

# Datengetriebene Leistungserfassung bei Bergbaumaschinen zur Effizienzsteigerung

Prof. Dr.-Ing. Dipl.-Wirt. Ing. Andreas Merchiers, Bochum University of Applied Sciences, Bochum, Deutschland

Prof. Dr. rer. nat. Henrik Blunck, Bochum University of Applied Sciences, Bochum, Deutschland

Arne Köller, M.Sc., Institute for Advanced Mining Technologies, RWTH Aachen University, Aachen, Deutschland

Dr.-Ing. Christian Gierga, Eickhoff Bergbautechnik GmbH, Bochum, Deutschland

## 1 Data Science im industriellen Umfeld

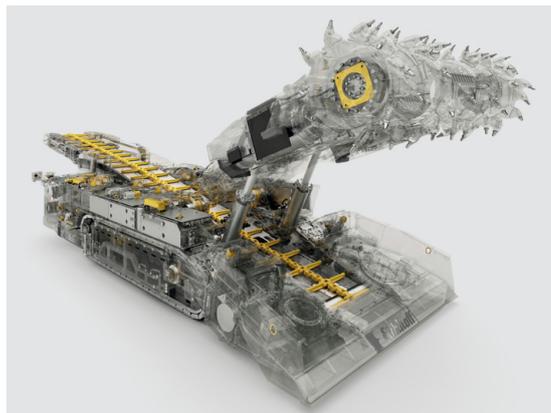
Die Unternehmen fast aller Industriezweige werden seit einigen Jahren erheblich gefordert, in puncto Digitalisierung ihre Hausaufgaben zu machen. Die stetig zunehmende Automatisierung und Ausstattung der Betriebsmittel mit Sensorik und Aktorik sowie immer umfassendere Systemintegrationen führen dazu, dass heute große Mengen von Maschinen- und Prozessdaten existieren – sowohl auf der Maschinenhersteller- als auch auf der Anwenderseite.

Die Maschinenhersteller haben diese Entwicklung als Chance begriffen, ihre Wettbewerbspositionen durch zusätzliche, digitale Leistungsangebote auszubauen. Zielsetzungen dabei sind vornehmlich, die Prozessbeherrschung auf Anwenderseite zu optimieren und gleichzeitig mittels Condition Monitoring Einsparpotenziale durch zustandsorientierte Instandhaltungsstrategien zu heben. Darüber hinaus birgt die Auswertung der gewonnenen Felddaten aber auch großes Potenzial für die eigene Organisation. Die Entwicklung und Konstruktion erhält dabei einen viel besseren Kenntnisstand über die genauen Einsatzbedingungen und das Nutzungsverhalten der Maschinen (**Bild 1**) – valide Datenreihen statt Multimomentaufnahmen gepaart mit proprietärem Erfahrungswissen im Team. Leistungsmerkmale und Funktionalitäten von Maschinen lassen sich auf dieser Basis viel zielgerichteter anpassen und optimieren und ermöglichen es, die Bevorratungs- und Wartungsstrategie kunden- bzw. maschinenspezifisch auszulegen.

Die Industrie hat verstanden, dass die nachhaltige Stärkung der eigenen Wettbewerbsposition – sowohl in der Produktion als auch bei den Betriebsmittelherstellern im Bereich der Datenanalyse liegt und zwar über die eigenen Systemgrenzen hinaus mit Schnittstellen hinein in die peripheren Systeme. Doch das tatsächliche Anforderungsspektrum ist heute noch sehr diffus: Den einen reicht die bloße Visualisierung von Sensordaten mit einem Fokus auf Verlässlichkeit und Validität der Daten, andere sehen hier den Schlüssel dazu, Prozessstabilität und Reproduzierbarkeit unabhängig von der tatsächlichen Kenntnis der Wirkzusammenhänge in komplexen Systemen zu erzielen. Allen Intentionen gemein ist dabei allerdings die Zielsetzung, über die heuristischen Ansätze der Lean-Welt hinausgehend zu evidenzbasierten Vorschlägen auf Basis statistischer Modelle und Auswertungen zu gelangen.

Für Hersteller und Anwender von Bergbaumaschinen für den untertägigen Einsatz ergeben sich große Chancen aus der Weiterentwicklung von Sensorik und Aktorik. Die Erfassung und Analyse von Sensordaten ermöglicht Prozessoptimierungen im Bergwerksbetrieb, aber auch bei der Produktion der Maschinen und damit erhebliches Einsparpotenzial. Der Artikel erläutert die Möglichkeiten und besonderen Herausforderungen des Data Mining Prozesses, manuell-regelbasierter Modellierungen, der Datenarchitektur, der Datenvisualisierung, statistischer Analysen und des Machine Learning (ML). Fallbeispiele für Walzenlader und Continuous Miner der Eickhoff Gruppe zeigen den Wissensgewinn und die Chancen auf.

**Bergbau • Digitalisierung • Data Science • Maschinen • Prozessoptimierung • Einsparung**



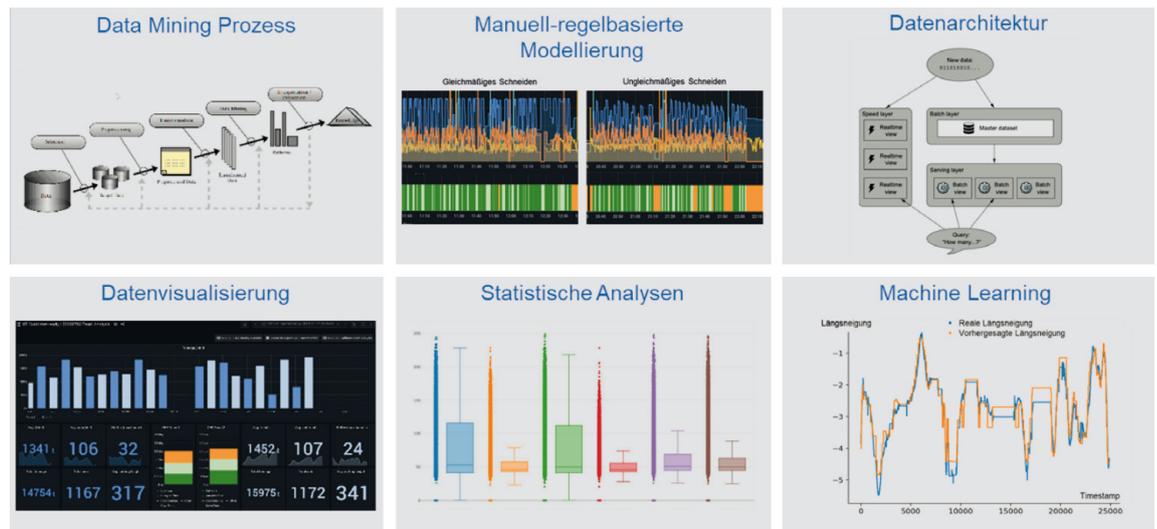
**Bild 1:** Bergbaumaschine vom Typ Continuous Miner der Eickhoff Bergbautechnik GmbH

Foto: Godehardt

Grundvoraussetzung für den Erfolg ist damit in allen Fällen die Beherrschung der typischen Schritte und Aspekte einer Datenpipeline – beginnend mit der Sensordatenentnahme, über die Datenpufferung, -speicherung und -prozessierung bis hin zur Visualisierung.

## 2 Data Science im Kohlenbergbau

Für den in diesem Artikel fokussierten Anwendungsbereich des untertägigen Bergbaus – einerseits den Strebbau mit Walzenladern und andererseits den Kammer-Pfeiler-Bau mit Continuous Minern – sind ähnliche Entwicklungen wie in anderen Industrien zu verzeichnen. Das Potenzial für den Ausbau des Leistungsspektrums und bestehender Geschäftsmodelle ist enorm. Die untertägig eingesetzten Maschinen sind zumeist mit zahlreichen Sensoren ausgerüstet, die die



**Bild 2:** Data-Science-Algorithmen und -Verfahren für untertägige Bergbaumaschinen der Eickhoff Gruppe

Informationsgrundlagen für die Maschinensteuerung und -regelung liefern. Darüber hinaus können die von den Sensoren erfassten Daten persistiert werden, etwa um späteren Analysen unterzogen zu werden. Die gewonnenen Sensordaten sind durch die immer besser ausgebaute IIoT-Infrastruktur in den Bergwerken übergeordneten Informationssystemen zugänglich und können so zur Machine-To-Machine-Kommunikation [1], SCADA-Visualisierung (SCADA für: Supervisory Control And Data Acquisition) oder bergwerksexternen Speicherung und Verarbeitung [2] genutzt werden. Fließen die Daten mehrerer Maschinen des gleichen oder unterschiedlichen Typs in ein Informationssystem zur maschinenübergreifenden Speicherung und Verarbeitung, lassen sich zudem bestehende Hypothesen und datenanalytische Modelle an gesamten Flotten gegenprüfen und bergbauliche Teil- oder Gesamtsysteme quantitativ bewerten. Letzteres kann operative und strategische Optimierungsentscheidungen der Bergwerksbetreiber und der Maschinenhersteller maßgeblich beeinflussen.

