatisch berechnet. Weiterhin werden die Lager (Lager + Typ) mit den Frequenzen für Innenring, Außenring, Wälzkörper, Käfig für jede Getriebestufe eingetragen.

US 1-8: Weiterhin möchte der CMI die kinematischen Daten der Hauptlager und der Generatoren einpflegen. Die Lager (Lager + Typ) werden mit den Frequenzen für Innenring, Außenring, Wälzkörper, Käfig eingegeben.

6 Daten von CMC

6.1 Existierende Datenhaltung von CMC

CMC speicherte ihre Daten zu Beginn des Projektes an mehreren Orten redundant ab. Zum einen auf einem Server im Windowsexplorer zum anderen auf einer SQL Datenbank, die mit Observer verbunden ist. Die Dateien im Explorer sind zum Großteil Excel Tabellen, Word Dokumente, PDF Dokumente und Fotos.

Mit einer stetig steigenden Menge an Daten stand CMC zunehmend vor verschiedenen Problemen:

1. Die Dateien waren nicht von einer Zentralenstelle durchsuchbar.
2. Viele Daten lagen mehrfach in verschiedenen Dateien vor.
3. Die Dateien konnten nicht gleichzeitig von mehreren Anwendern gleichzeitig geöffnet und bearbeitete werden.

6.2 Entwicklung der Stammdatenbank

Die Stammdatenbank soll alle Stammdaten der Windenergieanlagen enthalten. Dabei ersetzt sie mehrere Excel Tabellen, die im Laufe der Zeit entstanden sind. Ein besonderer Fokus liegt dabei auf der Durchsuchbarkeit und einer einfachen Handhabung. Zunächst wurde ein Prototyp in Excel entwickelt. Dieser Prototyp basiert auf der Schadensliste. Die Datenbank soll die in den User Stories beschriebenen Funktionen besitzen. Von der Funktionalität konnte er bereits einige Anforderungen erfüllen, jedoch andere scheiterten an der Limitierung von Excel. Die Tabelle

Anforderung	**Erfüllt**
Dursuchbarkeit	✓
Einfache Änderbar	✓
Diagramme	✓
Integration von Schadens- und Störungsliste	×
Vermeidung von Redundanz	×
Mehrere Gleichzeitige Benutzer	×

Tabelle 6-1: Anforderungen an Datenhaltung

zeigt welche Anforderungen von Excel unterstützt wurden und welche nicht.

Die fehlenden Funktionen konnten nicht mehr in Excel integriert werden. Darum wurde entschlossen, eine Datenbank zu verwenden. Um die Datenbank später für CMC zur Verfügung zu stellen, wurden zunächst bestehende Technologien im Unternehmen betrachtet. Aus den Gegebenheiten ließen sich weitere Anforderungen abbilden.

1. CMC arbeitet mit Windows und möchte weiter in der Umgebung arbeiten
2. Die IT Abteilung von CMC kennt sich nur mit SQL aus

Damit war die Suche nach einer passenden Datenbank schnell beendet. Es standen nur noch SQL Datenbanken im direkten Vergleich, weshalb sich für Microsofts SQL Server entschieden wurde, da dieser nativ Unterstützt wird.

Zusätzlich zur Datenbank wurde ein passendes *Frontend* entwickelt. Es wurde in C# zusammen mit ASP.Net, Razor und LINQ geschrieben. Die Technologien wurden von Microsoft für den betrieb auf Windows entwickelt und arbeiten auch nativ mit SQL Server zusammen. Deshalb konnte Entwicklungszeit gegenüber anderen Technologien eingespart werden. Das Frontend ist eine Webseite, auf der sich die Mitarbeiter im Intranet anmelden können. Über dieses haben sie Zugriff auf die Datenbank und können gleichzeitig an den verschiedenen Einträgen arbeiten.

7 IT - Architektur

In diesem Kapitel wird erklärt, warum sich für eine verteilte Architektur entschieden wurde. Speziell wird auf die Vor- und Nachteile von Hadoop als Plattform eingegangen.

7.1 Auswahl einer passenden Architektur

Für die Architektur gab es verschiedene passende Technologien zur Auswahl. Die folgende Tabelle zeigt eine Auswahl von verschiedenen SQL Datenbanken, NoSQL Datenbanken und andere Datenverwaltungsframeworks, die im Laufe des Projektes betrachtet wurden. Dabei wurde sich bei dieser Tabelle nur auf eine Auswahl von bestimmten Merkmalen beschränkt, um einen Grad an Übersichtlichkeit zu behalten.

Name	**Plattform**	**Lizenzkosten**	**Datenmodell**
Hadoop	Linux	Keine, Support kosten	Dateisystem
HBase	Hadoop	Keine	Spalten
Cassandra	Linux, Windows	Keine	Spalten
Couch DB	Linux, Windows	Keine, Support kosten	Dokument
Mongo DB	Linux, Windows	Keine, Support kosten	Dokument
Elasticsearch	Linux, Windows		Dokument
Splunk	Linux	Skalieren mit der Datenmenge	Dokument
„Normales“ SQL	Linux, Windows	Skalieren mit der Datenmenge / Keine	Zeilen

Tabelle 7-1: Mögliche Datenbank Systeme

Für das Projekt wurde ein Schwerpunkt auf *Big Data* Technologien gelegt, weshalb neben traditionellen SQL-Datenbanken auch moderne Konzepte wie NoSQL-Datenbanken zum Einsatz kommen sollten. Diese Entscheidung wurde getroffen, weil SQL-Systeme im Verhältnis zu NoSQL-Systemen nicht so gut horizontal skalieren. Die Skalierbarkeit wird besonders durch das CAP Theorem und das ACID-Transaktionsmodell limitiert. Zusätzlich sind die neueren Technologien auch in den Use Cases festgehalten.

Des Weiteren wurden möglichst kostengünstige Lösungen bzw. Lösungen ohne Lizenzkosten in Betracht gezogen. Durch diese Einschränkung verringerte sich der Pool von möglichen Frameworks weiter, wobei es auch für die meisten verbleibenden Frameworks kostenpflichtige Optionen gibt. Zum Beispiel bieten Hortonworks und Cloudera einen kostenpflichtigen Support Service für Hadoop an. Ein weiteres Beispiel ist MongoDB, die eine Enterprise Version mit zusätzlichen Funktionen im Angebot haben.

Die letzten Kandidaten wurden an ihren Datenanalysemethoden sortiert. Übrig geblieben ist Hadoop mit seinem ausgereiften Ökosystem. Die einzigen Frameworks, die eine Datenanalyse mit *Machine Learning* Verfahren oder Neuralen Netzwerken

ermöglich(t)en, sind Hadoop und Spark. Spark gehört auch zum Hadoop Ökosystem und bietet eine deutlich schnellere Engine zur Verarbeitung von Daten. Abbildung _ zeigt einen kleinen Ausschnitt aus dem momentanen Ökosystem mit einigen der nennenswerten Frameworks.

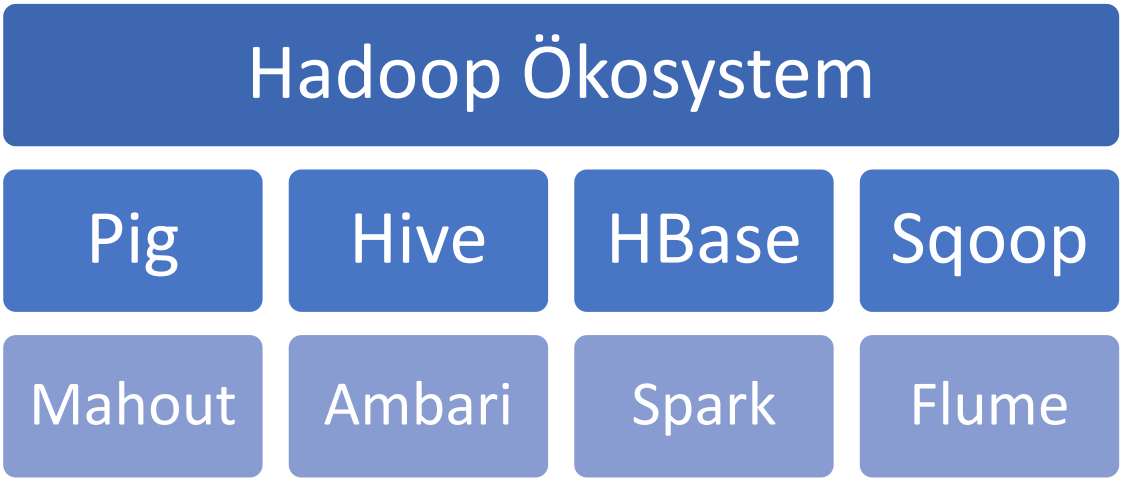

Abbildung 7-1: Auszug Hadoop Okösystem

An dieser Stelle ist noch zu erwähnen, dass Hadoop inzwischen mit allen aufgelisteten Frameworks und Datenbanken zusammenarbeiten kann. Wenn sich im weiteren Bericht auf das „Hadoop Ökosystem“ bezogen wird, werden nur direkt verwandte Frameworks gemeint.

7.2 Vorstellung vom Hadoop Ökosystem

7.2.1 Hadoop

Hadoop in seiner Grundform implementiert ein Paper von Google. In diesem Paper beschreibt Google, wie sie auf mehreren Servern gleichzeitig an dem gleichen Datensatz arbeiten können. Dabei wird Hadoop als *Open Source* Projekt entwickelt und es kann auf „normaler“ Hardware, anstatt teurer Serverhardware, ausgeführt werden. Neben den geringen Kosten, überzeugte Hadoop durch die große Entwicklercommunity. Unter dieser finden unteranderem Firmen wie Facebook, Microsoft oder IBM wieder. Diese Zusammenarbeit findet unter der Apache Lizenz statt. Zusätzlich werden auch viele der anderen Frameworks im Hadoop Ökosystem unter dieser Lizenz entwickelt, wodurch diese auch ohne Lizenzkosten angeboten werden. Darüber hinaus deckt das umfangreiche Ökosystem bereits viele Anwendungsfälle ab, wodurch Entwicklungszeit eingespart wird.

Konkret implementiert Hadoop drei Hauptkomponenten. Diese teilen sich in den *MapReduce* Algorithmus, das *Hadoop Distributed File System* und YARN auf. Der *MapReduce* Algorithmus liefert die Grundlage für das verteilte Arbeiten auf einem Server Cluster und ist damit ein fundamentaler Bestandteil von Hadoop. Er teilt sich folgendermaßen auf (vgl. Abbildung 7-2):

- Split: Aufteilung der Daten (Hier schon durch das HDFS sichergestellt)
- Map: Übergabe einer Aufgabe / Algorithmus an verschiedene Server, die die Aufgabe auf den gesamten Daten ausführen
- Shuffle: Zuordnung der einzelnen Zwischenergebnisse
- Reduce: Zusammenführung des Ergebnisses

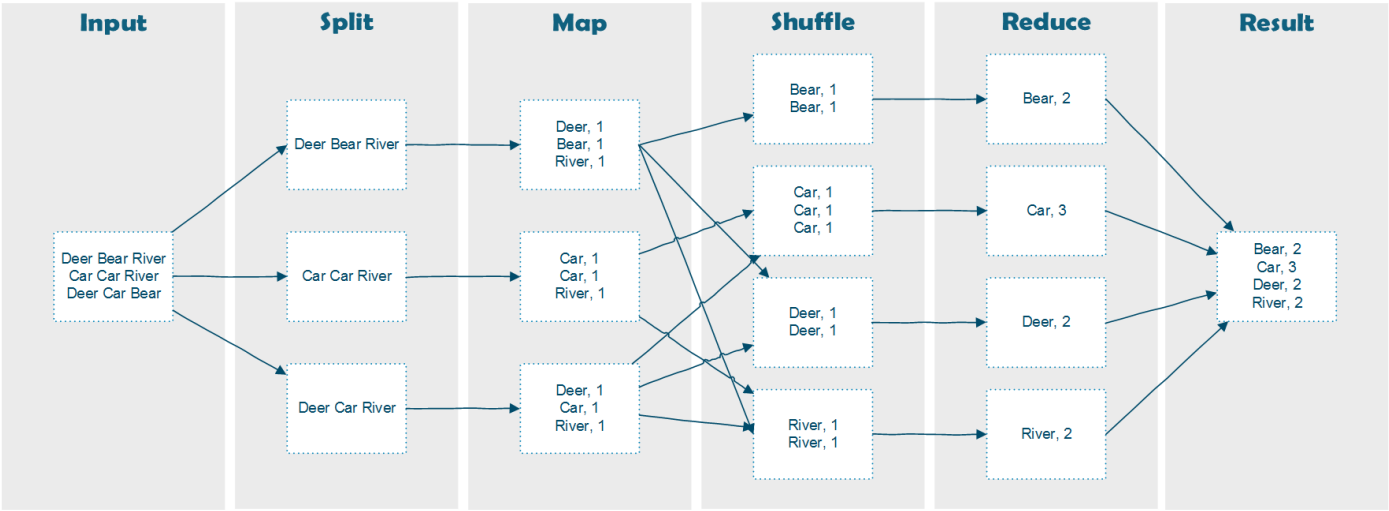

Abbildung 7-2: MapReduce Algorithmus

Darüber hinaus verwendet Hadoop seine eigene Struktur zur Datenspeicherung, das Hadoop Distributed File System oder kurz HDFS. Durch das HDFS können Daten auf einem verteilten System redundant abgelegt werden. Es bildet die Grundlage für die anderen Frameworks schnell auf die verfügbaren Daten zu zugreifen. So werden die Server herausgesucht für die Datenverarbeitung auf denen sich bereits die Datenbefinden. Dadurch werden Netzwerkkosten gespart und die Skalierbarkeit sichergestellt.

YARN stellt den anderen Frameworks und Hadoop selbst eine Art Betriebssystem zur Verfügung, welches Ressourcen verteilt und die Reinfolge von Jobs festlegt. Erst in Version 2.0 wurde YARN eingeführt. Seit der Einführung sind zahlreiche andere Frameworks entstanden, die über den MapReduce hinausgehen. Im Bericht werden beispielhaft Spark und Flume erwähnt.

7.2.2 Ambari

Um die Verwaltungs- und Installationszeit so gering wie möglich zu halten, wurde Ambari verwendet. Ambari ist ein *Cluster Management System* mit einem Schwerpunkt auf dem Hadoop Ökosystem. Es vereinfacht viele Schritte während der Installation, bietet einen zentralen Zugang zur Verwaltung und integriert viele Frameworks des Ökosystems. Zum Beispiel gewährt es Einblicke in die verfügbaren Ressourcen, wie zum Beispiel:

- verfügbaren Arbeitsspeicher
- Prozessorauslastung
- die Lebensdauer der einzelnen Softwarekomponenten

Das Framework wird hauptsächlich von Hortonworks entwickelt und kostenlos zur Verfügung gestellt. Falls Hilfe bei der Einrichtung benötigt wird, bietet Hortonworks kostenpflichtige Supportleistungen an.

7.2.3 Hive

Hive wurde Als SQL Dialekt für Hadoop entwickelt. Damit liefert Hive eine Schnittstelle zwischen bekanntem SQL und MapReduce auf einem Hadoop Cluster. Zur Speicherung der Daten werden codierte Dateien auf dem HDFS gespeichert. Der Zugriff erfolgt über den SQL Dialekt HiveQL, dieser wird vom System in einen MapReduce Job umgewandelt. Dadurch kann Hive, im Gegensatz zu anderen SQL Datenbanken, horizontal Skalieren. In den neueren Versionen kann auch *ACID* als Transaktionseigenschaft umgesetzt werden. Die Idee von Hive ist es auch Mitarbeitern, die bisher nur mit SQL gearbeitet haben, Hadoop als neues Framework zu erschließen.

Ein weiterer Vorteil von Hive ist es, dass es einfach ist Daten von andren SQL Datenbanken zu importieren. Zu diesem Zweck wird Sqoop eingesetzt. Das Vorgehen wird zu einem späteren Zeitpunkt genauer beschrieben. Darüber hinaus müssen Anwender, die zuvor in einer SQL-Umgebung gearbeitet haben, keine neue Programmiersprache lernen.

7.2.4 HBase

Die NoSQL Datenbank HBase basiert auf Googles *Big Table* und wird den Spaltenorientierten Datenbanken zugeordnet. Dabei verwendet HBase das HDFS in Kombination mit *Zookeeper* einem anderen Framework im Hadoop Ökosystem, um die Daten zu speichern und zu verwalten. Durch diese Abhängigkeit verfügt HBase die gleiche Skalierbarkeit, die vom Hadoop Cluster zur Verfügung gestellt wird.

Deswegen ist HBase im Gegensatz zu SQL Datenbanken darauf ausgelegt, große Mengen an Daten zu speichern. Am besten performt HBase bei Schreib Operationen und Lese Operationen, die eine oder mehrere ganze Spalten gleichzeitig betreffen. Diese Vorteile eignen sich sehr gut für unseren Anwendungsfall, da Sensordaten immer nur neu gespeichert werden und für die Analyse mit *Machine Learning* Algorithmen ganze Spalten verwendet werden.

Darüber hinaus ist HBase Datentyp unabhängig, somit können verschiedenste Daten gespeichert werden. Diese Eigenschaft ist sehr nützlich, falls neben den Sensordaten später auch Texte, Bilder oder Videodateien in der Datenbank gespeichert werden sollen.

7.2.5 Spark

Der bisher vorgestellte MapReduce Algorithmus ist inzwischen nicht mehr der schnellste Algorithmus, um auf einem verteilten System Daten zu verarbeiten. Spark hat eine solche schnellere Implementierung des Algorithmus. Vom Konzept her führt Spark immer noch einen MapReduce aus, wobei dieser sehr Optimiert ist. Zu den Verbesserungen gehören eine bessere Speicherverwaltung, bei der der Großteil der Daten im Arbeitsspeicher gehalten wird, sowie eine Optimierungs-Engine, die den Code als *directed acyclic graph*, kurz DAG, aufbaut. Der Graph entfernt unnötige Operationen und beachtet wo sich die Daten auf dem Cluster befinden, sodass insgesamt ein hundertfacher Geschwindigkeitsgewinn zu verzeichnen ist.