Anhand konkreter, in die Praxis umgesetzter Beispiele aus der Eickhoff Gruppe werden im Weiteren die aktuellen Möglichkeiten von Data-Science-Algorithmen und -Verfahren bei Bergbaumaschinen sowie die zu deren Operationalisierung erforderlichen infrastrukturellen Voraussetzungen vorgestellt (Bild 2).

Dabei teilt sich die Operationalisierung von Data-Science-Algorithmen und -Verfahren in die algorithmische Behandlung der Daten zum Zweck des Informations- und Wissenserwerbs und die Schaffung infrastruktureller Rahmenbedingungen zum Zweck der systematischen und kontinuierlichen Integration dieser Algorithmen in die Produkte und Prozesse eines bergbaulichen Maschinenherstellers auf. Hierzu wird zunächst im Abschnitt 3 am Beispiel der Detektion von Gewinnungszyklen für Continuous Miner und der sich darauf aufbauenden Berechnung des Abbauvolumens die algorithmische Behandlung von Daten, also der Data Mining Prozess inkl. Modellierung eines ma-

nuell-regelbasierten Systems, dargestellt. Anschließend wird im Abschnitt 4 exemplarisch eine infrastrukturelle Datenarchitektur vorgestellt, die die effiziente Berechnung arbiträrer Algorithmen auf arbiträren Daten ermöglicht. Auf dieser Basis werden im Abschnitt 5 abschließend weitere Beispiele für Anwendungsszenarien von Data-Science-Algorithmen und -Verfahren aufgezeigt. Abschnitt 6 enthält ein Fazit mit Ausblick.

### 3 Behandlung von Sensordaten am Beispiel der Gewinnungszyklen und Abbauvolumina für Continuous Miner

Beispielhaft für die Implementierung von Data-Science-Algorithmen und -Verfahren wird in diesem Abschnitt zunächst die algorithmische Behandlung von Sensordaten eines Continuous Miner zur Detektion von Gewinnungszyklen und zur Berechnung des Abbauvolumens dargestellt. Der Schneid- bzw. Gewinnungszyklus, besteht idealisiert betrachtet aus den folgenden für die Zielsetzung der Zyklererkennung abstrahierten Einzelschritten (Bild 3):

- 1 Vorfahren an den Stoß
- 2 Einschneiden in die Ortsbrust an der Firste
- 3 Senken des Schneidauslegers/Abwärtsschnitt
- 4 Einschneiden in die Sohle und Zurücksetzen mit gesenktem Schneidausleger und Sauberladen der Sohle

Die Einzelschritte des Gewinnungszyklus können je nach Bergwerk und Lagerstättengegebenheit variieren [3, S.22f, 56f]. Die idealisierte Abfolge der einzelnen Schritte des Gewinnungszyklus ist in den tatsächlichen Positionsdaten des Schneidauslegers aber zumeist nicht festzustellen, weshalb statische Schwellwert- und Regelsysteme um Methodiken aus dem Bereich der Data-Science-Verfahren ergänzt werden müssen. Hierunter fallen z. B., wie später dargestellt, explorativ-visuelle Datenanalysen. Ist der Gewinnungszyklus erkannt und bestimmt, können nachfolgend darauf basierend Leis-

tungsmetriken zur Zyklusdauer sowie zum Abbauvolumen berechnet werden.

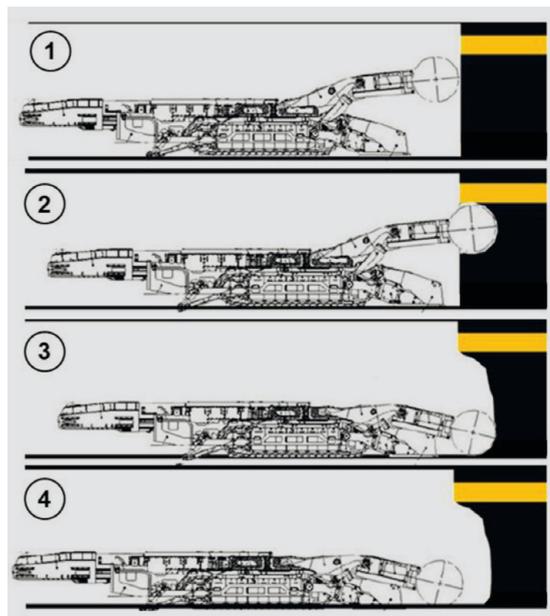
Das operativ-algorithmische Vorgehen zur Detektion von Gewinnungszyklen und Berechnung des Abbauvolumens entspricht typischerweise dem im **Bild 4** dargestellten Ablaufdiagramm mit folgenden Schritten:

- ▶ Selektion
- ▶ Datenvorverarbeitung
- ▶ Transformation
- ▶ Data Mining
- ▶ Interpretation/Bewertung

Auch im konkreten Anwendungsbeispiel wird das im **Bild 3** dargestellte Ablaufdiagramm komplett durchlaufen. Datenquelle ist eine relationale Datenbank, in die über ein Echtzeitprotokoll als Kommunikationsschnittstelle die Sensordaten des Continuous Miner fließen. Die **Selektion** der für den Anwendungsfall notwendigen Daten ist damit per SQL-Abfragesprache möglich. Zur Zyklendetektion und Abbaumengenberechnung sind insbesondere Sensordaten zu den Motorströmen und der Zylinderposition des Schneidauslegers zu nutzen. Weiterhin können im Schritt der **Datenvorverarbeitung** Schwellwerte der Motorströme zu Motorstart und -abschaltung festgelegt werden, um Datenausreißer erkennen und entsprechend behandeln zu können. Eine **Transformation** der Daten ist im vorliegenden Anwendungsfall nicht nötig.

**Data Mining** ist im Kontext dieser Arbeit als konkret-spezifischer Vorgang zum Erwerb von Informationen und Wissen aus vorverarbeiteten Daten definiert. Der Vorgang des Data Minings nutzt Data Science-Algorithmen und -Verfahren, um aus den Daten Wissen zu generieren [5, S.24]. Konkret stehen hierbei folgende Verfahren zur Auswahl:

- ▶ **Klassifikation:** Verfahren zur Vorhersage einer kategorialen Variablen auf Basis der Sensordaten
- ▶ **Regression:** Verfahren zur Vorhersage einer numerischen Variablen auf Basis der Sensordaten

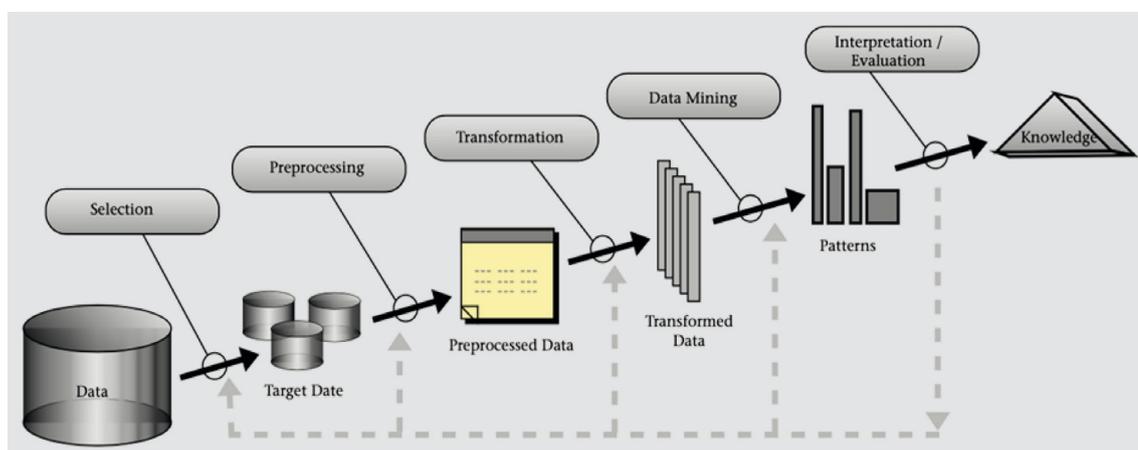


**Bild 3:** Vereinfachter, auf Zykluserkennung fokussierter Gewinnungszyklus eines Continuous Miner

- ▶ **Clustering:** Verfahren zur Aufteilung der Gesamtdaten in Gruppen mit jeweils ähnlichen Merkmalsausprägungen

Im Kontext lernender Algorithmen, Machine Learning genannt, werden Klassifikations- und Regressionsverfahren als ‚supervised‘ Verfahren bezeichnet, während Clustering-Verfahren als ‚unsupervised‘ Verfahren betitelt werden [5, S.49 ff]. Im Kontext nichtlernender Algorithmen können alle drei oben aufgezählten Verfahren als manuell-regelbasierte Verfahren aufgefasst werden. Sowohl im Kontext lernender und nichtlernender Algorithmen werden Modelle über die Sensordaten gebildet. Diese Modelle haben der obigen Aufzählung entsprechend die Aufgabe der Vorhersage einer Variablen (Klassifikation und Regression) oder der Aufteilung der Daten in ähnliche Gruppen (Clustering).