Zusätzlich kann Spark auch mit Datenströmen umgehen. Somit werden Echtzeitanwendungen auch für große Datenmengen ermöglicht. Diese Ströme können aus verschiedensten Quellen stammen, unteranderem aus:

- Kafka
- Dem HDFS
- Selbstgeschriebene TCP Ströme

Der letzte Vorteil von Spark gegenüber Hadoop ist, dass es viele Interfaces bietet (vgl. Abbildung 7-3). So kann mit SQL von Spark aus auf die Daten zu gegriffen werden. Auch sind bereits viele *Machine Learning* Algorithmen mit Spark implementiert worden. Dadurch kann an dieser Stelle mehr Entwicklungszeit gespart werden. Nur im Bereich *Deep Learning* hat Spark noch einige Schwächen, wobei Hadoop oder andere verteilte Frameworks dafür im Moment auch keine Lösung anbieten. Die

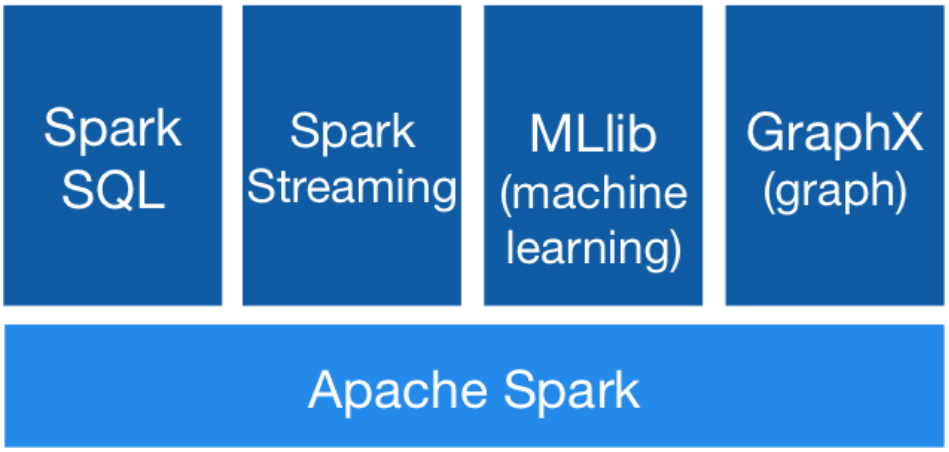

Abbildung 7-3: Spark APIs

letzte API ist GraphX mit deren Hilfe Daten als Graph aufbereitet werden können. Klassische Anwendungsfälle sind der Aufbau von Computernetzwerken oder Social Media Anwendungen.

8 Vorbereitung der Daten

8.1 Migration der Daten von MS SQL zu Hadoop

Nach den Workshops, um die Daten und die Arbeitsweise von CMC zu verstehen, mussten die Daten in das Big Data System der Fachhochschule migriert werden. Die genaue IT-Architektur von dem System wird im Kapitel 7. IT - Architektur besprochen. Für die Migration wurden zwei Möglichkeiten in Betracht gezogen. Zum einen ein *Datastream*, der permanent neue Daten an die Server schickt, zum anderen eine einmalige Migration der existierenden Daten. Durch technische und zeitliche Limitierungen sowie Sicherheitsbedenken wurde sich für die einmalige Migration entschieden.

Die benötigten Daten werden bei CMC auf einer MS SQL Datenbank gespeichert. Für die einmalige Migration wurden wieder verschiedene Optionen betrachtet. Im Zusammenhang mit Hadoop und anderen NoSQL Datenbanken gibt es keinen „*One size fits all*“ Ansatz, um SQL Daten von einem existierenden System zu einem NoSQL System zu schicken. Nach der Evaluierung von den verfügbaren Technologien wurde sich für eine Kombination aus Sqoop und Hive entschieden.

Hive ist ein SQL Interface für Hadoop. Mit Hive wird es ermöglicht SQL Daten direkt auf Hadoop zu verwenden, ohne diese vorher zu verändern. Gleichzeitig profitiert es von der Skalierbarkeit von Hadoop und einer umfangreichen Integration in das Hadoop Ökosystem. Um die Leistung von Hadoop zu verwenden, greift Hive für die Ausführung von SQL Anfragen auf MapReduce zurück.

Zur eigentlichen Migration von MS SQL nach Hive bzw. Hadoop wurde Sqoop verwendet. Sqoop nimmt SQL Daten entgegen und importiert sie auf den Hadoop Cluster (vgl. Abbildung 8-1). Für diesen Vorgang verwendet Sqoop ebenfalls MapReduce. Bei diesem Vorgang wird Sqoop die SQL Datenbank sowie der Hadoop Cluster übergeben. Sofern keine komplexen Objekte in der Datenbank vorliegen, kann Sqoop die Daten ohne weiteres migrieren.

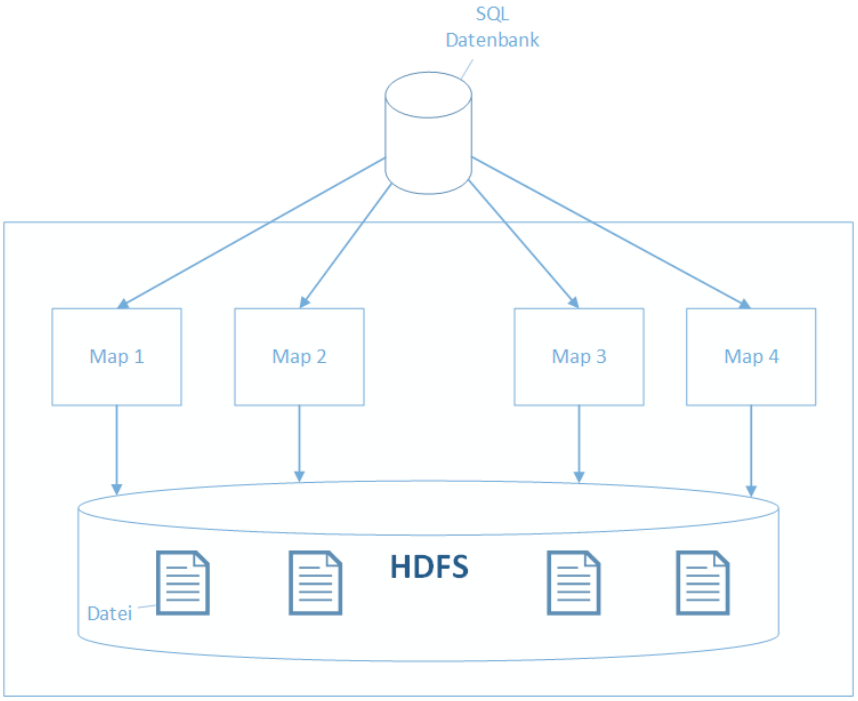

Abbildung 8-1: SQL Daten mit Sqoop in Hive importieren

Die Daten mit Hive verwaltet werden, können auch mit den anderen Frameworks aus dem Hadoop Ökosystem geladen und verarbeitet werden. Das Standardformat für Hive zur Datenverwaltung ist Parquet, welches inzwischen ein fester Bestandteil des Hadoop Ökosystems geworden ist. Es zeichnet sich durch seine gute Komprimierung und einfache Handhabung aus.

Wichtig für dieses Projekt ist die Verfügbarkeit der Daten in Spark. Denn mit Hilfe von Spark wurde der Großteil der Analysen auf dem *Cluster* durchgeführt. Auch Spark verfügt über eine solide API für Parquet. Dadurch mussten keine weiteren Änderungen an der Datenspeicherung vorgenommen werden.

9 Datenbereinigung

9.1 Für Artificial Neural Network (ANN) als prognostisches Modell

Bevor die Daten für das Training der ANN-Modelle verwendet werden, wird die folgende Datenverarbeitungspipeline durchgeführt:

9.1.1 Fehlende Daten bereinigen

Wie bereits erwähnt, wenn ein Verständnis der Daten erklärt wird, insbesondere die Art und Weise, wie Daten durch das *Condition Monitoring System* erfasst werden, Daten in unregelmäßigen Abständen aufgrund von Netzwerkproblemen oder Fehlern in den Sensoren gesammelt. Daher kann es vorkommen, dass der resultierende Datensatz fehlende Daten enthält, die in unregelmäßigen Zeitabständen erscheinen.

Um fehlende Werte in Zeitreihen zu adressieren, wurden verschiedene Ansätze entwickelt. Dies kann eine einfache Lösung sein, um die fehlenden Daten auszuschließen und die Analyse nur auf den vorhandenen Daten durchzuführen. Andere Strategien zielen darauf ab, die fehlenden Werte auszufüllen, wie z. B. die Durchführung von Re-Probenahme, um eine gleichmäßige Verteilung der Daten über den Beobachtungszeitraum zu gewährleisten, Glättung oder Interpolation, Spektralanalyse, Kernelmethoden, multiple Imputation und Erwartungsmaximierungsalgorithmus. Gleichzeitig gibt es aber auch intensive Forschungsarbeiten zu ANNs, wie z. B. das Training des Netzwerks, wie man mit fehlenden Werten umgeht.

Eine regelmäßige Abtastung der Zeitseriendaten wird durchgeführt, um eine Eingabe mit fester Länge zu erhalten, um die Modelle des ANN zu trainieren. Es wurde eine Abtastrate von 24 Stunden unter Verwendung eines Mittelwerts gewählt. Eine höhere Auflösung bietet zwar mehr Datenpunkte, kann aber auch unnötige Geräusche im Modell verursachen. Auf der Grundlage von Vorversuchen wurde festgestellt, dass diese 24-Stunden-Auflösung vernünftige Ergebnisse liefert. Darüber hinaus ergibt sich eine ähnliche Zeitauflösung, die von den Ingenieuren in der *Condition Monitoring Software* verwendet wird, um Trends zu beobachten, bei denen die Amplitudenschwankung über Tage und nicht innerhalb eines Tages betrachtet wird. Beachten Sie in diesem Zusammenhang, dass jeder Datenpunkt als ein Zeitschritt (entspricht einer Einheit von einem Tag) für die Auswertung des Prognosehorizonts behandelt wird. Gleichzeitig wurde eine lineare Interpolation auf Basis der gleichen Zeitauflösung durchgeführt, um die Lücken zu schließen. Obwohl es neben einer einfachen linearen Interpolation mehrere Ansätze gibt, hängt dies nach wie vor stark vom zugrundeliegenden Prozess ab, in dem die Daten gewonnen werden. Leider wurden die Daten, wie bereits erwähnt, bereits vom Monitoring-System verarbeitet und der Verarbeitungsalgorithmus ist unbekannt. In diesem Fall

wurde die lineare Interpolation gewählt, um die Einfachheit der Interpolation zu gewährleisten und um der Darstellung der Daten im Monitoringsystem zu entsprechen. Dennoch wurde bei diesem Ansatz eine angemessene Performance erreicht.

Die folgenden Abbildungen (vgl. Abbildung 9-1, Abbildung 9-2, Abbildung 9-3) stellen die Unterschiede zwischen den ursprünglichen Zeitreihen und den resultierenden

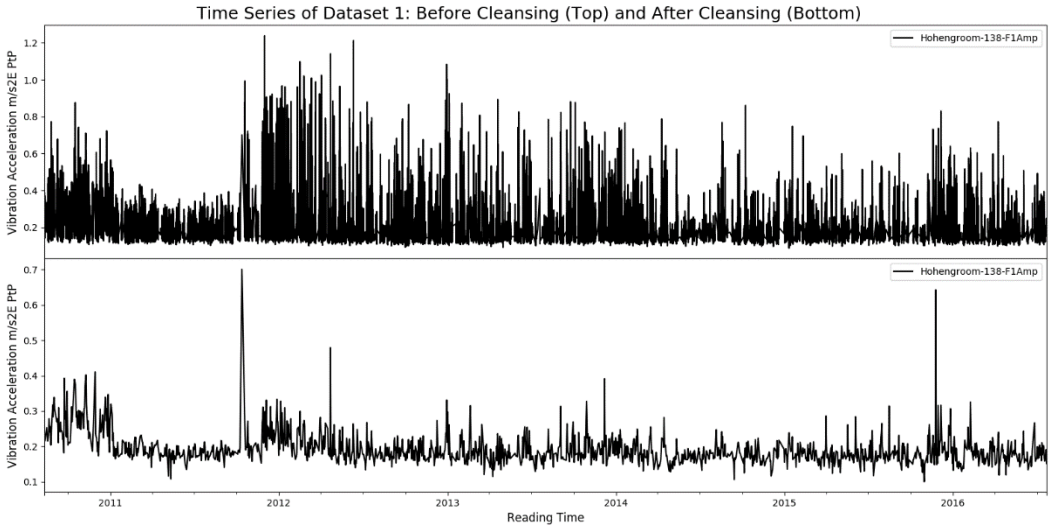

Abbildung 9-1: Datensatz 1 vor und nach der Bereinigung

Zeitreihen nach Durchführung der Datenbereinigung dar.

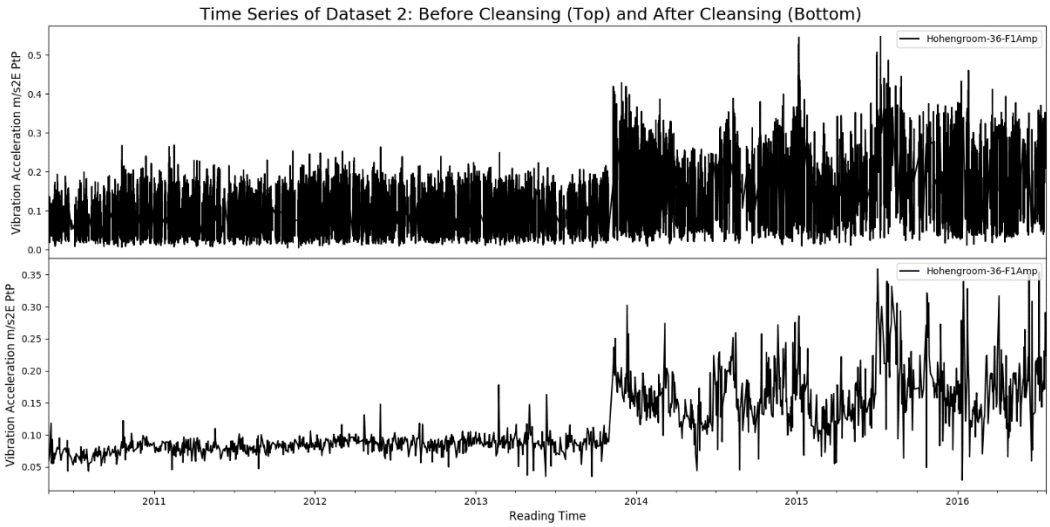

Abbildung 9-2: Datensatz 2 vor und nach der Bereinigung

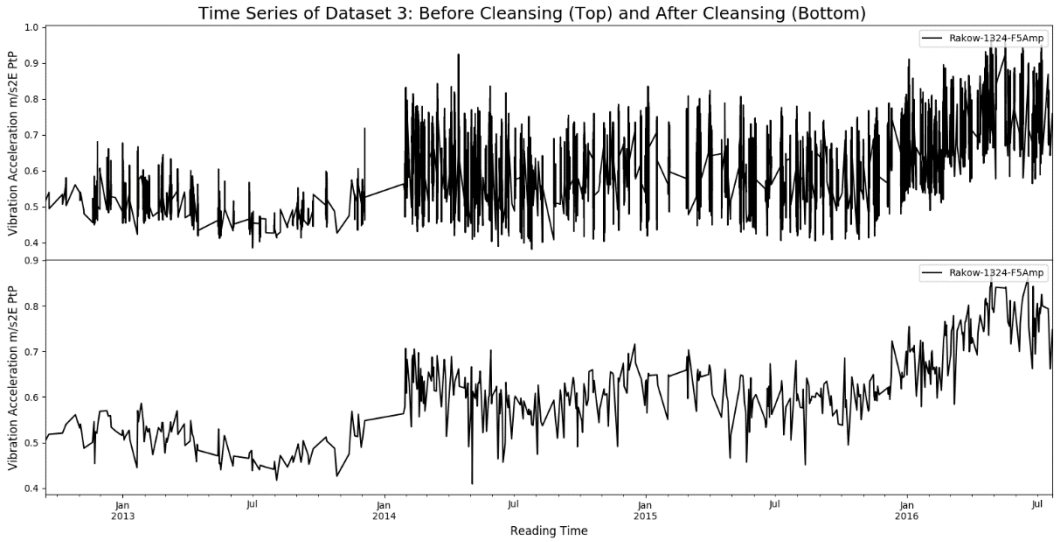

Abbildung 9-3: Datensatz 3 vor und nach der Bereinigung

9.1.2 Geteilte Daten für Training und Testen

Da die Modelle mit Hilfe von beaufsichtigten Lernprozessen trainiert werden, sind Trainingsdaten erforderlich, um den Modellen die Grundwahrheit zu vermitteln, gegen die sie lernen können. Dies bedeutete, dass es notwendig ist, den Datensatz zu splitten, so dass ein Teil des Datensatzes für das Training des Modells und der andere Teil für das Testen des trainierten Modells verwendet werden kann. Es gibt mehrere Überlegungen bei der Entscheidung über die optimale Aufteilung und dies ist in der Regel als ein statistisches Problem Sampling behandelt werden.

Hier kommt die Convenience-Sampling-Methode zum Einsatz, bei der die früheren Datenpunkte für das Training zugewiesen werden, während die letzten 360 Datenpunkte für das Testen genutzt werden. Dies entspricht einem Zeitfenster von zwölf Monaten. Auf Basis dieses Fensters werden dann die Vorhersagekräfte der trainierten Modelle ausgewertet. Gleichzeitig werden die Daten in ein Tupel aus Eingangsreihenfolge (historische Werte bis Zeitschritt *t*) und Ausgangsreihenfolge (Werte beginnend ab Zeitschritt *t+1* bis zur Länge des Prognosehorizonts) umformatiert. Im Rahmen der experimentellen Befunde werden unterschiedliche Sequenzlängen ausgewertet.

9.1.3 Trainingsdaten normalisieren

Dieser Schritt wird aus praktischen Gründen beim Training von ANNs durchgeführt. Die Normalisierung der Daten zwischen einem definierten Bereich (abhängig von der Aktivierungsfunktion oder den Daten) hat sich als vorteilhaft erwiesen, um das Training von ANNs schneller zu machen und die Wahrscheinlichkeit zu verringern, dass sie in einem lokalen Optimum stecken bleiben. Es gibt verschiedene Techniken wie Minimum-Maximum-Normierung, Dezimalskalierung, Normalisierung des *Z-Scores*, Median-Normierung und Sigmoid-Normierung mit verschiedenen Vor- und

Nachteilen. Bei dieser Implementierung werden die Daten mit Minimum-Maximum normiert, da die Vorversuche eine Eignung ergeben haben. Zu beachten ist auch, dass die Normalisierung nicht mit dem gesamten Datensatz, sondern mit den statistischen Informationen durchgeführt wird, die ausschließlich auf den Merkmalen der Trainingsdaten basieren. Damit soll verhindert werden, dass Informationen aus den Testdaten in das Modell gelangen. Der gleiche Skalierungsfaktor wird dann während der Inferenz auf die Testdaten angewendet.