Im vorliegenden Anwendungsfall wird ein manuell erstelltes, regelbasiertes Klassifikationsmodell für die



**Bild 4:** Ablaufdiagramm der algorithmischen Behandlung von Daten zum Erwerb von Wissen [4]



**Bild 5:** Gewinnungszyklen eines Continuous Miner in zwei temporalen Zoom-Stufen

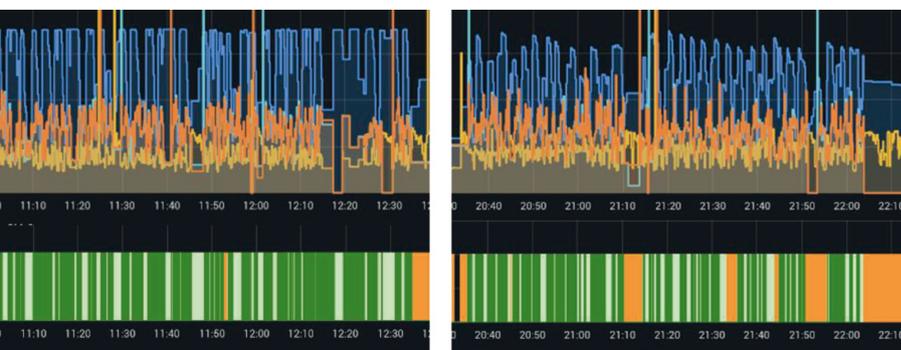
Die blaue Kurve stellt die Höhenposition des Schneidauslegers über die Zeit dar. Die orangen und türkisen Kurven stellen die Schneidmotorströme und die gelbe Kurve den Motorstrom des Hydraulikmotors dar. Das Balkendiagramm (unten) stellt im Zeitraum Schneidphasen (grün), Schneidpräparationsphasen (hellgrün), und übrigen (orange), wie etwa floor-cleaning-Phasen dar.

Detektion der Einzelschritte des Gewinnungszyklus ausgewählt. Grund dieser Auswahl ist, dass die Einzelschritte im Gewinnungszyklus mit einer eher kleinen und durch Prozess-Know-how sehr selektiv bestimmten Menge an Regeln ausreichend genau beschrieben werden können. Zur Bestimmung der Menge an Regeln wurde zunächst eine explorative Datenvisualisierung durchgeführt, deren Ergebnis im Folgenden erklärt wird. Im **Bild 5** ist eine Sequenz mehrerer Gewinnungszyklen über einen längeren Zeitraum dargestellt. Die blaue Kurve stellt die Höhenposition des Hydraulikzylinders des Schneidauslegers dar. Gut zu erkennen sind dabei die individuellen Gewinnungszyklen und insbesondere diejenigen Zeitpunkte, an denen der Continuous Miner den Schneidausleger zum Abwärtschnitt absenkt, da dort auch die im Bild orange dargestellten Schneidmotorströme höhere Amplituden und Varianzen aufweisen. Aus **Bild 5** wird aber auch deutlich, dass die realen Gewinnungszyklen nicht vollstän-

dig der idealisierten Beschreibung entsprechen und ein uneinheitlicheres Gesamtbild abgeben. Durch individuelle geologische Gegebenheiten oder die Fahrweise des Maschinenbedieners können sowohl die maximale als auch die minimale Schneidarmhöhe unterschiedlich sein und bei aufeinanderfolgenden Gewinnungszyklen stark differieren. Auch wird aus den Sensordaten ersichtlich, dass der Abwärtschnitt in verschiedenen Geschwindigkeiten und mit verschiedenen großen Pausen erfolgen kann. Darüber hinaus können vor einem erneuten Einschnitt an der Firste Säuberungsschnitte erfolgen.

Während diese Abweichungen vom idealisierten Gewinnungszyklus die Detektion der einzelnen Schritte im Gewinnungszyklus erschweren, können sie auch durchaus wertvolle Informationen liefern, beispielsweise für das Fahrtraining der Maschinenbediener. Dazu kann mittels einer Gewinnungszyklusdetektion das Fahrprofil verschiedener Maschinenbediener quantifiziert werden – unter Zuhilfenahme bekannter Abbau-mengen in den betrachteten Schichten.

Anhand der Visualisierung der Schneidauslegerhöhe werden, wie im **Bild 6** dargestellt, Differenzen zwischen den eingesetzten Gruppen/Teams von Maschinenbedienern deutlich. Die Schnitte der einen Gruppe von Maschinenbedienern haben sehr gleichmäßige maximale Auslegerhöhen, während die Daten einer anderen Gruppe ein ungleichmäßiges Verhalten aufweisen. Allerdings sind die Gewinnungszyklen im ungleichmäßigen Fall gegenüber dem gleichmäßigen Fall durchschnittlich kürzer, sodass die bloße Erkenntnis dieser Unterschiede noch keine Bewertung zulässt und letztendlich lediglich die Korrelation zwischen Maschinenbedienern und Schneidverhalten bestätigt, da diese



**Bild 6:** Vergleichende visuelle Datenanalyse zwischen Maschinenbedienergruppen beim Continuous Miner – Verlauf der Schneidauslegerhöhen für gleichmäßiges und ungleichmäßiges Schneiden

Unterschiede im konkreten Fall reproduzierbar unter gleichen Bedingungen auftreten.

Das vordringliche Ziel einer Gewinnungszyklusdetektion ist aber das Zählen der Gewinnungszyklen und damit die Performanceevaluation des eingesetzten Continuous Miner sowie das Quantifizieren des Abbauvolumens unter Einbezug der bekannten Schneidwalzengeometrie und Einschnitttiefe. Diese Ziele lassen sich trotz unregelmäßigen Schneidens erreichen, sofern der Beginn und das Ende eines Gewinnungszyklus korrekt und Säuberungsschritte nicht fehlerhaft als neue Schneidzyklen erkannt werden, wie in der stilisierten Darstellung im **Bild 7** grafisch erläutert. Darin sind in der blauen Kurve die Position des Schneidauslegers und in der roten Kurve der Schneidmotorstrom dargestellt. Zu erkennen sind zwei Gewinnungszyklen. Die Bewegungen zum Ende des zweiten Zyklus werden nicht fehlerhaft als neuer Gewinnungszyklus erkannt. Aus den Erkenntnissen dieser explorativen Datenvisualisierung kann ein regelbasiertes Klassifikationsmodell erstellt werden, das die Gewinnungszyklen des Continuous Miner korrekt bestimmt. Für die darauf basierende approximierte Berechnung des Abbauvolumens können mehrere Ansätze gewählt werden. So kann z. B. während einer Schicht die Anzahl der erkannten Gewinnungszyklen mit einem empirisch ermittelten, durchschnittlichen Abbauvolumen pro Zyklus multipliziert werden. Alternativ oder zusätzlich kann das Abbauvolumen über die Schneidhöhe, Einschnitttiefe der Walze und Walzenbreite für jeden Gewinnungszyklus berechnet werden. Zudem kann auch der Schneidmotorstrom einbezogen werden, um eventuelle Leerfahrten oder -schnitte aus der Berechnung zu nehmen.

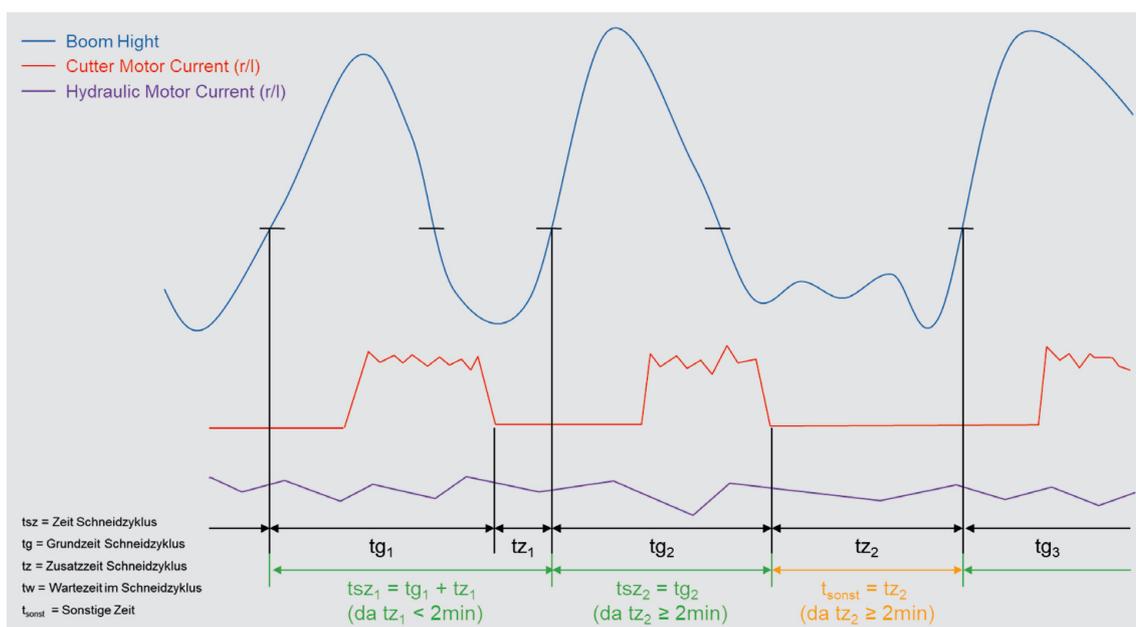
Für die genaue, verlässliche und robuste Berechnung der primären Zielgrößen, insbesondere der Anzahl der Schneidzyklen und der Tonnage, bedarf es – wie darge-