10 Deep Learning Frameworks

Dieses Kapitel erklärt, warum ein *Deep Learning Framework* gewählt wurde, um die Entwicklung zu beschleunigen. Die Eignung verschiedener Frameworks wird vorgestellt. Insbesondere werden die Vor- und Nachteile von Google TensorFlow, das schließlich als primäres Framework ausgewählt wurde, diskutiert.

Diese Frameworks können als Toolkit betrachtet werden, das folgende Vorteile bietet:

1. Verfügbarkeit von reichhaltigen Deep-Learning-Bibliotheken statt der neuen Entwicklung der Module
2. Einfache Modellierung und Schulung eines komplexen *Artificial Neural Network*
3. Möglichkeit, Berechnungen auf einzelnen oder mehreren GPUs statt nur auf CPUs durchzuführen, und verbessert so die Leistung
4. Starke Unterstützung durch die *Open-Source-Communities*
5. Möglichkeit, vorab trainierte Modelle für Transfer-Lernen zu verwenden

10.1 Der Vergleich der möglichen Deep Learning Frameworks

Eine Übersicht ist in der folgenden Tabelle zu finden.

Platform	TensorFlow	BigDL	Deeplearning4j	CNTK	MXNet	H2O
Release Year	2016	2017	2015	2016	2015	2014
Core	C++	Scala	C++	C++	C++	Java
API	C++, Python	Scala, Python	Java, Scala	Python	C++, Python etc.	Java, Python etc.
Synchronization Model	Sync or async	Sync	Sync	Sync	Sync or async	Async
Communication Model	Parameter server	P2P	Iterative MapReduce	MPI	Parameter server	Distributed fork-join
Multi-GPU	Yes	Yes	Yes	Yes	Yes	Yes
Multi-Node	Yes	Yes	Yes	Yes	Yes	Yes
Data Parallelism	Yes	Yes	Yes	Yes	Yes	Yes
Model Parallelism	Yes	No	No	No	Yes	No
Fault Tolerance	Check-point	Check-point	Check-point	Check-point	Check-point	No

Tabelle 10-1: Übersicht Deep Learning Frameworks

10.2 Vorstellung von Google TensorFlow

TensorFlow ist eine Open-Source-Softwarebibliothek von Google, die Datenflussgraphen für numerische Berechnungen verwendet. Während es für seine *Deep-Learning*-fähigkeiten beliebt ist, unterstützt es auch andere Algorithmen des *Machine Learning*, was es zu einem flexiblen Framework macht. Es verwendet ein mehrdimensionales Array primitiver Werte, den Tensor, als zentrale Dateneinheit. Jedes Mal, wenn ein TensorFlow-Programm ausgeführt wird, werden die Arbeitsabläufe in einem Knotendiagramm angeordnet, wobei jeder Knoten die Tensoren als Ein- und Ausgänge verwendet.

Diese Datenflussgrafik gibt TensorFlow die Möglichkeit, das Training von rechenintensiven *artificial neural networks* auf großen Datensätzen durch diese Techniken zu beschleunigen:

1. Data Parallel Training: Die Minimierung einer Zielfunktion wird durch Parallelisierung der Berechnung verschiedener Datenmengen über mehrere Geräte hinweg beschleunigt.
2. Modell Parallelschulung: Verschiedene Teile der Modellberechnung werden gleichzeitig auf verschiedenen Rechengeräten für den gleichen Datenstapel durchgeführt.
3. Gleichzeitige Schritte für die Modellberechnung *Pipeline*: Eine bessere Ausnutzung für das Training der ANNs wird durch *Pipeline* der Berechnung des Modells innerhalb derselben Geräte erreicht. Dies ist vergleichbar mit der asynchronen Datenparallelität innerhalb derselben Geräte.

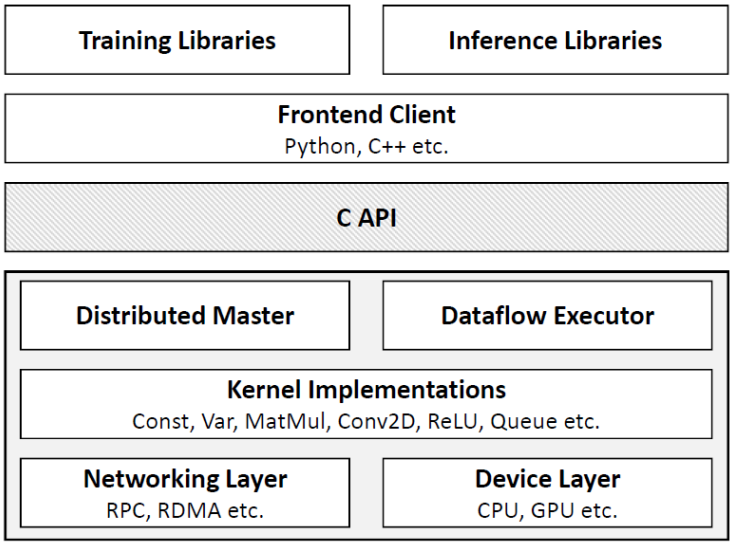

Abbildung 10-1: TensorFlow Schichten Diagramm

Die Abbildung 10-1 zeigt die allgemeine Architektur von TensorFlow. Die Kern-Laufzeit, die in C++ implementiert ist, besteht aus den Kernel-Implementierungen von *Neural Networks* mit dem verteilten Master und dem Datenfluss-Executor, der die Berechnungen mit Hilfe der Netzwerkschicht und der Geräteebene verwaltet. Diese Kern-Laufzeit wird über eine C-API bereitgestellt, die Clients wie Python und C++ unterstützt. Diese Clients stellen die Trainings- und Inferenzbibliotheken zur Verfügung, um das resultierende Modell zu entwickeln.

Ein wesentlicher Vorteil von TensorFlow ist die Möglichkeit, TensorFlow verteilt zu betreiben. Es verwendet Checkpoints, die den Zustand des zu trainierenden Modells speichern. Verteilte Cluster synchronisieren sich mit dem aktuellen *Checkpoint*, um sicherzustellen, dass alle Variablen des Modells über die Cluster hinweg synchronisiert werden. Dieses verteilte Verhalten wird in der vorgeschlagenen Implementierung durch die Kombination von TensorFlow mit Apache Spark ausgenutzt.

10.3 Distributed Deep Learning über das Hadoop-Ökosystem

Um großflächig verteiltes Tensorflow-Training und Inferenz auf Apache Spark-Cluster zu unterstützen, wurde Yahoo TensorFlowOnSpark eingesetzt. Die Architektur ist in der Abbildung 10-2 dargestellt.

Ein wichtiges Architekturdesign ist, dass es die direkte Tensor-Kommunikation zwischen den TensorFlow-Prozessen zwischen Mitarbeitern und Parameterservern unterstützt, ohne dass Spark-Treiber involviert werden müssen.

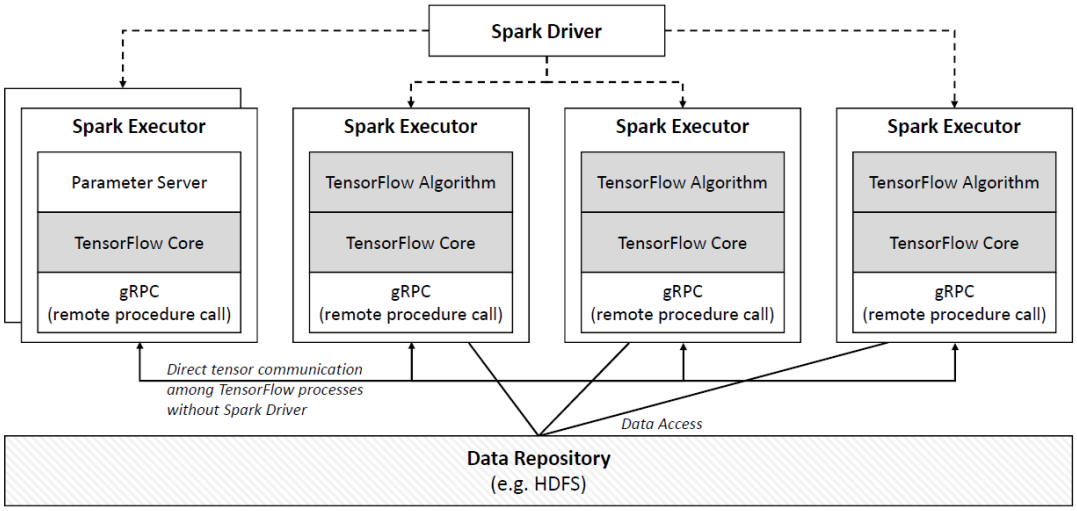

Abbildung 10-2: TensorFlowOnSpark

Es bietet folgende Vorteile:

1. Entwickelt für die Integration mit Spark SQL, MLlib und anderen Spark-Bibliotheken in einer einzigen *Pipeline* oder einem einzigen Programm
2. Unterstützt alle TensorFlow-Funktionalitäten: synchrones und asynchrones Training und Inferenz; Modell-/Datenparallelität; und TensorBoard
3. Kann mit verschiedenen Arten von Datensätzen arbeiten: auf HDFS oder aus anderen Quellen, die von Spark gepusht oder von TensorFlow gezogen werden
4. in der Lage sein, eigene Python-Distributionen mit TensorFlow und seinen Abhängigkeiten zu paketieren und an die Spark-Executoren zu liefern, anstatt die Bibliotheken direkt auf alle Knoten verteilen zu müssen, was einen Overhead verursacht hätte, um sie auf dem neuesten Stand zu halten
5. behebt die Lücke, in der die aktuelle Version des verteilten TensorFlow (Release 1.2.0) YARN als Clustermanager noch nicht unterstützt.

Obwohl es ein ähnliches Projekt von DataBricks gibt, das TensorFrames heißt und TensorFlow auf Spark DataFrames ermöglicht, ist es noch experimentell und wird nur als technische Vorschau angeboten.

10.4 Parallelisierung

Dieser Abschnitt beschreibt die tatsächliche Konfiguration der verteilten Umgebung, die für das Modelltraining eingerichtet ist und auf den zuvor erwähnten Informationen über die verteilten Fähigkeiten von TensorFlow und TensorFlowOnSpark aufbaut. Diese Konfiguration wird programmgesteuert durchgeführt.

Der Ansatz der Datenparallelität wurde gewählt und wird durch eine Kombination aus der Replikation zwischen Graphen und synchronem Training umgesetzt. Dies wird in der Abbildung 10-3 veranschaulicht. In der Zwischengraphenreplikation baut jeder

Arbeiter sein eigenes Modell (in Form eines Rechendiagramms), jedoch mit den gemeinsam genutzten Parametern, die an den Parameterserver angeheftet sind. Der Mitarbeiter trainiert sein Modell mit verschiedenen Chargen der Trainingsdaten, die aus dem *Hadoop Distributed File System* (HDFS) als *Spark Resilient Distributed Datasets* (RDDs) eingespeist werden. Nach jedem Trainingsschritt werden die aktualisierten Parameterwerte (z.B. Gradienten von Gewichten und Bias) an den Parameterserver gesendet. Beim synchronen Training wartet der Parameterserver auf alle Updates, bevor er eine Mittelwertbildung der Werte auf einmal durchführt. Diese Werte werden dann von den einzelnen Mitarbeitern abgeholt, um den nächsten Ausbildungsschritt fortzusetzen.

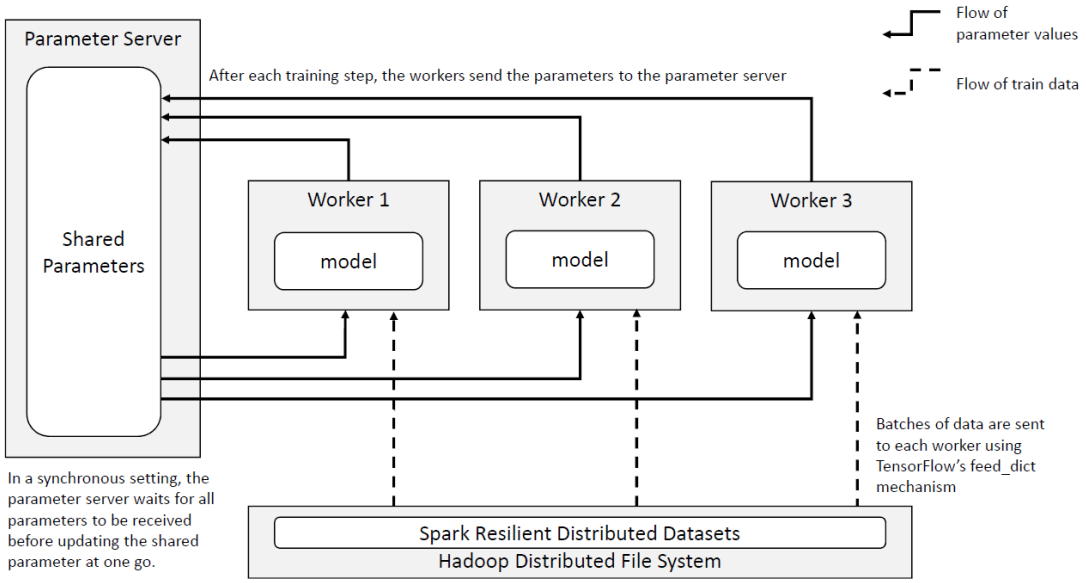

Abbildung 10-3: Datenparallelität von TensorFlow auf Spark

Diese Konfiguration bietet zwei wesentliche Vorteile:

1. Unterschiedliche Datenmengen können von den Mitarbeitern parallel verarbeitet werden. Dies kann als eine Form der Regularisierung angesehen werden.
2. Obwohl die Implementierung von synchronem Training die Gesamttrainingszeit im Vergleich zu asynchronem Training erhöhen kann, vermeidet es die Situation, dass ein langsamer Mitarbeiter veraltete Updates an den Parameterserver sendet. Dieses veraltete Update wäre an die anderen Arbeiter weitergegeben worden, was sich auf die Effizienz der Ausbildung auswirken und die Genauigkeit des Modells möglicherweise verringern würde.

Diese verteilte Trainingskonfiguration ist nicht trivial und erfordert die Zusammenarbeit von TensorFlowOnSpark und TensorFlow.

TensorFlowOnSpark ist verantwortlich für die Aufgabenverteilung und die Datenzuführung. Es nutzt YARN als Cluster-Manager für die Bereitstellung des Parameterservers und der Mitarbeiter im Cluster-Netzwerk und bietet eine direkte Tensor-Kommunikation zwischen dem Parameterserver und den Mitarbeitern. Es ermöglicht auch die Ausführung der TensorFlow-Prozesse an den Spark-Ausführern und beschleunigt so das Training.

Andererseits steuert TensorFlow die Datenparallelität, indem es sicherstellt, dass das synchrone Training über einen Synchronisationsoptimierer aufrechterhalten wird. Dieser Optimierer überwacht alle eingehenden Updates von den Mitarbeitern zum Parameterserver, sammelt die Parameter-Updates, mittelt sie und aktualisiert dann die gemeinsam genutzten Parameter.

Eine kurze Analyse mit experimentellen Ergebnissen über den Vorteil, dass mehrere Mitarbeiter gleichzeitig ein Modell trainieren können, findet sich in Abbildung 10-4. Es zeigt den Vergleich der Ergebnisse der Trainingsleistungen mit der unterschiedlichen Anzahl der Beschäftigten in einer verteilten Spark-Umgebung. Die Ergebnisse haben deutlich gezeigt, wie der Anstieg der Beschäftigtenzahl es dem Modell ermöglicht hat, eine niedrigere Fehlerquote bei gleicher Anzahl von Ausbildungsschritten zu erreichen.

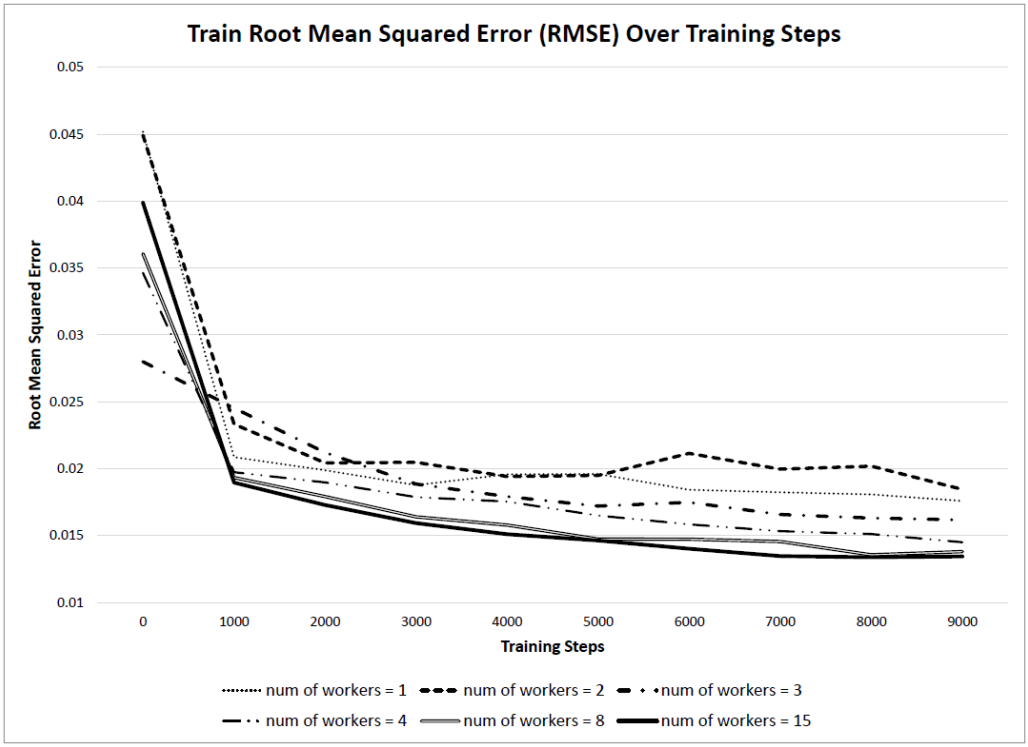

Abbildung 10-4: Trainigsgeschwindigkeit im Verhältnis zu der Anzahl der Arbeitern

11 Modellierung

11.1 Für *Artificial Neural Network* (ANN) als prognostisches Modell

Drei Arten von *Artificial Neural Networks* (ANNs) werden implementiert und anhand ihrer Effektivität beim Lernen der verschiedenen Zeitreihen miteinander verglichen.

Diese Netzwerke werden mit Google TensorFlow modelliert, wobei ein Rechendiagramm instanziiert wird, um die Netzwerktopologie und die zugehörigen mathematischen Berechnungen darzustellen.

In jedem Abschnitt wird die beste Modelldefinition vorgestellt, die anhand einer Reihe von experimentellen Bewertungen ausgewählt wurde.

11.1.1 *Feedforward Multi-Layer Perceptron* Modell

Dies wird auch als Multi-Layer-Perzeptron (MLP) bezeichnet, da es von der Idee eines Single-Layer-Perzeptrons ausgeht, vor allem um das Defizit bei der Lösung nichtlinear trennbarer Probleme zu beheben. Ein MLP definiert ein Mapping $y = f(x; \vartheta)$, bei dem es den Eingang x aufnimmt und versucht, die Parameter ϑ zu erlernen, um einen gewünschten Ausgang y zu erreichen. Die Idee der *Feedforward* kommt von der Art und Weise, wie Informationen durch das Netzwerk geleitet werden – Informationen werden zuerst von der Eingabe x ausgewertet, dann durch die Zwischenschichten verarbeitet, die zur Definition der Funktion f, und schließlich zur Ausgabe y verwendet werden. Es gibt keine Rückkopplungsanschlüsse, d.h. die Informationen an den Ausgängen werden nicht in das Netzwerk zurückgespeist.

Abbildung 11-2 zeigt ein Beispiel für einen MLP, der als gerichteter azyklischer Graph modelliert wurde. Jede Eingabe wird als Neuron dargestellt, ähnlich den Neuronen in der verborgenen Schicht. Jede Kante, die zwei Neuronen verbindet, ist mit einem Gewicht verbunden. Die Ausgabe wird ebenfalls als Neuron dargestellt. Je nach Anforderung der Aufgabe kann die Anzahl der Eingangsneuronen, die Anzahl der versteckten Schichten, die Anzahl der Neuronen in jeder versteckten Schicht und auch die Anzahl der Ausgangsneuronen variieren.

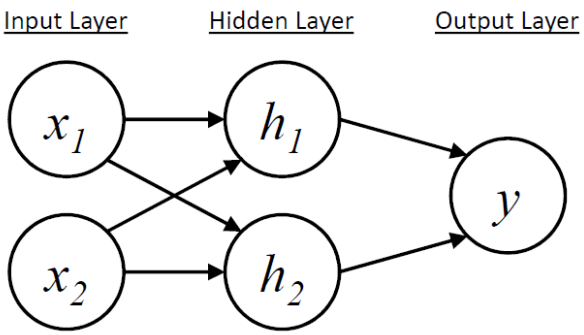

Abbildung 11-1: Einfaches Neuronales Netzwerk

Wie in Abbildung 11-2 dargestellt, besteht das implementierte *Feedforward Multi-Layer Perceptron* (MLP) Modell aus mehreren vollständig miteinander verbundenen Schichten von Neuronen (in der TensorFlow Terminologie auch als „*Dense*“ Schichten bezeichnet).

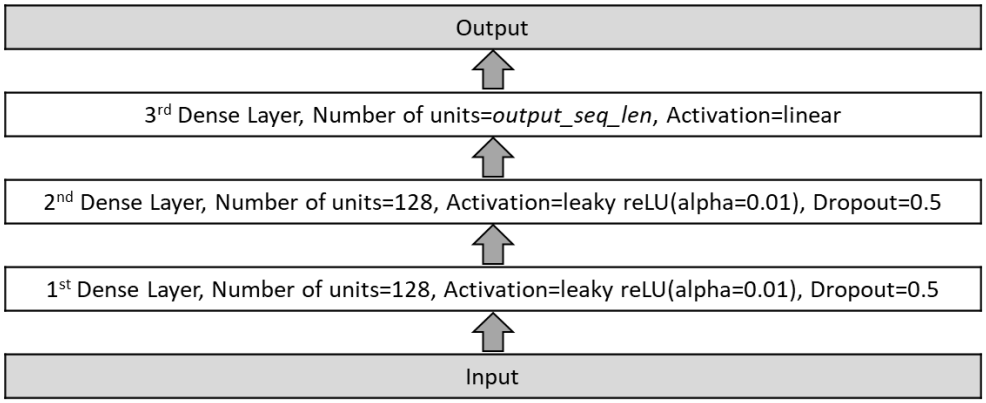

Abbildung 11-2: Implementiertes Feedforward Multi-layer Perceptron Modell - Definition

Der Wert des „*output_seq_len*“ ist ein Hyperparameter, der ausgewertet wird. Dieser Wert entspricht der Anzahl der Zeitschritte in der Zukunft, für die das Modell trainiert ist.

Mit Ausnahme der letzten dichten Schicht, wo eine lineare Aktivierungsfunktion verwendet wird, um die Regressionswerte auszugeben, verwenden alle anderen dichten Schichten eine leaky rectified linear unit (leaky reLU) mit *alpha=0.01*, die die Vorteile von reLU kombiniert, um die Konvergenz der stochastischen Gradientenabsenkung im Vergleich zu den *Sigmoid*- oder *Tanh*-Funktionen zu beschleunigen, ohne teure Operationen durchzuführen und die Fragilität anzugehen.

Zur Behebung von Überpassungsproblemen wird 50% Drop-Out auf alle ausgeblendeten Ebenen angewendet.

11.1.2 *Long Short Term Memory* Modell

Abbildung 11-3 zeigt das implementierte Long Short Term Memory (LSTM)-Modell. Wieder wird ein tiefes Netzwerk mit mehreren ausgeblendeten Ebenen verwendet.

Anstatt *Tanh* wie in der ursprünglichen LSTM-Literatur beschrieben zu verwenden, wird *Softsign* als Ausgangsaktivierungsfunktion in den LSTM-Schichten verwendet, da es schneller und weniger anfällig für Sättigung ist. Die Aktivierungsfunktion *Sigmoid* wird jedoch in den Gates (Eingang, Vergessen und Ausgang) verwendet, da sie einen Wert zwischen 0 und 1 angibt, d. h. um zu erfahren, wie viele Informationen durch das Gate fließen dürfen.

20% Ausfallende wird auf die versteckten LSTM-Schichten angewendet, aber nur auf die einmaligen Verbindungen.

Ähnlich wie beim MLP-Modell wird die abschließende dichte Schicht verwendet, um die Ausgabereihenfolge zu erzeugen, bei der die Länge auf „output_seq_len“ basiert.

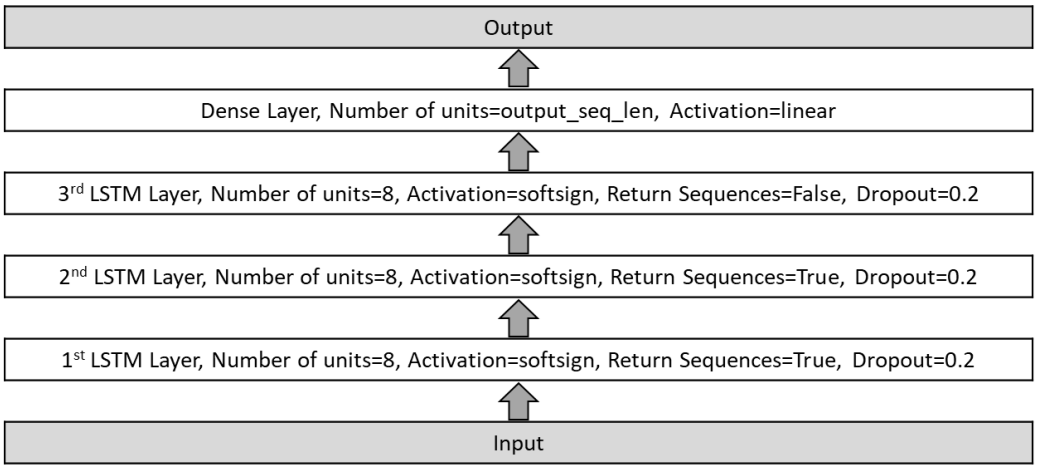

Abbildung 11-3: Long Short Term Memory Modell - Definition

11.1.3 *Sequence to Sequence* Modell

Wie in Abbildung 11-4 zu sehen ist, besteht das *Sequence to Sequence* (Seq2Seq)-

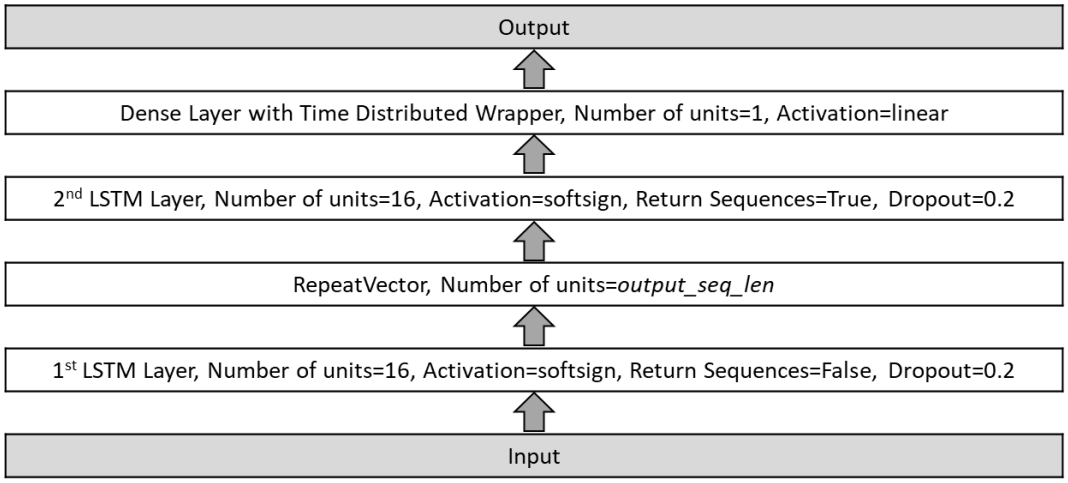

Abbildung 11-4: Sequence to Sequence Modell - Definition

Modell aus zwei Schlüsselteilen, die durch die RepeatVector-Schicht getrennt sind.

Der erste LSTM-Layer verarbeitet den Input und dient als Encoder zur Erfassung der zeitlichen Merkmale im Zeitablauf. Die zweite Schicht fungiert dann als Decoder, um den Ausgang zu erzeugen.

Beachten Sie, dass die erste LSTM-Schicht nicht die volle Sequenz an den RepeatVector zurückgibt. Der RepeatVector ist stattdessen eine Spezialschicht, die den finalen Ausgangsvektor des Encoders als konstanten Eingang für jeden Zeitschritt der zweiten LSTM-Schicht wiederholt. Dies berücksichtigt die unterschiedlichen Längen von Eingang und Ausgang und ermöglicht es, eine längere Eingangssequenz zu kodieren, um eine kürzere Ausgangssequenz vorherzusagen. Die Anzahl der Einheiten „output_seq_len“ entspricht der Anzahl der zu prognostizierenden Zeitschritte, d. h. der Länge der Ausgabe. Ein wesentlicher Unterschied zur letzten dichten Schicht hier zu der der Vorgängermodelle besteht darin, dass auf die Rücklaufsequenz aus der vorherigen LSTM-Schicht ein zeitverteiltes Wrapper aufgetragen wird, um die Ausgabesequenz zu erhalten. Damit wird sichergestellt, dass die Zustände aus jedem Zeitschritt zur Generierung der Ausgabereihenfolge herangezogen werden.

Ähnlich wie beim LSTM-Modell wird ein Ausfall von 20% auf die einmaligen Verbindungen der LSTM-Schichten angewendet.

11.2 Anomalieerkennung zur Fehlererkennung

Dieser Teil präsentiert drei Ansätze für die Erkennung von Anomalien zur Optimierung der Alarmanlage und Fehlerbeurteilung. Sie sind grafische, statistische und maschinelle Lernansatz. Im aktuellen Kontext sind Anomalien der interessante Teil von vibrationsbasierten Daten, die Fehler in mechanischen Teilen einer Windkraftanlage visualisieren. Die gesammelten Daten haben einen zeitlichen Aspekt.

11.2.1 Grafische Ansätze

Grafische Ansätze bieten eine große Wirkung für die Visualisierung von weitreichenden Korrelationen zwischen inter-abhängigen Datenobjekten. Es ist leicht zu verstehen mit minimalem technischen Wissen und einem Medium von großer Darstellung. Vier verschiedene Ansätze wurden untersucht:

1. Boxplot
   Boxplot ist ein fantastisches Werkzeug zur Darstellung von Positions- und Variationsinformationen in Datensätzen, insbesondere zwischen verschiedenen Datengruppen. Es ist der beste Weg, um Anomalien aufzuspüren. Tukey's Box Plot ist ein informeller Test, um das Vorhandensein

von Anomalien oder Ausreißern zu identifizieren. Aber Anomalie und Ausreißer sind im Kontext unterschiedlich. Anomalien sind das ungewöhnliche Muster in einer Beobachtung und Ausreißer ist ein Datenpunkt, der von anderen Beobachtungen entfernt ist.

Tukey's Boxplot ist ein einfaches und schnelles visuelles Werkzeug zur Datenanalyse. Es zeigt die Daten auf der Grundlage von fünf Zahlen beschreibenden Statistiken Zusammenfassung und Anomalien, falls vorhanden. Anomalien im Boxplot deuten manchmal auf den Nachweis nicht normal verteilter oder kontaminierter Daten hin.

Die folgende Abbildung 11-5 veranschaulicht das Ergebnis der Boxplots, die gegen die Datensätze durchgeführt wurden. Die markierten Punkte, die über dem Schnurrhaar liegen, gelten als extreme Anomalien. Die Breite des Boxplots gibt die Populationsgröße an.

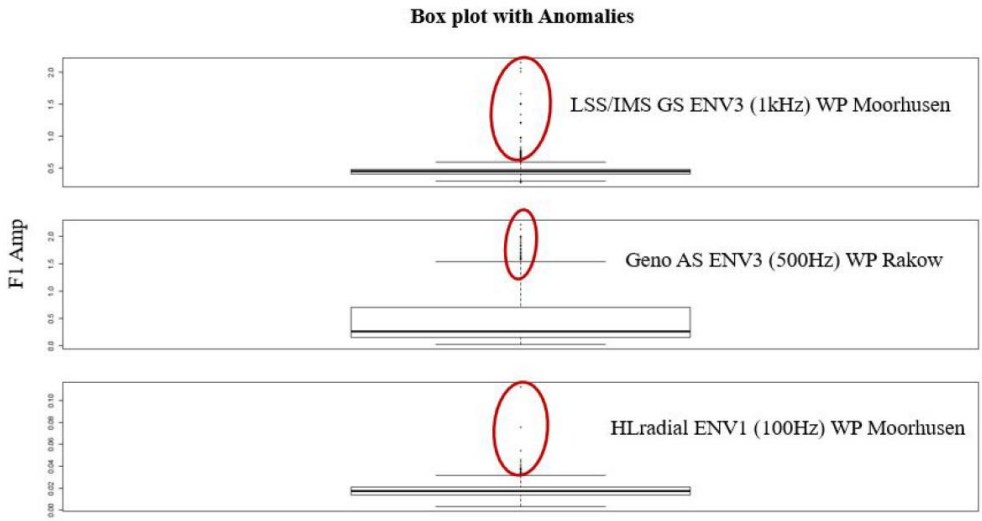

Abbildung 11-5: Boxplot

2. Scatterplots (Streudiagramme)

   Scatterplots sind die besten grafischen Diagramme, um die Korrelation zwischen Variablen, die Richtung der Beziehung zwischen Variablen und die Existenz von Anomalien zu identifizieren. Das *Scatterplot* verwendet die kartesischen Koordinaten, um die Beziehung zwischen geordneten Paaren von Zufallsvariablen darzustellen. Es hilft, einzigartige Natur und interessante Fakten über die Daten zu vermitteln. Meistens haben *Scatterplots* ein Muster.

   Wenn ein Datenpunkt, der nicht zu einem Muster oder einem Wert passt, der weit von der Hauptverteilung der Daten entfernt ist, dann werden diese Punkte als Anomalien in der Scatter-Plot-Darstellung betrachtet. Immer

hochdichte Regionen werden perzeptiv gruppiert. Die Anomalien im Boxplot unterscheiden sich von denen im Scatterplot. Wenn die Überlappung innerhalb der Punkte zunimmt, ist die Wahrscheinlichkeit groß, dass ein Scatterplot unwirksam wird.

In der Abbildung 11-6 zeigen blau markierte Punkte ein gesundes Verhalten und rote Farbpunkte ein ungewöhnliches Muster an. Diese Punkte befinden sich im weniger dichten Bereich und weit entfernt von den normalverteilten Datenpunkten.