stellt – des Handlings vieler Unregelmäßigkeiten, verschiedener Fahrstile und örtlicher Gegebenheiten. Hierfür werden Data-Science-Algorithmen und -Verfahren zu Hilfe genommen, die zunächst mit realen Fahrdaten trainiert werden, bei denen idealerweise bereits viele der obigen Variationen zu beobachten sind. Mit weiteren Realdaten wird die Erkennung von Gewinnungszyklen und Abbauvolumen auf globale Genauigkeit sowie auf unerwünschte lokale Artefakte getestet.

Eine hohe Genauigkeit bei Erkennung der Zielgrößen wird insbesondere dann erreicht, wenn die Variationen zu unterschiedlichen Ergebnissen für die Erkennung relevanter Kenngrößen führen, wie etwa typischer maximaler und minimaler Schneidauslegerhöhen, typischer Zykluszeiten und typischer horizontaler Fahrstrecken eines Zyklus. Maßgebend hierfür ist ebenfalls die Sammlung und Interpretation der im realen Einsatz gewonnenen „Big Data“, d. h. der erfassten Sensordaten.

#### 4 Datenarchitektur und erforderliche IT-Infrastruktur im Bergbau

Die Operationalisierung von Data-Science-Algorithmen und -Verfahren für die im vorherigen Abschnitt an einem Beispiel beschriebene analytischen Behandlung von Daten erfordert zuvor die Schaffung infrastruktureller IT-Systeme, mit denen die Sensordaten gespeichert, prozessiert und für die Visualisierung oder Weiterverwendung angeboten werden. Das stellt bergbauliche Maschinenhersteller vor die Herausforderung einer systematischen Integration derartiger Algorithmen und Verfahren in eine schon bestehende, erst teilintegrierte oder noch komplett zu schaffende Dateninfrastruktur. Als eine Dateninfrastruktur wird in diesem Artikel die Gesamtmenge aller datenverarbeitenden Systeme zur Datenübertragung, -speiche-



**Bild 7:** Identifikation von Gewinnungszyklen trotz Unregelmäßigkeiten

zung, -prozessierung und -visualisierung definiert. Beispielsweise werden zur Speicherung der Sensordaten persistierende Datenbanken, wie etwa Datenbanken, benötigt, die in eine kohärente Systemlandschaft eingepflegt werden müssen. Der Zugriff auf diese Datenbanken soll sich weiterhin als einheitlich, ausfallsicher und responsiv für nachgelagerte Datenprozessierungs- oder Datenvisualisierungssysteme darstellen [6, S.7 ff.]. Insbesondere für bergbauliche Maschinenhersteller ist eine Zwischenpufferung der Daten bei ihrem "Fluss" durch verschiedene Teilsysteme der Dateninfrastruktur unerlässlich. Hintergrund sind stark latenzbehaftete Kommunikationswege zwischen Datenursprung auf der Maschine und gegebenenfalls räumlich entfernten Datenspeicherungen und -prozessierungen. Die Komplexität einer Dateninfrastruktur wird dabei maßgeblich durch die Anforderungen, die an sie gestellt werden, bemessen und bestimmt. Dabei kann je nach Anwendungsfall und den sich daraus ergebenden Anforderungen bereits eine teilentegrative Dateninfrastruktur pragmatischen Zielen, die oftmals Anlass für den Einsatz von Data-Science-Verfahren sind, genügen.

Diese erste Beschreibung der Anforderungen an eine Dateninfrastruktur offenbart in jedem Fall bereits die software-architektonische Komplexität, der sich Maschinenhersteller, insbesondere vor dem Hintergrund von Big Data, im Kontext einer systematischen Integration von Data-Science-Algorithmen und -Verfahren stellen müssen. Ziel dieses Abschnitts ist daher die beispielhafte Darstellung einer solchen Dateninfrastruktur unter Berücksichtigung der speziellen Anforderungen im Bergbau. Hierfür wird zunächst der bereits erwähnte Begriff Big Data erläutert.

#### 4.1 Was ist Big Data?

Der Begriff Big Data beschreibt das Phänomen des starken Anstiegs von Datenmengen hinsichtlich der benötigten Speicherressourcen, der Generierungs-, Übertragungs- und Verarbeitungsgeschwindigkeit sowie der Inhomogenität von Datenformaten [7, 8]. Die zunehmende Integration von Sensoren in Maschinen führt zu einer kontinuierlichen Steigerung der zu verarbeitenden Datenmengen [9, S. 1ff.]. Die Forderung nach aufwändigen Echtzeitverarbeitungen dieses angestiegenen Datenflusses, z.B. zur Unterstützung operativer Prozesse mittels vorausschauender Maschinensteuerungen, potenziert die Herausforderung. Erschwerend kommt hinzu, dass diese Datenverarbeitungen meistens nicht von der eingebetteten Hardware auf der Maschine selber übernommen werden können. Zudem können die Sensordaten untereinander eine strukturelle Ungleichheit (hier als Inhomogenität bezeichnet) hinsichtlich ihrer (Format-)Semantik und ihres logischen Datentyps aufweisen. Eine maschinenübergreifende Dateninfrastruktur muss in ihren Charakteristika eben diesen vielen Anforderungen an große, hochfrequente und inhomogene Datensammlung genügen.

#### 4.2 Charakteristika einer Big Data Dateninfrastruktur

Eine Dateninfrastruktur hat zum Ziel, arbiträre Berechnungen auf arbiträren Datenmengen effizient auszuführen [6, S. 27, 83]. Die notwendigen Charakteristika einer Big-Data-Dateninfrastruktur lassen sich vor folgendem Hintergrund systematisch ableiten und konkretisieren:

- ▶ Große Datenmengen und damit große Speicherbedürfnisse
- ▶ Hohe Datengenerierungs-, Übertragungs-, und Verarbeitungsgeschwindigkeit
- ▶ Inhomogene Datenformate

Zunächst versuchen Big-Data-Dateninfrastrukturen große Datenmengen effizient zu verarbeiten. Dabei spielt die Lesegeschwindigkeit der Daten von einem Festspeicher eine herausragende Rolle: Ein z.B. mit fünf Sensoren bestückter Continuous Miner generiert im Takt von 250 ms je Datenpunkt insgesamt 1,73 Mio. Datenpunkte pro Tag, welche nach ihrer Übertragung in einer Dateninfrastruktur abgespeichert werden müssen. Heutige typische terabytegroße Festspeicher bieten hierfür zunächst ausreichend großen und günstigen Speicherplatz. Als das eigentliche Problem dieser großen Datenvolumen offenbart sich jedoch die Lesegeschwindigkeit dieser Festspeicher. Beträgt diese, im Falle der schnellen Solid State Disks (SSDs) als Festspeichertechnologie etwa 600 MB/s, dauert es trotzdem noch ca. 28 Minuten, den kompletten Inhalt des Festspeichers zu lesen. Dem erklärten Ziel einer Dateninfrastruktur, arbiträre Berechnungen auf arbiträren Datenmengen effizient durchführen zu können, ist damit nur unzureichend Genüge getan. Hier bietet sich als Lösung die parallele Prozessierung der Datenmengen mittels eines verteilten Systems an [9, S. 3 ff.]. Dabei wird die gesamte Datenmenge auf mehrere Festspeicher aufgeteilt, von denen gleichzeitig gelesen wird. Ist eine Datenmenge im Umfang von 1 Terabyte, also 1.000 Mrd. Bytes, auf 25 SSDs mit je 600 MB/s Lesegeschwindigkeit disjunkt aufgeteilt und wird die Datenmenge von allen SSDs gleichzeitig gelesen, dauert es nur noch ca. 1 Minute und 6 Sekunden, um die Datenmenge von 1 Terabyte einzulesen.