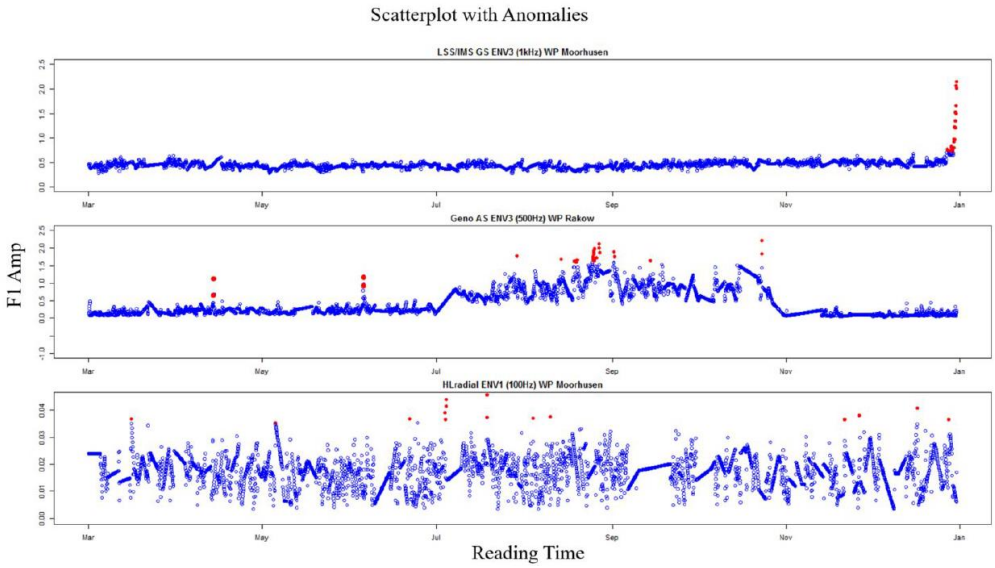

Abbildung 11-6: Scatter Plot

3. Quantile-Quantile Plot (Q-Q Plot)

   Ein Q-Q-Plot ist ein informeller grafischer Test, um festzustellen, ob eine Datensequenz aus einer bestimmten Verteilung stammt oder nicht. Andererseits kann sie auch beurteilen, ob zwei Datensätze von Stichproben aus derselben Verteilung stammen. Die Q-Q-Plot ordnen die Werte der Probendaten in aufsteigender Reihenfolge an und stellen diese Werte dann den erwarteten Werten für die angegebene Verteilung bei jedem Quantil in den Probendaten gegenüber. Die Quantil-Werte der Probendaten liegen entlang der *y-axis*, und die theoretischen Werte der angegebenen Verteilung bei gleichem Quantil liegen entlang der *x-axis*. Wenn die Probendaten aus einer bestimmten Verteilung stammen, fallen die Punkte im Plot entlang einer geraden Linie. Die Form normaler Q-Q-Plots hilft bei der Projektion von Verteilungsasymmetrie, Ausreißern, dicken Schwänzen, Multimodalität oder anderen Datenanomalien.

   Das Q-Q-Plot drückte die Werte in der realen Ebene R^2 aus. Die roten Punkte in Abbildung 11-7 sind die Anomalien, die im Q-Q-Plot dargestellt sind.

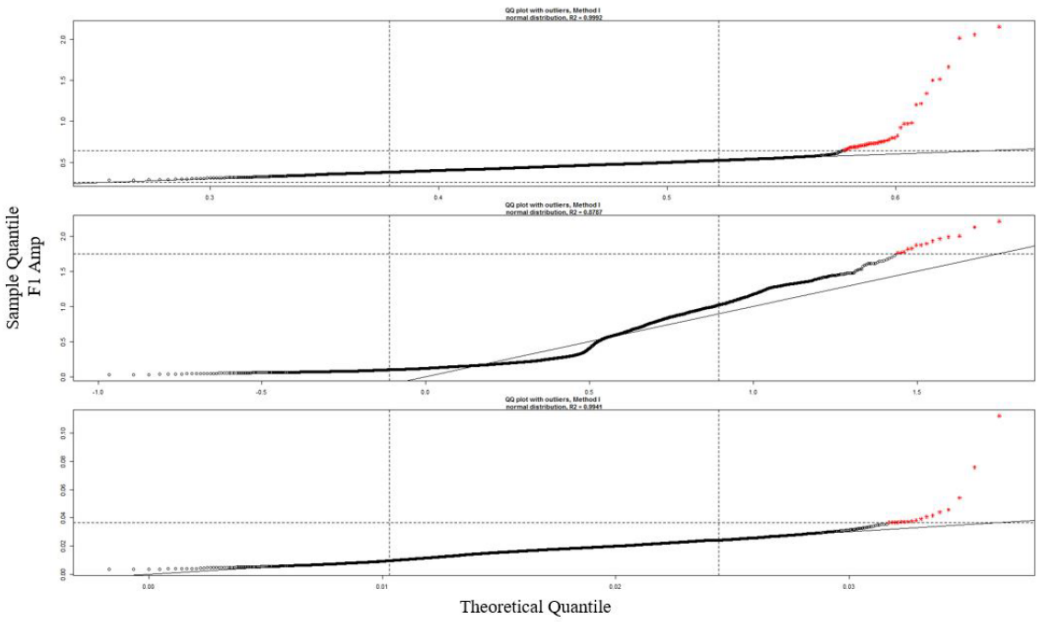

Abbildung 11-7: Quantile-Quantile Plot

4. Control Chart (Qualitätsregelkarte)

   Eine *Control Chart* ist ein hervorragendes statistisches Analyse-Diagramm, das verwendet wird, um zu verstehen, wie sich ein Prozess im Laufe der Zeit verändert. Es ist auch ausreichend für die Erkennung von Anomalien in Zeitreihendaten, wenn sie eine subtile Verschiebung von einem Mittelwert der spezifischen Messgröße anzeigen. Es ist ein einfaches Diagramm, das zur Überwachung der Stabilität und Kontrolle eines sich wiederholenden Prozesses verwendet wird, und es stellt die Variation über die Zeit dar.

   Wenn eine Variation als *Common Cause Variation* bezeichnet wird, dann nur, wenn ein Prozess stabil und kontrollierbar ist. Basierend auf den historischen Werten kann er voraussagen, wie sich der Prozess in der Zukunft verändern wird. Ein Prozess wird nur dann als statistische Kontrolle definiert, wenn er eine Variation der gemeinsamen Ursache erfährt. Ein Prozess ist außerhalb der statistischen Kontrolle oder instabil oder unberechenbar, wenn er einer Variation mit besonderer Ursache oder einer nicht-zufälligen Variation unterzogen wird. Eine Variation der besonderen Ursache kann aufgrund externer Faktoren auftreten, und eine Wahrscheinlichkeitsverteilung kann sie nicht definieren.

   Die Abbildung 11-8 zeigt die berechnete obere Kontrollgrenze und die untere Kontrollgrenze, die helfen, die Datenpunkte zu identifizieren, die sich zu stark von den normalen Daten unterscheiden.

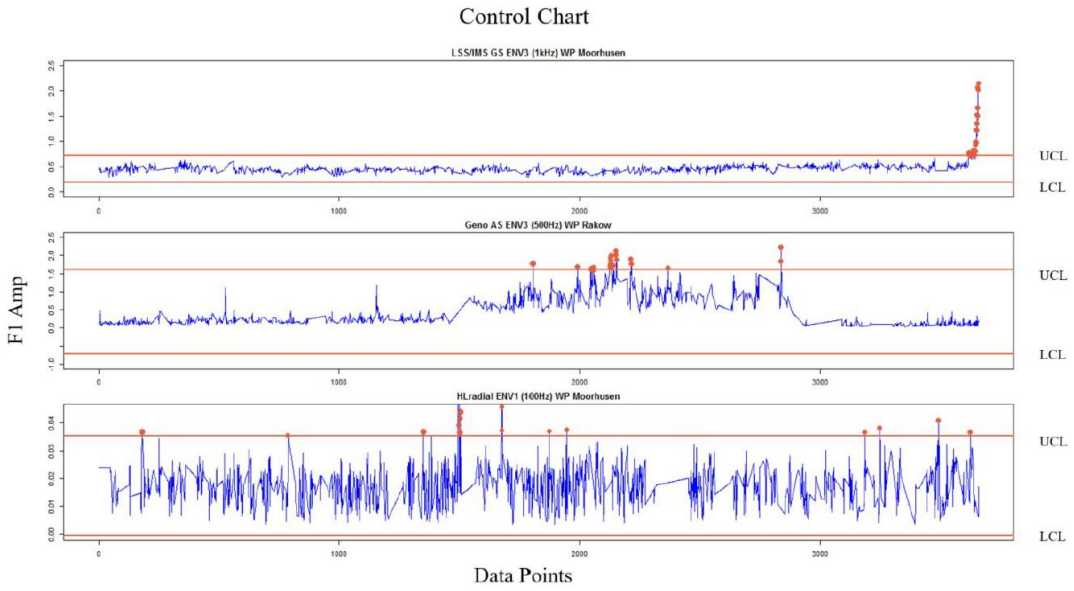

Abbildung 11-8: Control Chart

11.2.2 Statistische Ansätze

Die Forschung zur Anomalie Erkennung im Strom der Statistik begann vor fünf Jahrzehnten. Statistische Ansätze sind der frühere und Standard-Algorithmus verwendet, um Anomalien zu erkennen. Das Thema der statistischen Anomalie-Erkennung besagt, dass, wenn eine Beobachtung eine geringe Wahrscheinlichkeit in einer gegebenen statistischen Verteilung oder einem Modell hat, dann wird sie als Anomalie markiert, sonst wird sie als eine gewöhnliche Beobachtung betrachtet. Das Wesen dieses Ansatzes ist, dass die normale Beobachtung einem Erzeugungsmechanismus der statistischen Verteilung folgt, wie der Mittelwert, die Standardabweichung usw. und die abnormale Beobachtung von diesem Erzeugungsmechanismus abweichen.

Zwei am weitesten verbreitete statistische Verfahren zur Erkennung von Anomalien in univariaten Zeitreihen werden hier vorgestellt:

1. Grubbs-Test
   Grubbs-Test ist ein statistischer Test, der von Frank E. Grubbs entwickelt wurde, um Anomalien in einem univariaten Datensatz aufzuspüren. Er wurde 1950 eingeführt und 1969 und 1972 erweitert. Der *Grubbs-Test* lokalisiert Anomalien, die in einem univariaten Datensatz unter Verwendung von Mittelwert, Standardabweichung und tabellarischem Kriterium vorhanden sind. Es handelt sich dabei um einen statistischen Hypothesentest, der auch als *„maximum normed residual test“* oder „extreme studentized deviate“ (ESD)-Test bezeichnet wird, wobei davon ausgegangen wird, dass der Datensatz normalverteilt ist. Die Grubbs-Teststatistik ist definiert als die größte absolute Abweichung vom Stichprobenmittelwert in Einheiten der Stichprobenstandardabweichung.

Abbildung 11-9 zeigt das Flussdiagramm für *Grubbs-Test*-Algorithmus. Die unten aufgeführten Histogramme (Abbildung 11-10, Abbildung 11-11, Abbildung 11-12) sind das Testergebnis von Grubbs für die drei im Projekt verwendeten Datensätze.

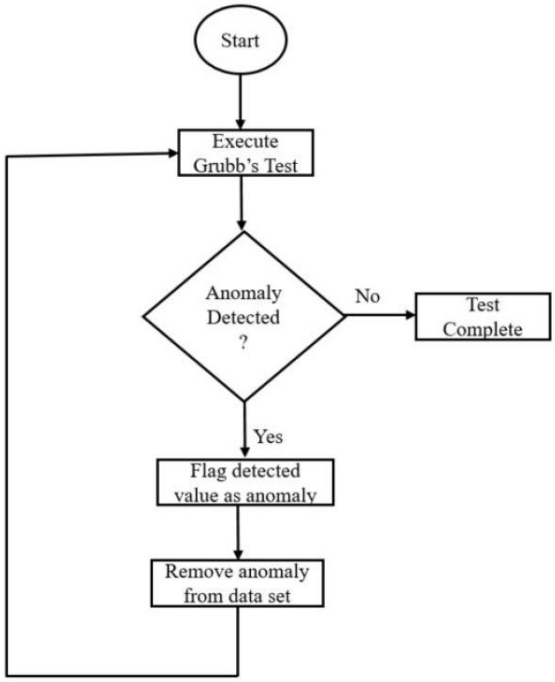

Abbildung 11-9: Grubbs Test Algorithmus

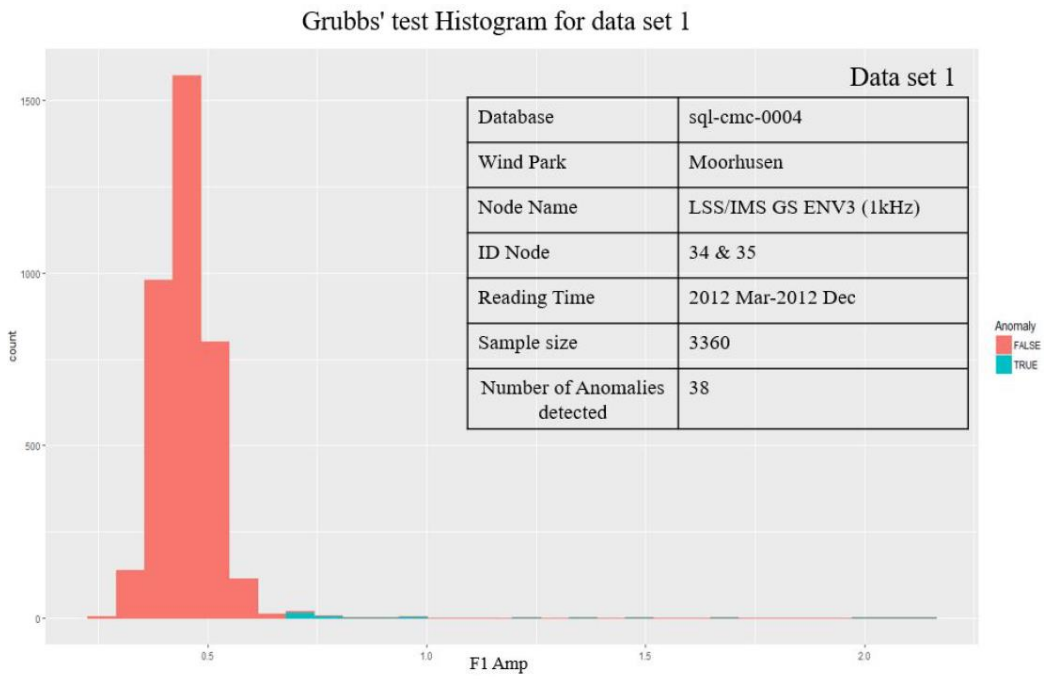

Abbildung 11-10: Grubbs test Histogram 1

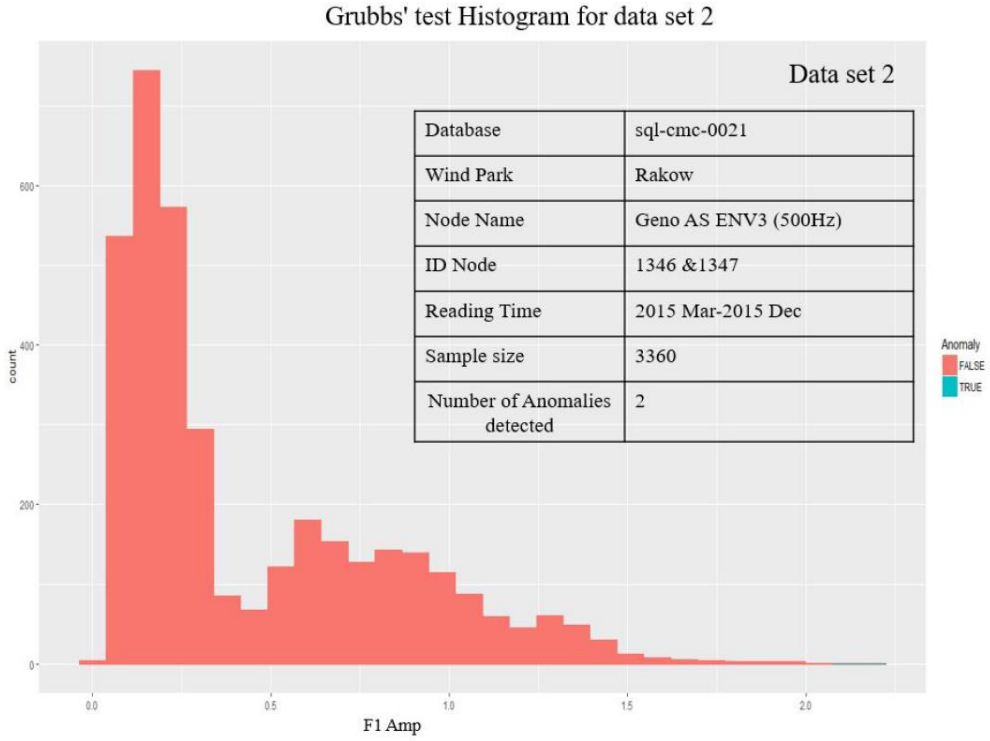

Abbildung 11-11: Grubbs test Histogram 2

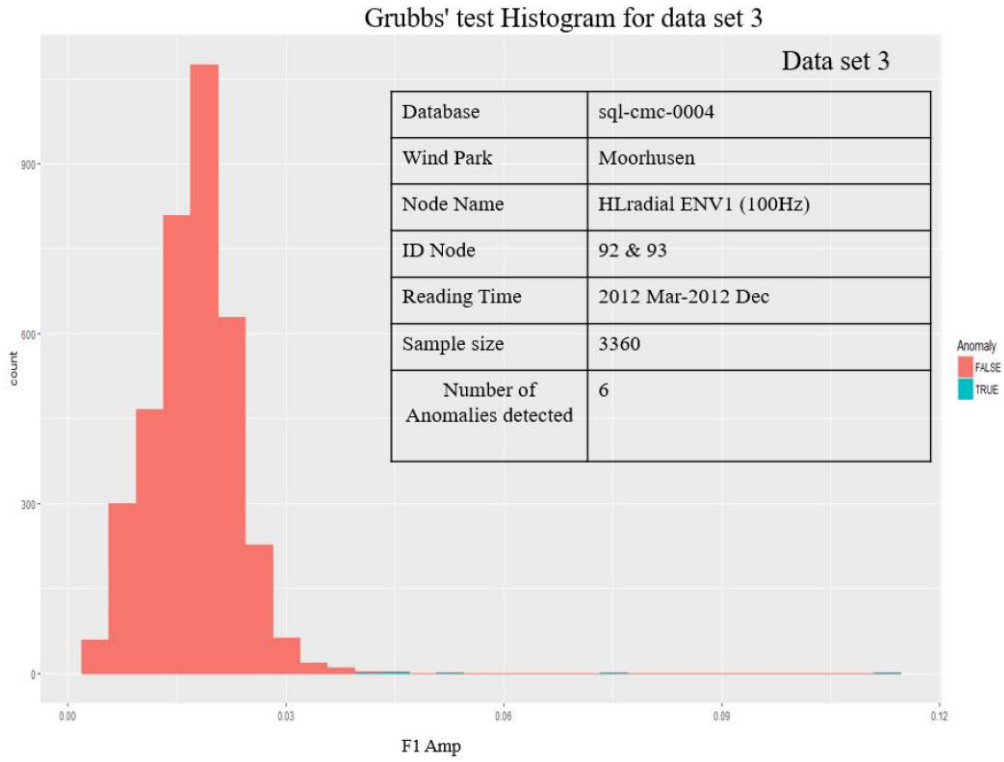

Abbildung 11-12: Grubbs test Histogram 3

2. Seasonal Hybrid ESD Algorithm (S-H-ESD)

   Seasonal Hybrid ESD Algorithmus ist eine erweiterte Version von Grubbs Test. Es kann zur Erkennung globaler und lokaler Anomalien verwendet werden. S-H-ESD liefert eine überlegene Leistung durch Zeitreihenzerlegung und robuste

statistische Metriken und Techniken. Im Vergleich zu anderen Algorithmen ist S-H-ESD besser in der Lage, den hohen Prozentsatz von Anomalien in Zeitreihendaten zu finden. Die meisten der traditionellen statistischen Methoden verwenden Mittelwert und Standardabweichung, um Anomalien zu finden. Diese Metriken reagieren empfindlich auf anomale Daten und können dazu führen, dass tatsächliche Anomalien als nicht anomal gekennzeichnet werden. Es kann dazu führen, dass die Zahl der Falschnegative zunimmt. S-H-ESD-Algorithmus verwendet mediane und mediane absolute Abweichung. Sie sind statistisch robust gegenüber Mittelwert und Standardabweichung.

Die folgenden Abbildungen (Abbildung 11-13, Abbildung 11-14, Abbildung

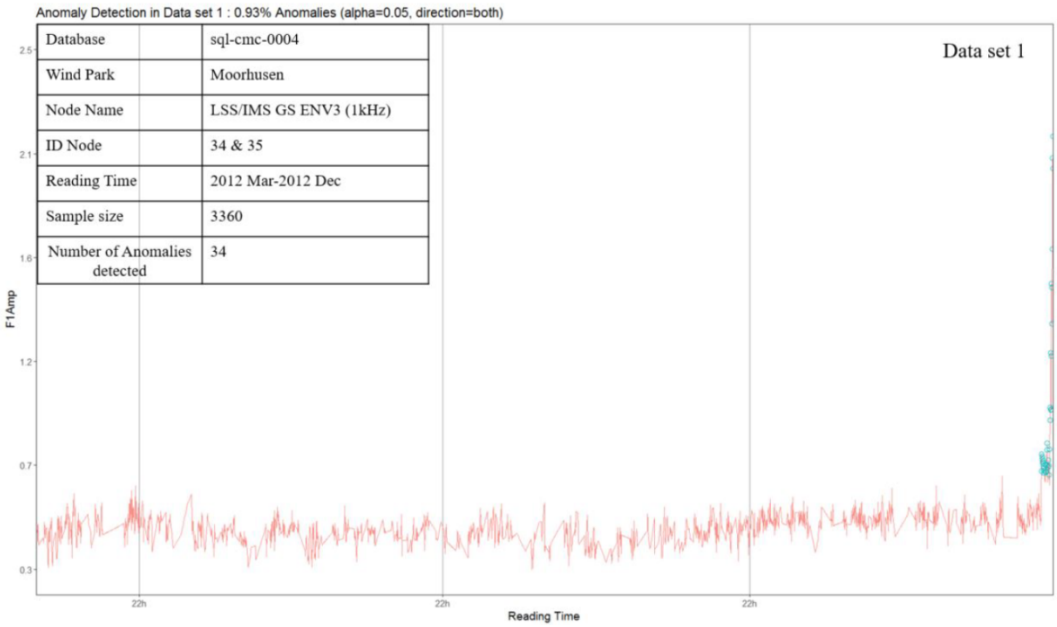

Abbildung 11-13: Anomalieerkennung Datensatz 1

11-15) zeigen die Ergebnisse, wenn sie auf die Datensätze angewendet werden.

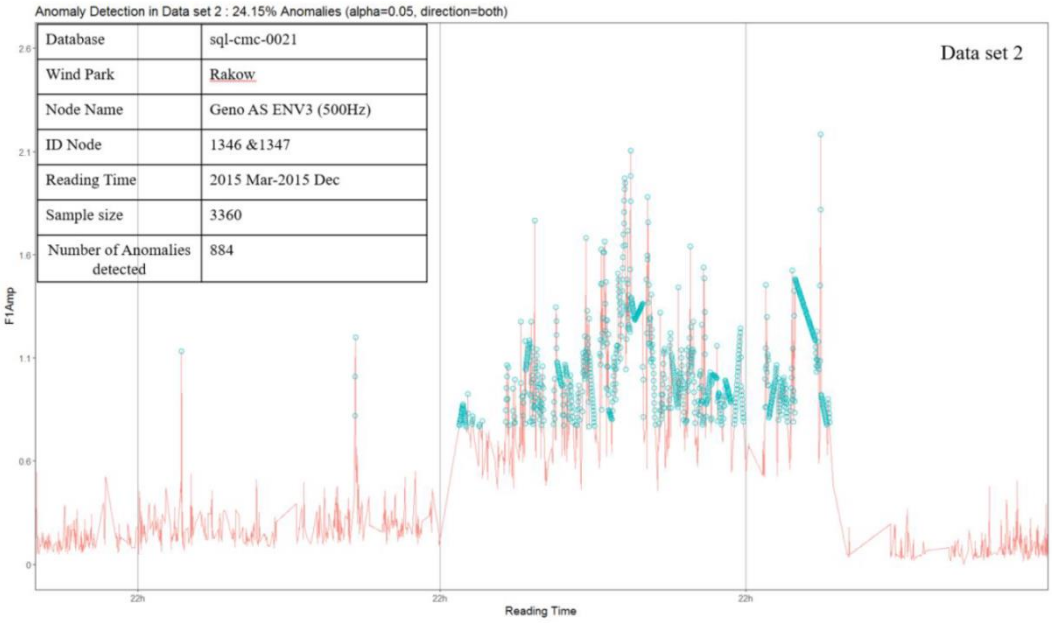

Abbildung 11-14: Anomalieerkennung Datensatz 2

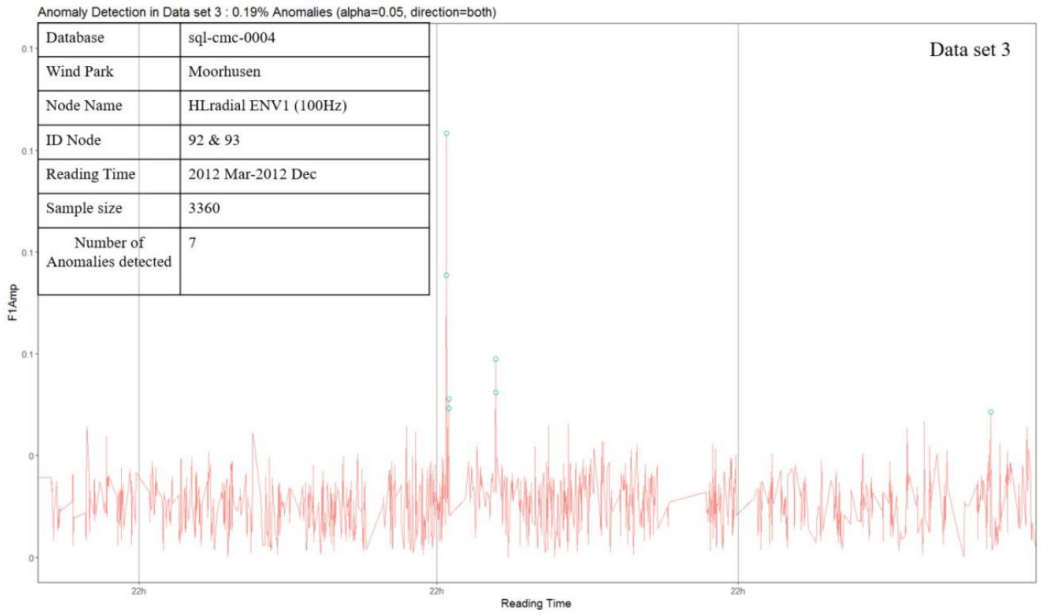

Abbildung 11-15: Anomalieerkennung Datensatz 3

11.2.3 Maschinelle Lernansätze

Arthur Samuel beschrieb *Machine Learning* erstmals im Jahr 1959 als "*Machine Learning* ist das Feld, das Computer die Fähigkeit zu lernen gibt, ohne sie explizit zu programmieren". Es wird auch als das Forschungsgebiet unter Künstliche Intelligenz betrachtet. Es liegt daran, dass Maschinen mit Hilfe von historischen Daten und Statistiken Entscheidungen treffen können. Das maschinelle Lernen ist auch ein Teil

des Data Mining und wird in Anwendungen wie Muster- und Bilderkennung, Web-Such-Filterung, Textanalyse, E-Mail-Spam-Filterung, Netzwerk-Intrusion Detektion, etc. eingesetzt. Das Maschinelle Lernen kann in das betreute Lernen und das unbeaufsichtigte Lernen eingeteilt werden. In beaufsichtigtem Lernen, wie der Name schon sagt, braucht es Hilfe als markierte Daten und genügend Wissen über die Daten. Im unbeaufsichtigten Lernen gibt es vorherige Kenntnisse über Daten nicht erforderlich und Entscheidungen basieren auf dem gesamten Input.

Zwei unbeaufsichtigte Techniken des maschinellen Lernens wurden überprüft:

1. *K-means Clustering*-Algorithmus
   K-means Clustering ist der am häufigsten verwendete unbeaufsichtigte Algorithmus für maschinelles Lernen. Es teilt einen gegebenen Datensatz in einen Satz von *k* Clustern oder Gruppen auf, wobei *k* die Anzahl der vom Analytiker genannten Cluster ist. Der *K-means Clustering*-Algorithmus ist eine auf dem Schwerpunkt basierende Partitionierungstechnik, und jeder Cluster wird durch sein Zentrum (Schwerpunkt) repräsentiert. Es teilte n Beobachtungen in *k* Cluster auf der Grundlage der Ähnlichkeit zwischen einander. Die Beobachtungen innerhalb eines Clusters sind so ähnlich wie möglich. *K-means Clustering* kann mit größeren Datenmengen umgehen als hierarchisches Clustering. Der Standard-Algorithmus ist der Lloyd-Forgy-Algorithmus, der eine verfeinerte iterative Technik ist, bei der die euklidische

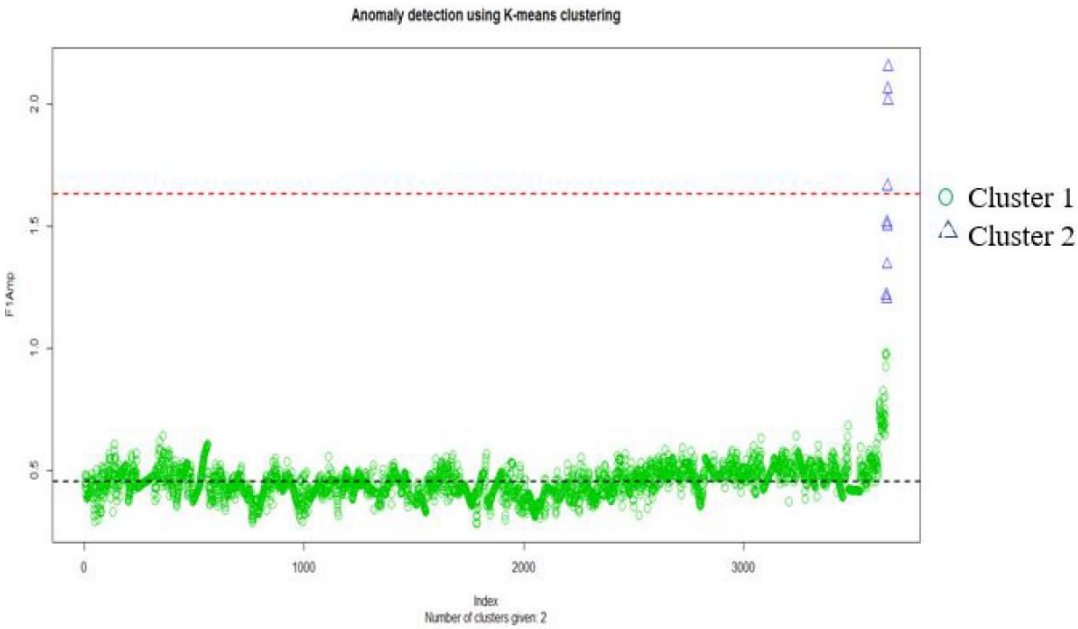

Abbildung 11-16: K-means Clustering

Distanz verwendet wird.

In der Abbildung 11-16 sehen wir zwei Cluster, die sich durch zwei verschiedene Farben wie Blau und Grün unterscheiden. Der grüne Kreis zeigt

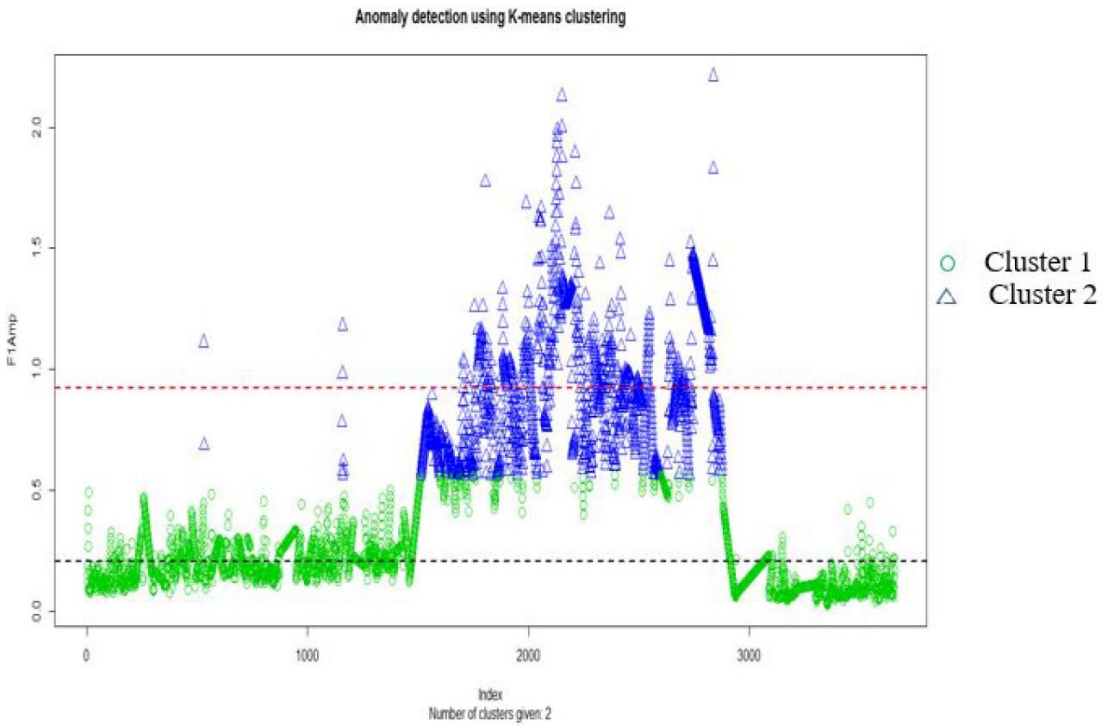

Abbildung 11-17: K-means Clustering

das normale Verhalten und das blaue Dreieck das anomale Verhalten an. Linien werden auf Clusterzentren gezeichnet. Für die anderen Datensätze gilt das gleiche Verfahren wie in den folgenden Abbildung 11-17 und Abbildung 11-18.

2. Density-Based Clustering-Algorithmus (Dichtebasierter Clustering)

Der *Density-Based* Clustering-Algorithmus ist eine unbeaufsichtigte maschinelle Lerntechnik und einer der entfernungsbasierten Ansätze. Es

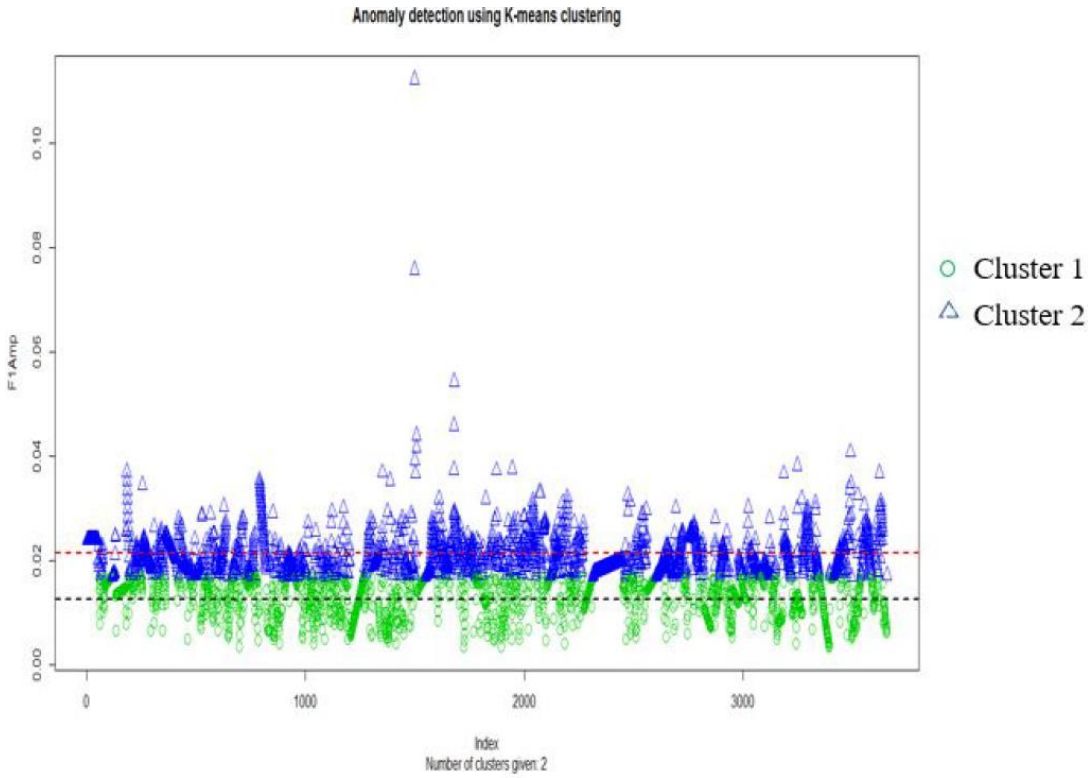

Abbildung 11-18: K-means Clustering

klassifiziert oder identifiziert Cluster in den Daten auf der Grundlage der Regionen mit hoher Dichte. Die Datenpunkte in den Bereichen mit geringerer Dichte werden als Anomalien betrachtet. *Density Based Spatial Clustering of Applications with Noise* (DBSCAN) wurde erstmals 1996 von Ester et al. eingeführt und hat bewiesen, dass es der stärkste Algorithmus ist, der auf Massen- und dichten Datensätzen basiert, um Anomalien zu finden. Es unterscheidet sich von der üblichen Clustering-Technik dadurch, dass es anomale Punkte definiert, die in keine Cluster passen.