Ein verteiltes System besteht jedoch nicht nur aus den Rechnern bzw. Datenspeichern mit jeweils 25 SSDs, sondern benötigt auch eine Instanz, die vor- und festhält, welche Daten auf welchem Festspeicher liegen. Ebenso ist die Datenmenge nicht notwendigerweise komplett disjunkt über die Festspeicher verteilt, sondern kann, u. a. zur Ausfallsicherheit, auch redundant vorgehalten werden. Fällt ein Festspeicher aus, sind die jeweiligen Datensätze noch auf weiteren Festspeichern zu finden.

Diese Verbindungen und Abhängigkeiten zwischen den einzelnen Systemteilen – auf der einen Seite den datenspeichernden Rechnern und auf der anderen Sei-

ten die Datenverteilung koordinierenden und protokollierenden Rechnerknoten – sind die konstituierenden Elemente für eine verteilte Dateninfrastruktur. Wird mehr Speicherplatz benötigt oder soll die Resilienz gegenüber Hard- oder Softwareausfällen erhöht werden, kann dies mittels verteilter Dateninfrastrukturen erreicht werden: durch horizontale Skalierung, d.h. das Hinzufügen neuer Rechner in den Rechnerverbund, oder durch vertikale Skalierung, d.h. die meist hardwaretechnische Aufwertung im Rechnerverbund bestehender Rechner.

Darüber hinaus stellt Big Data eine Dateninfrastruktur auch vor die Herausforderung einer schnellen Verarbeitungsgeschwindigkeit im Kontext der Datenprozessierung. Ein Beispiel für einen solchen Datenprozessierungsvorgang ist die Anwendung des Klassifikationsmodells aus Abschnitt 3 auf verteilt gespeicherte Sensordaten eines oder mehrerer Continuous Miner. In der Praxis werden hierfür zumeist der MapReduce-Ansatz oder I/O-effiziente Datenprozessierungssysteme genutzt [9, S. 18f, S. 98; 10, S. 14f]. Im ersten Fall, dem MapReduce-Ansatz, werden zunächst Teilergebnisse auf Teilmengen der Gesamtdaten berechnet und später zu einem Gesamtergebnis zusammengeführt. Damit kann die Umsetzung des MapReduce-Ansatzes auch als parallele Datenprozessierung beschreiben werden. Im zweiten Fall, dem von Apache Spark verfolgten I/O-effizienten Ansatz, versucht das verteilte System, möglichst viele Berechnungen speichereinheitlich und mit hoch-performanten Speichern, wie z.B. Prozessorregistern, Prozessor-Caches oder Random Access Memories, auszuführen, um zeitintensive Festspeicheroperationen zu vermeiden. Eine ausreichend schnelle Verbindungsgeschwindigkeit kann durch moderne Telekommunikationsstandards bereitgestellt werden und bei Bedarf mit Edge-Computing-Konzepten ergänzt werden.

Final stellt Big Data schließlich auch die Anforderung an eine Dateninfrastruktur, möglichst viele domänenrelevante Datenformate und -typen zu unterstützen. Dabei kann grob in strukturierte Datenformate (z.B. relationale Datenquellen), semi-strukturierte Datenformate (z.B. XML oder JSON) und unstrukturierte Datenformate (z.B. Video- oder Audio-Dateien) unterteilt werden [9, S. 5 f.]. Gerade für das Anwendungsbeispiel eines Continuous Miner ist eine Speicherung über die als Zeitreihen zu typisierenden Sensordaten hinaus zweckmäßig, wie z.B. die Speicherung von Mikrophon-, Video- und Bilddaten oder sogar RADAR- und LIDAR-Daten.

Zusammenfassend können die Charakteristika einer Big-Data-Dateninfrastruktur wie folgt beschrieben werden [6, S. 7 ff]:

- ▶ Robust gegenüber Software- und Hardwarekorruption und fehlertolerant gegenüber menschlichem Versagen
- ▶ Geringe Leselatenz des Datenspeichers und Ad-hoc-Datenabfragen

- ▶ Horizontale Skalierbarkeit der Datenspeicher- und Datenprozessierungshardware
- ▶ Generalisierbarkeit bzgl. der Datenformate und -typen und einfache Erweiterbarkeit der Datenprozessierungsumgebung
- ▶ Geringer Pflegeaufwand des Datenspeicherungssystems durch Nutzung inhärent verteilter Systemarchitekturen
- ▶ Vereinfachte Fehlersuche und -behebung durch getrennte Speicherung der prozessierten Daten von den Rohdaten

Wie eine Big-Data-Dateninfrastruktur – etwa um das in Abschnitt 3 beschriebene Klassifikationsmodell auszuführen – mit solchen Charakteristika ausgestaltet und für den tatsächlichen Einsatz in Produkt und Unternehmen operationalisiert werden kann, soll im Weiteren konkretisiert werden.

### 4.3 Beispiel eines Big-Data-Dateninfrastruktur-Konzepts mit Lambda-Architektur

Um die verschiedenen Systeme und Werkzeuge für die Entwicklung einer Dateninfrastruktur zu einem Gesamtsystem zu integrieren, kann eine Lambda-Architektur herangezogen werden [6, S. 14], die sich software-topologisch als Schichtenmodell darstellen lässt. Dabei wird zwischen Batch-Layer, Serving-Layer und Speed-Layer als Systembestandteilen unterschieden (Bild 8).

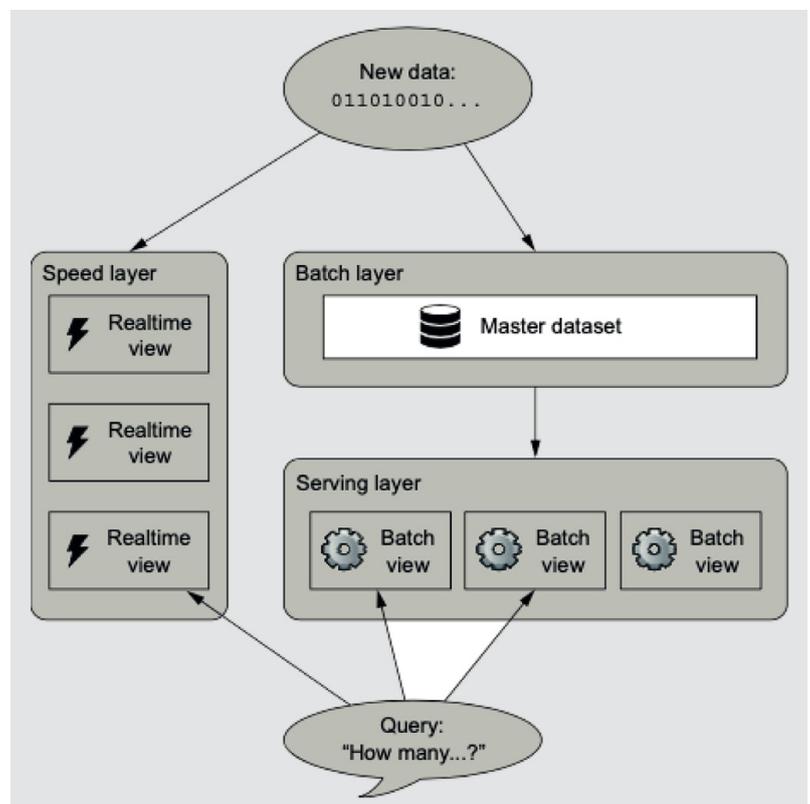


Bild 8: Lambda-Architektur als Schichtenmodell [6, S. 19]

### 4.3.1 Der Batch-Layer

Der Batch-Layer einer Lambda-Architektur speichert und prozessiert die Daten in festen Zyklen. Im Anwendungsbeispiel des Continuous Miner werden Sensordaten nach ihrer zwischengepufferten und bergwerksexternen Datenübertragung zunächst an den Batch-Layer der Lambda-Architektur übergeben und dort gespeichert. Den vorherigen Ausführungen entsprechend erfolgt die Speicherung der Sensordaten in einem verteilten System auf mehreren Rechnern mit Festspeichern, und möglicherweise redundant aufgeteilt. Dies schafft Robustheit gegen Software- und Hardwarekorruption einer Dateninfrastruktur und ermöglicht eine horizontale Systemskalierbarkeit. Ein Beispiel für ein solches verteiltes System als Batch-Layer einer Big-Data-Dateninfrastruktur ist Apache Hadoop, dessen Funktionalitäten und Merkmale im Weiteren für die Umsetzung der Lambda-Architektur herangezogen werden.

Generiert nun der Continuous Miner kontinuierlich Sensordaten und schickt diese an die Big-Data-Dateninfrastruktur, sammeln sich mit der Zeit immer größere Datenmengen im Batch-Layer an. Soll nun eine arbiträre Funktion, beispielsweise die Gewinnungszyklusdetektion aus Abschnitt 3 oder ein Machine-Learning-Modell, auf dieser Datenmenge ausgeführt werden, kann durch die parallele Prozessierung mittels MapReduce oder die I/O-effiziente Prozessierung mit dafür geeigneten Systemen die Laufzeit der Berechnung entscheidend verkürzt werden. Für ein Ergebnis in Echtzeit reicht die Verkürzung jedoch oft nicht aus.