Die folgenden Abbildungen zeigen das Ergebnis der Anwendung von DBSCAN

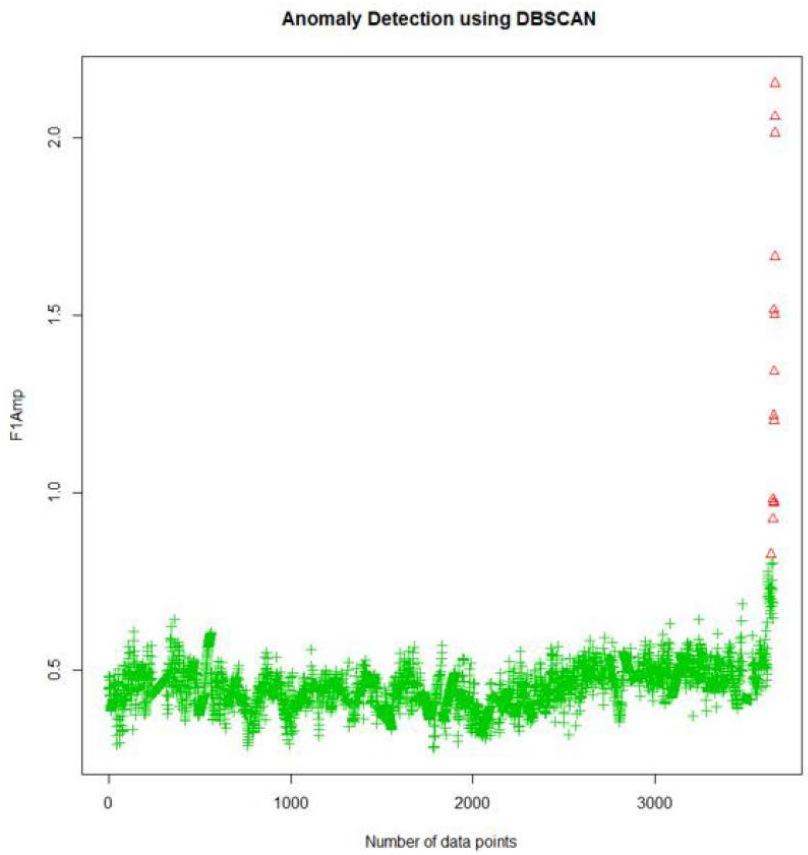

Abbildung 11-19: DBSCAN Datensatz 1

auf die Datensätze.

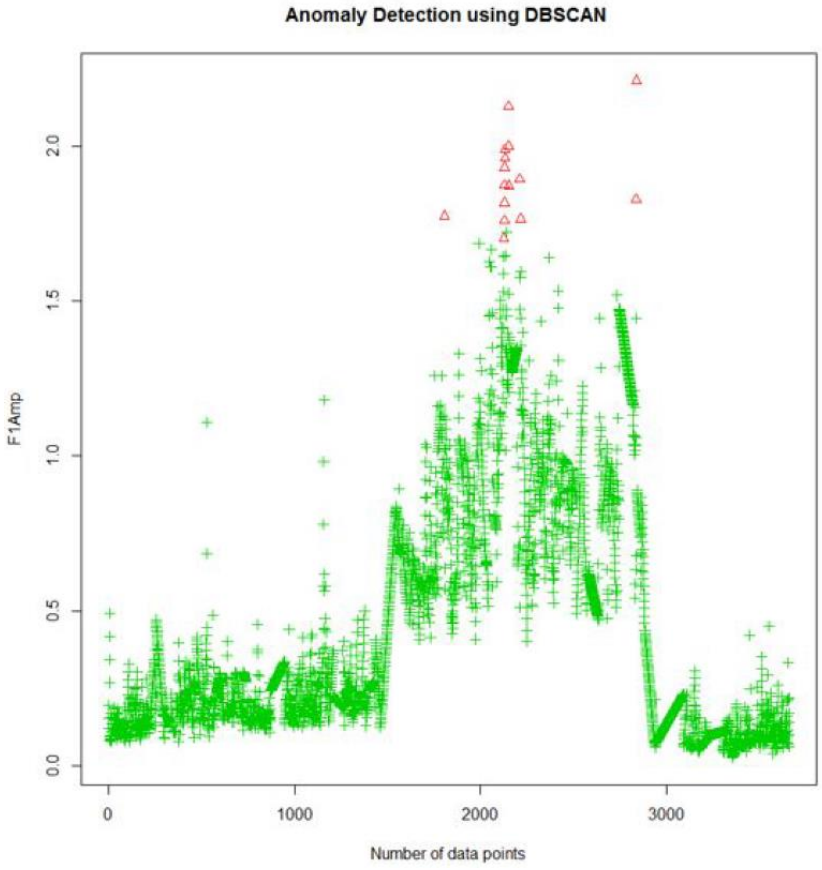

Abbildung 11-20: DBSCAN Datensatz 2

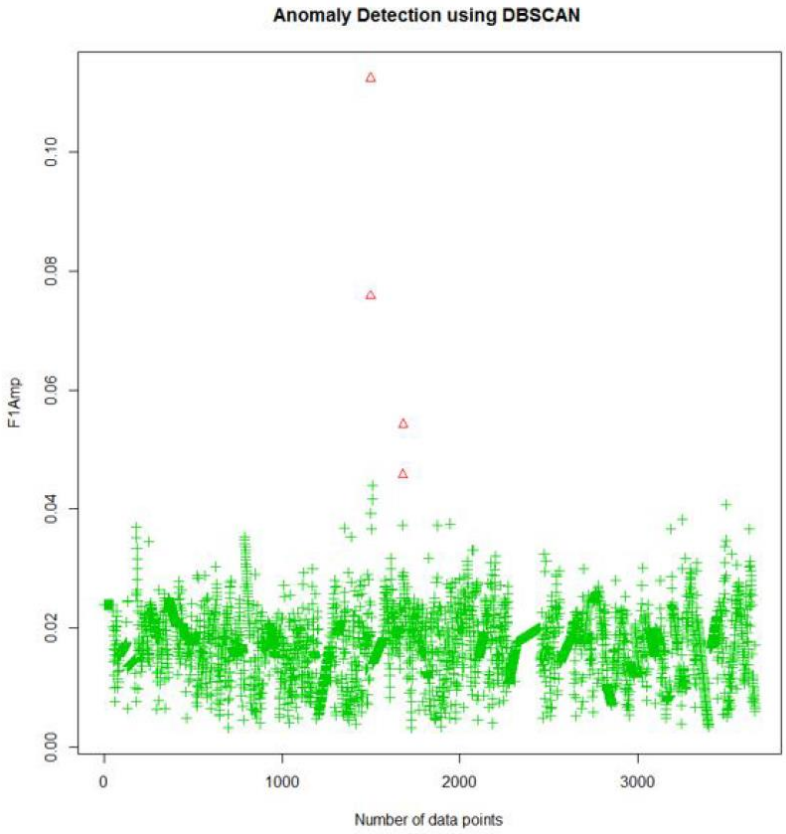

Abbildung 11-21: DBSCAN Datensatz 3

12 Schlussfolgerungen

Aus der Projektarbeit ergeben sich Schlussfolgerungen in den Bereichen:

- Verbesserung von IT-Strukturen
- Vermehrter Einsatz neuer System-Architekturen
- Vermehrter Einsatz moderner Techniken zur Datenanalyse

Diese werden in der Folge dargestellt.

12.1 Hebung von Potenzialen bei IT-Strukturen in kleinen- und mittelständischen Betrieben

Dieser Abschnitt richtet sich an mögliche Verbesserungen, die auch ohne die Verwendung der neusten Technologien in Unternehmen umgesetzt werden können. Als Basis dient an dieser Stelle das gewonnene Wissen aus der Stammdatenbank. Viele Daten sind bereits im Unternehmen vorhanden, aber es fehlt an einer gut organisierten Datenhalten, Analyseverfahren, IT-Personal und einem Überblick über die verschiedenen Möglichkeiten.

Für die angesprochenen Herausforderungen existieren unterschiedliche Lösungsansätze. Eine Möglichkeit wurde mit diesem Projekt bzw. Bericht aufgezeigt. Das Unternehmen konnte Einblicke in eine andere Datenhaltung gewinnen und Verbesserungspotenziale für Ihre IT-Infrastruktur entdecken. Des Weiteren sollten sich Unternehmen besser über IT-Systeme Informieren, da an dieser Stelle viel Zeit der Mitarbeiter und somit Geld gespart werden kann. Dabei müssen keine ganz neuen Architekturen zum Einsatz kommen. Sondern es würde ausreichen, wenn bereits bestehende Industriestandards wie SQL Datenbanken verwendet werden.

12.2 Einsatz neuer System-Architekturen

Big Data Architekturen werden in den nächsten Jahren immer mehr an Bedeutung gewinnen. Sie sind Kostengünstig, Skalieren im Verhältnis zu den klassischen Architekturen deutlich besser und ermöglichen Echtzeitanalysen mit geringen Mehraufwand. Diese Attribute resultieren unteranderem aus der horizontalen Skalierbarkeit. Horizontale Skalierbarkeit bedeutet, dass anstatt einen großen Server zu verwenden, wie es bei vertikalen Skalierung üblich ist, werden mehrere Server verwendet (vgl. Abbildung 12-1). Dadurch können bei einer Erhöhung der Anforderungen an das System, diese durch das Anschaffen von mehr Servern erfüllt werden. Die neuen Server werden mit in das bestehende System integriert und skalieren, je nach System, beinah linear. Deshalb entstehen Kosten erst, wenn tatsächlich mehr Leistung benötigt wird. Während bei einer vertikalen Architektur der Großteil der Kosten dann entsteht, wenn der Server gekauft wird. Sollte dieser nicht mehr über eine ausreichende Leistung verfügen, muss ein komplett neuer Server gekauft werden, der den alten vollständig ersetzt.

Die Skalierbarkeit bezieht sich dabei nicht nur auf große Datenmengen. Inzwischen performen in vielen Bereichen *Big Data* Technologien auch bei kleinen Datenmengen ähnlich wie etablierte Technologien. Dadurch lohnt es sich für neue Anwendungen auf einer *Big Data* Architektur aufzubauen. Denn auch wenn zur Entwicklungszeit noch keine Architektur zur Verarbeitung benötigt wird, die mit tausenden Gigabyte umgehen kann, sind Big Data Architekturen sehr kostengünstig und das Unternehmen ist auch für die Zukunft gerüstet.

Eine besondere Herausforderung bei der Umsetzung von neuen Anwendungen sind die mangelnden Fachkräfte. Da die Big Data Technologien ein sehr umfangreiches Wissen voraussetzen, werden ins Besondere kleine und mittelständische Unternehmen Schwierigkeiten haben sich das Personal einzukaufen oder eigenes Personal weiterzubilden. Jedoch arbeiten die Industrie und Bildungseinrichtungen schon an Lösungen. So bieten immer mehr Universitäten Kurse mit einem Fokus auf

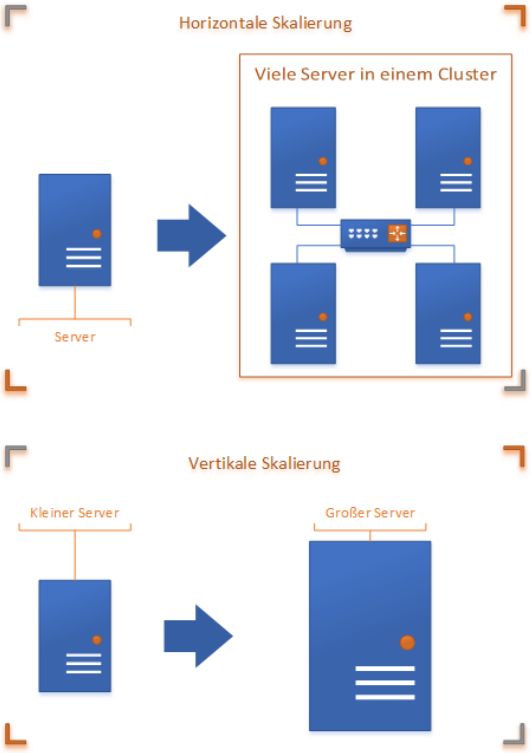

Abbildung 12-1: Horizontale vergleich mit Vertikaler Architektur

Big Data an. Auch werden immer mehr Online Ressourcen zur Verfügung gestellt, die häufig kostenlos zur Verfügung stehen. Darüber hinaus bemühen sich die Entwickler der Technologien, den Umstieg so einfach wie möglich zu gestalten, sodass ein Umstieg deutlich weniger Wissen erfordert als noch vor fünf Jahren.

Darüber hinaus lässt sich diese Art der Architektur sehr gut in der Cloud skalieren. Die Cloud bietet Unternehmen sehr flexible Kostenmodelle an. Die meisten Anbieter lassen sich für die gebrauchte Leistung pro Stunde bezahlen. Somit kann sich ein

Unternehmen zusätzlich Leistung hinzukaufen für genau die Zeit, die benötigt wird. Auch fallen Instandhaltungs- und Lagerkosten für die Hardware weg. Mehr zur Cloud wird auf Seite 67 besprochen.

12.3 *Moderne Techniken der Datenanalyse: Machine Learning / Deep Learning*

Machine und *Deep Learning* wird immer mehr an Bedeutung gewinnen. Die Algorithmen können inzwischen mit deutlich weniger Aufwand entwickelt werden als noch vor zwei Jahren. Dies resultiert aus den etablierten Frameworks, wie TensorFlow und Spark.

Für kleinere Firmen, die über nicht so viele Anlagen verfügen, würde es sich lohnen, Daten mit anderen kleineren Firmen zu teilen. Dadurch können Synergieeffekte entstehen. Da es sich häufig um Maschinendaten handelt, können die Daten gut anonymisiert werden. Somit geben die Unternehmen keine unnötigen Informationen preis. Im Fall, dass eine Firma über genug eigene Daten verfügt, muss keine Anreicherung von anderen Firmendaten durgeführt werden. Stattdessen können mit den eigenen Daten gute Ergebnisse erreicht werden.

In jedem Fall können Unternehmen, die viele Daten zur Verfügung haben, mehr Chancen in verschiedenen Bereichen finden, wie zum Beispiel:

- Industrie 4.0
- Smart Electric Grids
- Predicative Maintenance
- Priorisierungen von Maschinenalarmen
- Vorhersagen von Energiepreisen

13 Zukünftige Arbeiten

Nach der Beendigung des Projekts sind noch immer einige Fragen offen und neue sind in der Zwischenzeit hinzugekommen. Daraus lassen sich folgende zukünftige Arbeiten ableiten.

13.1 Machine Learning und *Big Data*

Die verwendeten Technologien haben im Verlauf große Fortschritte gemacht. Damit sind unsere Empfehlungen und Einschätzungen der Technologien nicht falsch, dennoch könnten dieselben Framework nochmals evaluiert werden. Zum Beispiel haben sich einige der Technologien insbesondere Spark und Tensorflow sehr verändert. Darüber hinaus sind können innerhalb derselben Frameworks auch noch andere Methoden und Herangehensweisen getestet werden. Insbesondere neue Ansätze von Spark verteiltes *Machine Learning* und *Deep Learning* sowie *Streaming* in einer hohen Abstraktionsebene anzubieten, bieten sich für zukünftige Projekte an. Für einen Streaming-Service wäre zum Beispiel ein interessantes Gebiet die Echtzeit-auswertung von Daten.

Ansonsten wurden wegen der zeitlichen Beschränkung nicht alle der möglichen Frameworks vollständig betrachtet. Zu diesen gehören unteranderem:

- Elasticsearch
- Kafka
- HBase
- Intel Neon
- Neo4j

13.2 Cloud

Ein weiterer Aspekt ist eine *Cloud*-Architektur. Die meisten der vorgestellten Anwendungen lassen sich auch in der Cloud nutzen. In der Cloud können Kosten geringgehalten werden, sofern die deutschen Datenschutzgesetze eine Auslagerung der Daten zulassen. Daher ergeben sich folgende Fragen in diesem Bereich:

- Was sind mögliche Kostenmodelle in der Cloud?
- Wie sieht Datensicherheit in der Cloud aus?
- Privat, Hybrid oder Public Cloud?

14 Anhang

14.1 Analyse und Optimierung der Datenströme in der Zustandsüberwachung von Windenergieanlagen hinsichtlich ganzheitlicher Betrachtung komplexer Betriebs- und Schadenszustände

Masterthesis von Nikita Loban (Abstract)

Das Hauptziel der vorliegenden Master-Arbeit – die Verbesserung der Qualität der bestehenden Geschäftsprozesse mittels der Analyse und Optimierung der Datenströme im Monitoring Center der CMC GmbH und der nachfolgenden Entwicklung einer Software-lösung zur einheitlichen Verwaltung der Daten aus der Zustandsüberwachung der Windenergieanlagen – konnte weitgehend erreicht werden.

Die Analyse der Geschäftsprozesse erfolgte nach dem Subsystem „Systemerstellung“ des V-Modells 97. Diese stellte die Aktivitäten zur Erfassung des Ist-Zustands im *Monitoring Center* zur Verfügung und ermöglichte die Formulierung der Anwender-anforderungen an die entwickelte Softwarelösung. Die Schwachstellenanalyse zeigte, dass die Verwaltung der Daten meistens mit Hilfe von Microsoft Office 2010 erfolgt. Bei der Informationsermittlung sind die Mitarbeiter auf eine manuelle Suche und einen Vergleich von mehreren Quellen angewiesen. Der dafür benötigte Zeitaufwand ist enorm, da die Dokumentationen öfters die Information doppelt verwalten und nicht selten widersprüchliche Angaben enthalten. Ausgehend von den gewonnenen Kenntnissen sind in den regelmäßigen Workshops und Sitzungen mit Ingenieuren des Monitoring Centers Anforderungen an vier Datenbanken sowie an deren Funktionalität formuliert worden.