Um auf Ad-hoc-Datenabfragen sofort Ergebnisse zurückliefern zu können, bietet sich die Teil- oder Vorberechnung der Funktion gemeinsam mit der im Serving-Layer stattfindenden Speicherung der Berechnungsergebnisse als sogenannter Batch-View an [6, S. 15]. Ein Batch-View enthält dann beispielsweise die gewünschten Ergebnisse der Gewinnungszyklusdetektion, also zu welchem Zeitpunkt sich der Continuous Miner in welchem Einzelschritt des Gewinnungszyklus befand. Ein weiterer Batch-View enthält die Ergebnisse zur Anzahl der Gewinnungszyklen pro Tag oder zum Abbauvolumen des Continuous Miner pro Minute. Die eigentlichen Sensordaten im Batch-Layer, auch Master-Datensatz genannt, bleiben bei der Berechnung eines Batch-Views unangerührt. Dies stellt einen Paradigmenwechsel im Vergleich zu inkrementellen Architekturen dar, da die Batch-Views zyklisch, z. B. alle drei Stunden, erneut berechnet werden [6, S. 9 f, S. 88ff]. Insbesondere wird so die Berechnungslogik, die die Funktion zur Batch-View-Erstellung vorschreibt, gänzlich aus dem Master-Datensatz entfernt. Enthält beispielsweise die Funktion zur Gewinnungszyklusdetektion einen Software-Bug, dann ist nur der Batch-View fehlerhaft und nicht der Master-Datensatz und kann im nächsten Batch-Zyklus mit korrigiertem Programmcode überschrieben werden. Die Interpretation des Datenformats von Sensordaten geschieht dabei idealerweise zur Laufzeit der Datenprozessierung. So können

etwa Codec-formatierte Sensordaten, wie Audio-, Bild- oder Videodaten, als Binärdatei abgespeichert werden. Durch zusätzliche Unterstützung der Speicherung von Sensordaten in reinen Textformaten ist es einer auf Apache Hadoop basierenden Dateninfrastruktur möglich, vielfältige Datenformate gleichzeitig zu unterstützen.

### 4.3.2 Der Serving-Layer

Der Serving-Layer hat die Aufgabe, dem Endnutzer, z. B. einer Datenvisualisierung oder einer API, die vom Batch-Layer berechneten Batch-Views bereitzustellen [6, S. 179]. Die Bereitstellung der Daten in den Batch-Views muss schnell, also ohne hohe Latenz geschehen. Hierfür werden im Serving-Layer Index-Datenstrukturen genutzt, die es ermöglichen, schnell bestimmte Daten innerhalb eines Datenspeichers zu finden. Die Datenspeicherung im Master-Datensatz des Batch-Layer erfolgt dagegen nicht zwangsläufig mit Indizes, da Indizes Insert-Operationen in den Datenspeicher verlangsamen können. Die Möglichkeit einer Ad-hoc-Datenabfrage in der Lambda-Architektur wird also durch die Erstellung von Batch-Views im Batch-Layer und die zur Verfügungstellung der Batch-Views im Serving-Layer erreicht.

### 4.3.3 Der Speed-Layer

Da der Batch-Layer Berechnungen auf dem Master-Datensatz nur zyklisch, z. B. alle drei Stunden, ausführt, wäre ohne eine zusätzliche Systemkomponente, die Batch- und Serving-Layer ergänzt, nicht sichergestellt, dass Berechnungsergebnisse für Ad-hoc-Datenabfragen immer für die aktuellsten Daten vorliegen. Beträgt der Berechnungszyklus des Batch-Layers z. B. drei Stunden, warten anderenfalls nachfolgende Systeme, wie die Datenvisualisierung, im schlechtesten Fall drei Stunden auf neu berechnete Daten. Für Condition-Monitoring-Systeme oder sogar Echtzeit-Empfehlungssysteme ist diese Zeitspanne in jedem Fall zu lang. Deshalb werden Batch- und Serving-Layer um einen Speed-Layer ergänzt, der inkrementell Berechnungen auf den in der Zwischenzeit anfallenden Daten ausführt.

## 4.4 Technologien und Frameworks zur Realisierung einer Dateninfrastruktur

Die beschriebene Lambda-Architektur ist nur ein mögliches Architekturmuster für eine Big-Data-Dateninfrastruktur dar. Sind die Ergebnisse der Berechnungen auf Sensordaten nicht zeitkritisch, da nicht in Echtzeit visualisiert werden soll, kann der Speed-Layer weggelassen werden. Haben die Ergebnisse der Berechnungen hingegen ausschließlich zeitkritischen Charakter, z. B. weil ein Echtzeit-Condition-Monitoring umgesetzt werden soll, können Batch- und Serving-Layer weggelassen werden.

Die Lambda-Architektur beschreibt demnach die Vereinigung dieser beiden Architekturmuster. Konkret kann für den Batch-Layer Apache Hadoop genutzt

werden. Ergänzend hierzu kann Apache Spark für eine I/O-effiziente Datenprozessierung genutzt werden. Mögliche Technologien für den Serving-Layer sind ElephantDB oder Apache HBase. Als Speed-Layer sind Apache Storm oder Apache Kafka mit entsprechenden Microservices als Prozessierungselementen denkbar.

Im folgenden Abschnitt sollen nun weitere Anwendungsbeispiele verschiedener Algorithmen für untertägige Bergbaumaschinen vorgestellt werden, die sich mittels der beschriebenen Lambda-Architektur einer Big-Data-Dateninfrastruktur operationalisieren lassen.

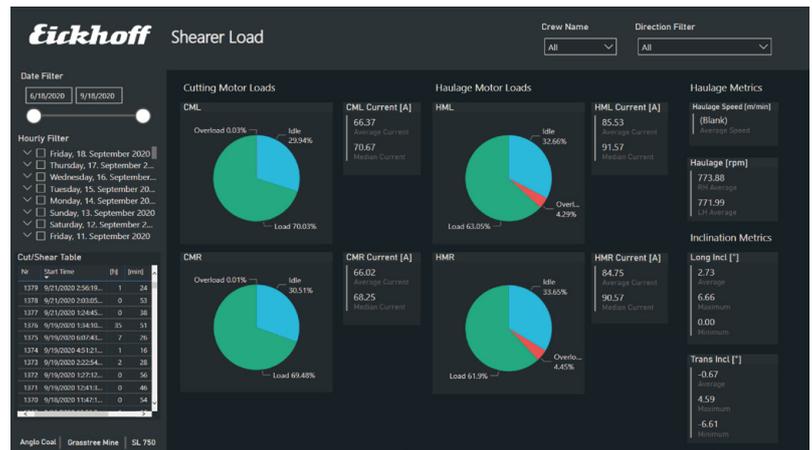
## 5 Anwendungsbeispiele aus der Eickhoff Gruppe

Das in den vorangegangenen Abschnitten beschriebene Data-Science Set-Up, also die Gesamtmenge der verwendeten Data-Science-Algorithmen und -Verfahren, kann in verschiedenen Ausbaustufen zum konkreten Nutzen der Endnutzer aufgebaut werden.

### 5.1 Ausbaustufen einer Data-Science-Infrastruktur für die Bergbau-Domäne

Die **erste Stufe** ist oft die Rohdatenvisualisierung in sogenannten Maschinendaten-Dashboards. Je nach Anwendungszweck lassen sich dazu marktübliche Lösungen für den stationären oder den web-basierten Gebrauch als Visualisierungs-Front-End nutzen. Im konkreten Fall der Eickhoff Gruppe kommen Microsoft PowerBI (**Bild 9**) sowie Grafana (**Bilder 5 und 6**) als web-basierte Nutzer-Umgebung zur Datenvisualisierung zum Einsatz. Zielsetzung der Rohdatenvisualisierung ist die Zustandserfassung wichtiger Maschinen und Prozesskenngrößen und deren Darstellung in Rohform und im zeitlichen Verlauf, sodass insbesondere Diskontinuitäten und Abweichungen vom Regelfall schnell erkannt und im Detail analysiert werden können. Entsprechend bietet sich hier vor allem die Cockpit-Darstellung an, bei der feste Ziel- oder Durchschnittswerte der Vergangenheit als Benchmark herangezogen werden – analog zum Drehzahlmesser im Auto. Mit Blick auf die untertägigen Bergbaumaschinen Walzenlader und Continuous Miner sind hier insbesondere Motorströme und damit die Belastung und zugleich die produktive Nutzung der Maschinen, Fehlerhäufigkeiten sowie allgemeine (Produktions-)Parameter, wie Wegstrecke, Laufzeit etc., zu visualisieren. Neben einem Basis-Set-Up an Kenngrößen bieten die eingesetzten Lösungen auch immer den Spielraum für kundenindividuelle Anpassungen und Auswertungen.