Auf Wunsch des Auftraggebers erfolgte die Realisierung der Softwarelösung mittels Microsoft Excel und der Programmiersprache Excel-VBA. Da es sich dabei um keine Datenbanksoftware handelt, mussten die Datenbanken auf vier xlsm-Dateien auf-geteilt werden, die jeweils ein Datenbanksystem darstellen. Damit ist der Absturz der Softwarelösung wegen der Überlastung des Arbeitsspeichers durch die große Daten-menge vermieden worden. Aus dem gleichen Grund konnten die allgemeingültigen Anforderungen an die relationalen Datenbanksysteme nicht komplett erfüllt werden.

Das DBS „WEA-Stammdaten“ bildet das Fundament des Gesamtsystems. Dieses erfasst die Stammdaten der WEA und stellt diese den anderen Systemen zur Verfügung. Das DBS „Logbuch“ ermöglicht die Erfassung der relevanten Ereignisse in der Zustandsüberwachung der Antriebstränge. Wobei das DBS „Störungsliste“ in der Lage ist, alle Störungen und Ausfälle des *Condition Monitorings* Systems zu registrieren. Das DBS „Schadensdatenbank“ dokumentiert alle detektierten Schäden an den Hauptkomponenten eines Antriebstrangs.

14.2 Anomaly Detection in Periodic Big Data Streams of Wind Energy Conversion Systems for Alarm Optimization

Masterthesis von Norvin Thomas (Abstract)

With the increasing demand for electrical power and increased environmental regulations, wind energy turbine parks gained popularity around the world. Thus, the continuous availability, minimized downtime and early failure detection of a wind turbine is very crucial in this industry. Arbitrary failure of an element in wind turbine results in an immense economic loss. Condition Monitoring (CM)is the most efficient technique to prevent such misfortune failure and unplanned outages. Besides, most CMs demand a substantial number of fault indicators for the accurate diagnosis of component failures.

Statistical and machine learning techniques can surpass problem as mentioned above in CM; it focusses on developing a system that improves its performance based on historical data in place of understanding the process that generated the data. Anomaly Detection is the process of finding patterns in data that do not belong to a predefined expected behavior. Anomaly Detection has inevitable relevance in real world applications like fault detection, intrusion detection, fraud detection, system health monitoring, etc.

The primary focus of this thesis is to develop an early failure diagnosis model for alarm optimization in wind energy conversion systems based on graphical, statistical and unsupervised machine learning approach. It consists of four sections as follows: section one discussing existing analysis and techniques for fault detection in industrial application. Section two contains the algorithms purposed for anomaly detection in wind turbines. In section three, the proposed algorithm is validated with the existing test data from rolling elements of wind turbines. Then section four evaluates the accuracy of proposed algorithm with the concerned diagnostic engineer. It also emphasizes on the evaluation of different clustering techniques, and its significance in data mining.

14.3 Application of Machine Learning Techniques to Drive Decision-Making in Fault Diagnosis and Prognosis in Condition Monitoring Systems: Using a Cluster Computing Framework

Masterthesis von Soo Yam Tan (Abstract)

Condition monitoring systems in wind turbines aim to collate sensor data from various gear systems within a wind turbine to generate meaningful trends and highlight potential occurrences of equipment failure to empower engineers to make informed decisions in fault diagnosis and prognosis. These systems establish a condition-based maintenance and repair strategy as opposed to conventional schedule-based preventive maintenance. To achieve this, sensor data is collected over the lifetime of a wind turbine, building up a sizable yet valuable repository of historical data.

Recent advancements in machine learning techniques, especially in artificial neural networks, coupled with the availability of data have shown how machine learning can leverage on data to perform promising predictions. The intent is not to replace the role of engineers but to add a layer of dependable analysis that can improve decision-making. Timely and accurate diagnosis of the health of the wind turbine is critical in minimizing unplanned downtime which ultimately affects revenue.

This master thesis investigates the latest machine learning techniques, focusing primarily on artificial neural networks. Three different types of deep networks, Feedforward Multi-Layer Perceptron, Long Short-Term Memory and Sequence to Sequence models, are implemented on a practical time series prediction problem using real data provided by a condition monitoring service provider for wind energy systems. To illustrate the scalability of handling large volume of data, these models are trained using a distributed architecture leveraging on Apache Spark.

14.4 Big Data Design Practices and Implementation with Focus on Architectural Aspects: An effective decision forecast for different plots under different technologies

Masterthesis von Shaharyar Khan (Abstract)

With the increase in use of services, clients and users are also increasing which is directly proportional to the generation of humongous data. The topic of data management and analytics is very popular and nowadays a lot of research is being held on it. These days many technology vendors are providing many advance solutions and with the advancement in research, NoSQL solutions are being introduced. The real problem occurs when technologists has to decide the proper and reliable solution for a scenario and has to choose among different tools which suits best under the situation. There is no justified rule which can be followed under such kind of circumstances especially when each vendor is claiming themselves as best. There is also a possibility that selected tool is much less for the use case so it urges the need to add other components to complete the solution which can grow up to many component. This approach will end up in a complex solution which is definitely difficult to manage.

The research develops the concept under different scenarios at abstract level so best suited solution can be decided. These architectural aspects provide the details of different components which can be used according the desired plot.

The goal of this thesis is to provide a way by which technologists can easily choose the better option with the accordance of use case. The main focus will be on the designing of architecture by use of tools. The development of these solutions will provide insight about the tools and technology selection so that selected tool wouldn't be too high or much less in functionality for the scenario.

There are some implications in this research because of having less information about the topic. There is no rule for testing rather than actual production environment which can be overcome with test data but still test data can't provide absolute confirmation of proof of concept until it reaches to real plot.

Results will indicate that if considerations provided in this thesis are being taken then there is less possibility of having useless solutions which can be expensive as per their usage. Results also demonstrates that provided information by research having more suited solution rather bigger or lesser then need. At the end, solution should be less complicated, easy to manageable and appropriate as a component for use.

14.5 Data Driven Prognostic Methods for Fault Detection in Wind Energy Conversion System: Pattern Recognition in Time Series using Dynamic Time Warping

Masterthesis von Vini Vasundharan

Renewable energy sources have gained a tremendous attention due to the recent energy crisis and the demand for clean energy. As the demand for wind energy continues to grow at exponential rates, reducing Operation and Maintenance (OM) costs and improving reliability have become top priorities in Wind Turbine (WT) maintenance strategies. In addition to the development of more highly evolved WT designs intended to improve availability, the application of reliable and cost-effective Condition-Monitoring (CM) techniques offers an efficient approach to achieve this goal. It is important to maintain the healthy condition of the running turbine because the consequences after faults are miserable.

Preventative maintenance of wind mills using condition monitoring technologies have become really important in today's world as the demand for renewable energy sources has dramatically increased. This leads to the fact that it is necessary to reduce the cost of operation and maintenance of wind turbines. It is very important to detect the faults and failures as early as possible to minimize downtime and maximize productivity.

Pattern recognition and finding similarity among time series is a well discussed topic. This thesis mainly aims at developing the failure prognostic model based on pattern matching and similarity measures in time series. The problem of efficiently and accurately locating patterns in time series is a non-trivial problem in many applications Various pattern recognition methods for time series are discussed and reviewed in this study. More emphasis is given on Dynamic Time warping algorithm. Emphasis is also placed on data preparation and various data preparation techniques as it is important to prepare and clean the data based on the requirement of the model.

14.6 Design und Implementierung einer Datenbank für das Life Cycle Management vom Windenergieanlagen

Bachelorthesis von Jonas Gerth (Abstract)

Das Kieler Unternehmen CMC steht mit ihrem Informationsmanagement vor immer mehr Herausforderungen, wie auch andere kleine und mittelständische Unternehmen. In der Thesis wurden verschiedene Programme mit dem Ziel untersucht, am Ende das Informationsmanagement von CMC zu verbessern. Angefangen wurde mit einer Untersuchung von Excel und anderen vorherrschenden Technologien im Unternehmen. Dabei kamen wir zu dem Schluss, dass sich mit dem Einsatz von Excel viele Herausforderungen nicht mehr vermeiden lassen.

Daraufhin haben wir verschiedene andere Systeme betrachtet, darunter waren Dokumentenmanagementsysteme, relationale und nicht relationale Datenbanken sowie Big Data Systeme. Die ausführliche Analyse von den Systemen brachte die Eigenschaften der Systeme und die daraus resultierende Use Cases der einzelnen Technologien hervor. Unternehmen, darunter auch CMC, haben im Informationsmanagement Use Cases, die auf einige der vorgestellten Technologien passen, wie zum Beispiel von relationalen Datenbanken, Graph Datenbanken oder Dokumenten Datenbanken. Jedoch scheitert die Einführung der Managementsysteme häufig an dem fehlenden Know-how oder den finanziellen Möglichkeiten von Unternehmen in dieser Größenordnung.

Am Ende haben wir uns entschlossen, eine relationale Datenbank auf Basis von SQL Server und einen Anwendungsserver basierend auf ASP.Net für CMC zu implementieren, die sich in ein bestehendes System von CMC integrieren lassen. Diese Anwendung wird später das primäre System zum Informationsmanagement für CMC sein.

14.7 Design and Implementation of Time Series Analysis Tool for Wind Energy Systems using Structural Pattern Matching: Data Reduction and Representation using SAX

Masterthesis von Daniel Christopher Vallavaraj (Abstract)

The problem of efficiently and accurately locating patterns of interest in massive time series data sets is an important and non-trivial problem in a wide variety of applications, including diagnosis and monitoring of complex systems, biomedical data analysis, and exploratory data analysis in scientific and business time series. In this paper a probabilistic approach is taken to this problem. The problem of finding patterns of interest in time series databases (query by content) is an important one, with applications in virtually every field of science. A variety of approaches have been suggested. Similarity searching is not a trivial problem. The two primary difficulties are time complexity and defining a similarity measure.

Time series account for a large amount of data stored in databases. A common task with a time series database is to look for an occurrence of a particular pattern within a longer sequence. Such queries have obvious applications in many fields, such as:

• Identifying patterns associated with growth in stock prices.

• Detecting anomalies in an online monitoring system.

In this paper, an efficient way of finding a pattern in the time series is proposed. The main contribution is the comparison of existing algorithms for pattern matching in time series and find the performance efficiency based on time and spatial complexity.

A new symbolic representation of time series is used. The representation is unique in that it allows dimensionality/numerosity reduction, and it also allows distance measures to be defined on the symbolic approach that lower bound corresponding distance measures defined on the original series. SAX allows a time series of arbitrary length n to be reduced to a string of arbitrary length w, ($w < n$, typically $w << n$). This approach has successfully implemented in classification and clustering of time series.

This thesis paper is part of the project that is being done at Fachhochschule Kiel with the association of CMC GmbH. The project mainly deals with implementing a predictive maintenance for wind mills under their Conditional Monitoring System(CMS). One of the user requirements in the project is to identify, if a particular trend in the time series has been encountered in the past, and another one is the prediction of the time series of drive train component's frequency. In this thesis paper, both the user stories have been combined. One of the major challenges faced

by the company is the inability to save all the patterns that have aroused during a malfunction.

14.8 Entwicklung und prototypische Umsetzung einer Architektur für ein Datenanalysesystem zum Monitoring von Windenergieanlagen

Bachelorthesis von Jana K. Kröger (Abstract)

Im Rahmen der Arbeit sollen Anforderungen an System zum Monitoring von Windenergiesystemen ermittelt werden, um auf dieser Basis prototypisch eine System-Architektur zu entwickeln. Im Einzelnen sollen Sensordaten an Windenergie-anlagen ausgewertet werden, um Rückschlüsse auf die Entwicklung bestimmter Schadenverläufe zu erhalten. Durch die Analyse von verschiedenen Entwicklungen soll eine bessere Informationsbasis für eine vorbeugende Instandhaltung erreicht werden und damit auch die Mitarbeiter, die für die Überwachung zuständig sind, in ihren Entscheidungen unterstützen.

Ziel dieser Arbeit ist der Entwurf einer möglichen Softwarearchitektur, die in der Lage ist die funktionalen wie nicht-funktionalen Anforderungen an das System zu erfüllen.

Mithilfe der vorhandenen Anforderungen und Randbedingungen werden mögliche Systemkomponenten und deren Schnittstellen spezifiziert und eine Strukturierung der einzelnen Komponenten untereinander vorgenommen.

14.9 Stammdatenbank Tabelle

Anlagentyp	
Attribut	Beschreibung
Id	Identifikationsnummer
Name	Name des Typs
Nennleistung	Leistung in Kilo Watt
Rotordurchmesser	Durchmesser in Metern
Narbenhöhe	Höhe der Anlage in Metern
Regelungskonzept	Pitch oder Stall
HerstellerAktuellId	Verweis auf die Hersteller Tabelle für den aktuellen Hersteller
AerstellerAltId	Verweis auf die Hersteller Tabelle für den alten Hersteller

Ansprechspartner	
Attribut	Beschreibung
Id	Identifikationsnummer
Name	Nachname
Vorname	Vorname
Titel	Titel des Ansprechpartners
Geschlecht	Mann, Frau
E-Mail	E-Mail-Adresse
Position	Position im Unternehmen
VertragspartnerId	Verweis auf die Vertragspartner Tabelle auf das Unternehmen

Dokumente	
Attribut	Beschreibung
Id	Identifikationsnummer
Name	Name der Datei
Datei	Speicher für eine Datei Fotos oder Berichte
Datum	Erstellungsdatum des Eintrags
WindenergieanlagenId	Verweis auf die Windenergieanlagen Tabelle

Generator	
Attribut	Beschreibung
Id	Identifikationsnummer
Synchron	Gibt an, ob der Generator synchron oder asynchron läuft
KomponententypId	Verweis auf die Komponententyp Tabelle

Getriebe	
Attribut	Beschreibung
Id	Identifikationsnummer
Konzept	Konzept des Getriebes
Übersetzungsverhältnis	Übersetzungsverhältnis vom Getriebe
KomponententypId	Verweis auf die Komponententyp Tabelle

Hauptlager	
Attribut	Beschreibung
Id	Identifikationsnummer
KomponententypId	Verweis auf die Komponententyp Tabelle

Hersteller	
Attribut	Beschreibung
Id	Identifikationsnummer
Name	Name des Herstellers
produziertGenerator	Angabe, ob der Hersteller Generatoren produziert
produziertGetriebe	Angabe, ob der Hersteller Getriebe produziert
produziertHauptlager	Angabe, ob der Hersteller Hauptlager produziert
produziertWindenergieanlagen	Angabe, ob der Hersteller Windenergieanlagen produziert
produziertTeilkomponenten	Angabe, ob der Hersteller Weilkomponenten produziert

KinematikGetriebe	
Attribut	Beschreibung
Id	Identifikationsnummer
Innenring	Frequenz des Innenrings in Herz
Außenring	Frequenz des Außenrings in Herz
Drehfrequenz	Drehfrequenz in Herz
Wälzkörper	Frequenz des Wälzkörpers in Herz
Käfig	Frequenz des Käfigs in Herz
TeilkomponentenId	Verweis auf Teilkomponenten Tabelle
KinematikLagerUndGenerator	
Attribut	Beschreibung
Id	Identifikationsnummer
Innenring	Frequenz des Innenrings in Herz
Außenring	Frequenz des Außenrings in Herz
Drehfrequenz	Drehfrequenz in Herz
Wälzkörper	Frequenz des Wälzkörpers in Herz
WK-Satz	Frequenz des WK-Satz in Herz
TeilkomponentenId	Verweis auf Teilkomponenten Tabelle
KinematikStufen	
Attribut	Beschreibung
Id	Identifikationsnummer
AnzahlStufen	Anzahl der Stufen
TeilkomponentenId	Verweis auf Teilkomponenten Tabelle
Komponente	
Attribut	Beschreibung
Id	Identifikationsnummer
Name	Name der Komponente
Seriennummer	Seriennummer der Komponente
Aktiv	Wert, ob die Komponente aktiv ist
Baujahr	Baujahr der Komponente
KomponententypId	Verweis auf Komponententyp Tabelle
WindenergieanlagenId	Verweis auf Windenergieanlagen Tabelle
Komponententyp	
Attribut	Beschreibung
Id	Identifikationsnummer
Name	Name des Typs
HerstellerAktuellId	Verweis auf die Hersteller Tabelle, für den aktuellen Hersteller
HerstellerAltId	Verweis auf die Hersteller Tabelle, für den alten Hersteller
Teilkomponente	
Attribut	Beschreibung
Id	Identifikationsnummer
Typ	Typ der Teilkomponente
Seriennummer	Seriennummer der Teilkomponente, falls vorhanden
Position	Position in der Komponente
HerstellerId	Verweis auf die Hersteller Tabelle
KomponentenId	Verweis auf die Komponenten Tabelle
Vertragspartner	
Attribut	Beschreibung
Id	Identifikationsnummer
Name	Firmenname
Website	Website der Firma
Land	Land des Firmensitzes
Bundesland	Bundesland des Firmensitzes
Ort	Ort des Firmensitzes
PLZ	Postleitzahl vom Ort

Straße	Straße des Firmensitzes
Hausnummer	Hausnummer des Firmensitzes
Telefonnummer	Telefonnummer der Firma
Windenergieanlage	
Attribut	Beschreibung
Id	Identifikationsnummer
Seriennummer	Seriennummer der Anlage
AlteSeriennummer	Alte Seriennummer der Anlage
Aktiv	Wert, ob die Anlage noch Überwacht wird
AnlagennummerimWindpark	Nummer der Anlage im Windpark
IbnDatum	Erster Tag der Überwachung
Steighilfe	Steighilfe der Anlage (z.B. Leiter, Aufzug)
Lagerungskonzept	Konzept des Antriebsstrangs
AnlagentypId	Verweis auf die Anlagentyp Tabelle
WindparkId	Verweis auf dir Windpark Tabelle
VertragspartnerId	Verweis auf die Vertragspartner Tabelle auf das Unternehmen
Windpark	
Attribut	Beschreibung
Id	Identifikationsnummer
Name	Nachname
Land	Land des Windparks
Bundesland	Bundesland des Windparks
Ort	Ort des Windparks
PLZ	Postleitzahl vom Ort
Breitengrad	Breitengrad vom Windpark
Längengrad	Längengrad vom Windpark
Aktiv	Wert, ob der Park noch überwacht wird
HotelId	Verweis auf die Hotel Tabelle, für das nächste Hotel bei Wartungsarbeiten