Für die Visualisierung von Rohdaten ist gemäß den Ausführungen in Abschnitt 4 insbesondere ein Serving-Layer notwendig, der möglichst responsiv direkten und nicht nur sequenziellen Zugriff auf die Daten ermöglicht. Hierbei ist vor allem auf die Menge der darzustellenden Daten achtzugeben. Übersteigt diese viele zehntausend Einzeldatenpunkte, was bei Betrachtungen von Sensordaten im Zeitbereich über lange Zeiträume häu-



**Bild 9:** Rohdatenvisualisierung in einem Maschinendaten-Dashboard für einen Walzenlader

fig vorkommt, beeinflusst diese Datenmenge die Darstellungsgeschwindigkeit der Datenvisualisierung für den Endnutzer. Hier kann wiederum der Batch-Layer die Rohdatenmenge in verschiedenen Batch-Views mit unterschiedlichem Detailgrad voraggrieren und bereitstellen. So kann beispielsweise ein Batch-View als Datenpunkte berechnete Durchschnitte der Sensordaten im Zeitbereich eines Tages bereitstellen, während ein weiterer Batch-View Datenpunkte mit sekundlicher Auflösung bereitstellt. Letzter wird beim starken Zoomen in einen kleinen Zeitbereich genutzt. Ersterer wird zum Hinauszoomen genutzt, um einen Überblick über ein ganzes Jahr zu erhalten, ohne hierfür Millionen von Datenpunkten explizit laden und anzeigen lassen zu müssen. Adaptiv zum gewählten Detailgrad der Visualisierung werden so die Rohdaten unterschiedlich detailliert bereitgestellt. Sollen zudem auch Echtzeitdaten visualisiert werden, empfiehlt sich die Nutzung einer datengetriebenen Aktualisierung der Visualisierungselemente. Die Rohdaten werden aktiv in die Visualisierung übergeben, sobald diese vom Speed-Layer verarbeitet wurden, und nicht als Batch-Views bereitgestellt, aus denen die Visualisierung die Rohdaten laden kann.

Eine **nächste Ausbaustufe** ist die Visualisierung berechneter Werte unter Nutzung klassisch-statistischer und manuell-regelbasierter Modelle. Die Dateninfrastruktur, die die benötigten berechneten Werte prozessiert und dem Visualisierungs-Front-End zur Verfügung stellt, muss, wie im Abschnitt 4 dargestellt, in der Lage sein, jedwede Berechnung auf den Daten auszuführen. Hierfür können beispielsweise Rohdaten-Zeitreihen verschiedener Sensoren in Beziehung zueinander gebracht werden, um Muster und Korrelationen sichtbar machen und zuvor aufgestellte Hypothesen prüfen zu können. Die mithilfe der Dateninfrastruktur berechneten Werte entsprechen in diesem Fall dem Ergebnis einer Korrelationsanalyse. Überdies stehen Fehler-Ursache-Analysen, durch Bestimmung von Abhängigkeiten bestimmter Zustände mit Fehlergründen im Fokus der Visualisierungen dieser Ausbaustufe. Ebenso dienen die Visualisierungen von Analysen zur Wahl der richtigen Instandhaltungsstrategie auf Basis des Maschinenver-

haltens im Benchmark zu Referenzmarken im Sinne eines „Machine-Fingerprints“. Hierfür sind im **Bild 10** exemplarisch die Belastungen von Schneidmotoren von Walzenladern gleichen Typs in verschiedenen Bergwerken dargestellt. Während, wie im **Bild 10** visualisiert, die Daten für die Felder bzw. Bergwerke 1 und 2 deutlich darauf hinweisen, dass unidirektional geschnitten wird, weil einer der beiden Schneidmotoren wesentlich stärker als der andere belastet wird, lassen die Daten zu Bergwerk 3 auf einen bi-direktionalen Betrieb schließen. Gleichwohl erkennt man deutlich, dass die Belastungen der jeweiligen linken Motoren (CML) der Walzenlader in den Bergwerken 1 und 2 in den Spitzen und generell häufig über den zugehörigen rechten Motoren (CMR) in demselben Bergwerk sowie über den konstanteren Belastungswerten beider Motoren der Walzenlader aus Bergwerk 3 liegen. Das sind zum einen wichtige Erkenntnisse und Informationen für den Service im Hinblick auf die Ersatzteilversorgung und zum anderen für den Bergwerksbetreiber, im **Bild 10** etwa für Bergwerk 3 im Hinblick auf das ungenutzte Maschinenpotenzial.

Analog zur Visualisierung von Sensorrohdaten werden bei der Visualisierung berechneter Werte Batch-, Serving- und Speed-Layer einer Dateninfrastruktur benötigt. Batch- und Speed-Layer sind für die Berechnungen auf den Rohdaten zuständig. Die Ergebnisse solcher Berechnungen, beispielsweise das Ergebnis einer Korrelationsanalyse oder eines statistischen Hypothesentests, werden jeweils in Batch-Views im Serving-Layer abgespeichert, sodass nachliegende Systeme, wie die Visualisierung, auf die Ergebnisdaten responsiv zugreifen können.

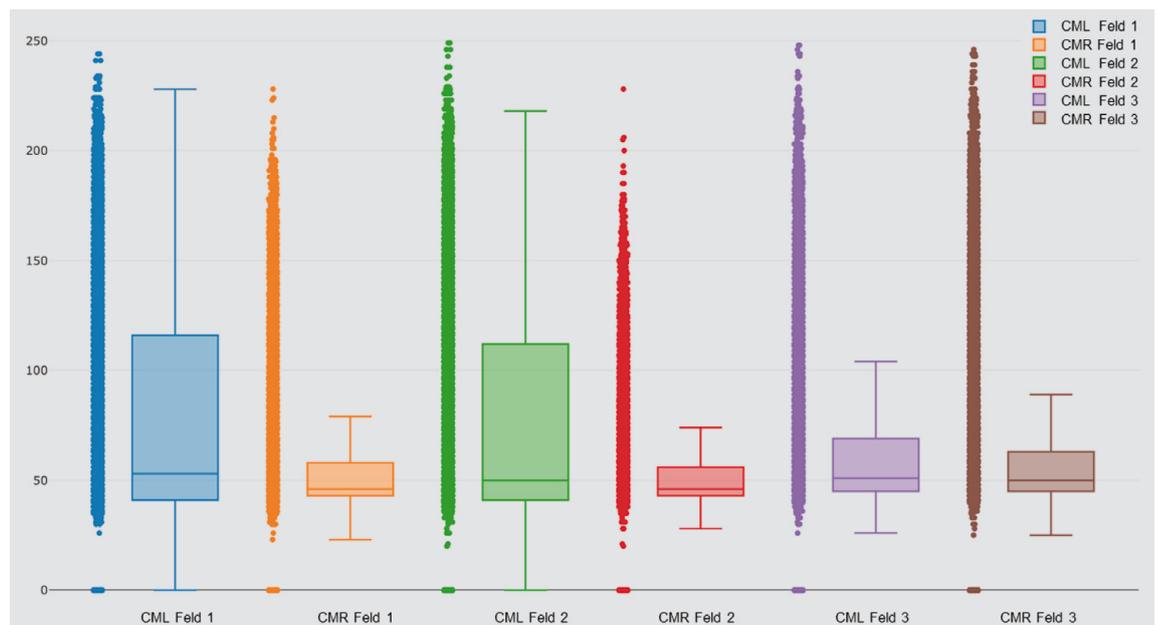
Eine **weitere Ausbaustufe** des dargestellten Data-Science-Set-Ups ist die Einbindung exogener, d.h. in diesem Falle maschinen-externer Daten. Dies soll am

Beispiel eines zweischichtig betriebenen Continuous Miner dargestellt werden. Die exogenen Daten umfassen dabei Informationen über die Schichtenzuordnung des Bergwerksbetreibers.

Bei der Auswahl eines längeren Zeitraums lassen sich verschiedene Leistungsmetriken ermitteln, anhand derer eine Bewertung und damit Interpretation der Zustände möglich wird. So sieht man im **Bild 11** deutlich, dass zwei Teams von Continuous-Miner-Maschinenbedienern über den dargestellten zweiwöchigen Zeitraum zwar ähnlich viele Schnitte pro Schicht durchführen, aber Team 2 (rechts) grundsätzlich den Schneidausleger höher hebt und die Standardabweichung der Schneidhöhe als Maß für die Irregularität des Schneidens deutlich niedriger als bei Team 1 ist (**Bild 6**). Hinsichtlich der erreichten Abbaumenge ergibt sich eine Mehrleistung von Team 2 von knapp 10%, was sich wiederum auch im OEE-Diagramm (Mitte) für diesen Zeitraum widerspiegelt. Die summierte produktive Zeit (hell- und dunkelgrün) und vor allem die Schneidzeit (dunkelgrün) von Team 2 liegen deutlich über der von Team 1.

Die Identifikation und vor allem auch der Evidenznachweis solcher Ungleichheiten sind wertvolle Ansatzpunkte für den Bergwerksbetreiber, um erfolgreiche Betriebsstrategien flächendeckend umzusetzen und Produktivitätspotenziale zu heben. Aber auch für den Maschinenhersteller sind diese Erkenntnisse von großer Relevanz: Er kann unter Berücksichtigung der Maschinennutzung im Feld durch Einschränkungen und Vorgaben die Funktionalitäten der Maschine optimieren und Schulungen hierzu und zu relevanten Automatisierungsfunktionalitäten anbieten.

Die Visualisierung berechneter Werte auf Basis klassisch-statistischer und manuell-regelbasierter Modelle legt zudem die Grundlagen, Machine-Learning-



**Bild 10:** Machine-Fingerprinting am Beispiel von Walzenladern

Jeder Boxplot stellt die Verteilung eines bestimmten Schneidmotorstroms jeweils im linken und rechten Schneidmotor (Cutter Motor Left (CML) und Cutter Motor Right (CMR)) in einem bestimmten Abbaufeld eines Bergewerks dar.

Modelle (ML-Modelle) für bestimmte Anwendungsfälle aufzubauen und anzuwenden. Der Nutzen von ML-Ansätzen ist einerseits, komplexes Know-how der Maschinenbediener algorithmisch abzubilden und damit – beispielsweise über Regressions- oder Klassifikationsmodelle – einen Beitrag zur intelligenten Automatisierung zu liefern. Andererseits lassen sich – beispielsweise über Clustering-Modelle – komplexe Wirkzusammenhänge mit multiplen Einflussgrößen entdecken, ohne dass die fortwährende präzise Kenntnis dieser Einflussgrößen für alle auftretenden Szenarien notwendig ist. Für den ersten Anwendungsfall werden im Folgenden exemplarisch zwei Anwendungen vorgestellt.

## 5.2 ML-Beispiel – Controlling und Optimierung der Abbau- bzw. Fahrwege

ML-Modelle können beispielsweise zum Controlling und zur Optimierung der Wege von Walzenladern in Strebsystemen verwendet werden und zur Automatisierung beitragen. Forschungsgegenstand ist in diesem Beispiel die Vorhersage der Längsneigung im Strebsystem zur Unterstützung der Tragarmautomation – insbesondere zur Anpassung an sich verändernde Strebneigungen. Konkret lassen sich so Vorschläge zur Steuerung der Maschine unterbreiten bzw. ein Controlling der Steuerung implementieren.

Das ML-Modell erlernt basierend auf den ersten 40 Schnitten des Walzenladers die Neigungssituation des Strebs – im **Bild 12** in Heat-Maps in Form positiver und negativer Maschinenlängsneigung über die Streblänge dargestellt. Bildlich gesprochen erlernt das Modell, dass eine Welle im Kohlenflöz mit jedem Folgeschritt früher oder später geschnitten wird. Dem erlernten Muster entsprechend gelingt es dann, die Neigungswerte der nächsten drei bis fünf Schnitte mit hoher Präzision vorherzusagen.

Sobald Sensordaten über die Längsneigung des Walzenladers bereitstehen, lassen sich auf Grundlage dieser Daten lernende ML-Modelle erstellen. Die zurückliegenden Längsneigungswerte und weiteren Variablen (Walzenladerposition im Streb, Motorströme etc.) werden im Lernschritt auf die Zielvariable, hier die Längsneigungswerte für die nächsten fünf Schnitte, abgebildet. Das neuronale Netz hat die Aufgabe, die Abbildungsvorschrift zu approximieren, sodass nach dem Lernschritt das neuronale Netz in der Lage ist, selbstständig die Zielvariable zu berechnen. Im konkreten Fall wurden verschiedene Architekturen neuronaler Netze, wie reine feed-forward-Netze oder Netze mit Zeitreihenoptimierung, ausprobiert und validiert. Ergebnis ist schließlich ein reines feed forward Netz (**Bild 13**), das die Längsneigungswerte von bis zu fünf zukünftigen Schnitten besser als die Referenzmetrik gleichbleibender Längsneigungen hervorbringen kann.

Das beschriebene ML-Modell kann – wie die Berechnung klassisch-statistischer und manuell-regelba-

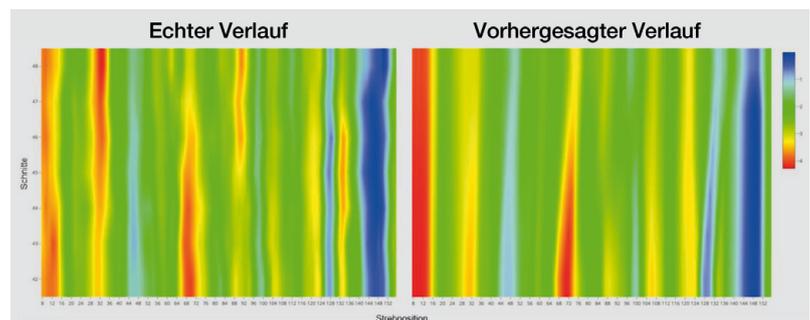


**Bild 11:** Analyse der Abbaumenge verschiedener Gruppen von Continuous-Miner-Bedienern

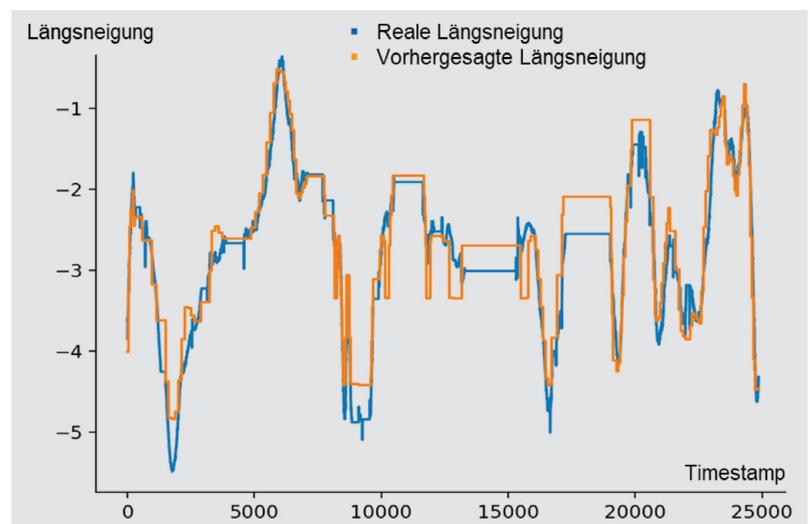
sierter Modelle auch – im Batch- und Speed-Layer einer im Abschnitt 4 dargestellten Big-Data-Dateninfrastruktur berechnet werden.

## 5.3 ML-Beispiel – Kompensation eines Sensorausfalls bei Schneidzykluserkennung

Der zweite exemplarische Anwendungsfall hat die verlässliche Gewinnungszykluserkennung und damit



**Bild 12:** Reale und prädierte Längsneigung eines Walzenladers über den Strebverlauf



**Bild 13:** Deep-Learning-Prädiktion mit MLP-FF-Regressor-Netz mit 8 Schichten und 86.785 Parametern



**Bild 14:** Reale und prädierte Höhendaten eines CM

die Abbaumengenermittlung für Continuous Miner trotz Sensorausfall zum Ziel. Die Abbaumenge korreliert bis auf Varianzen in der Schneidhöhe stark mit der Anzahl der durchgeführten Schneidzyklen. Wie bereits im Abschnitt 3 dargestellt, ist die bestimmende Einflussgröße für die Gewinnungszykluserkennung die Veränderung der Höhe des Schneidauslegers. Oft muss jedoch der Ausfall dieser Größe ermittelnden Sensors bzw. die Nichtübermittlung der aufgezeichneten Daten kompensiert werden – nicht zuletzt aufgrund der zum Teil sehr unwirtschaftlichen Rahmenbedingungen im Kammer-Pfeiler-Bau und der damit verbundenen starken Beanspruchungen der Maschinen. Solche Ausfälle können durch eine alternative Bestimmung der Höhe des Schneidauslegers über die Zylinderwege bzw. mittelbar über die diese auslösenden und ausfallsicher messbaren Ventilströme kompensiert werden (**Bild 14**).

Die physikalischen Zusammenhänge liegen zwar auf der Hand, sind jedoch nur sehr schwer in Formeln abzubilden. Es sind zu viele Einflussgrößen – von Reibwiderständen über überlagernde dynamische Belastungen bis hin zu Temperaturen – zu berücksichtigen. Zudem sind diese Einflussgrößen nicht für alle Anwendungsfälle konstant. Sie können je nach Abbaupanel stark variieren. Die Lösung ist die Erstellung eines ML-Modells, welches für einen repräsentativen Monat angelernt wird. Hiermit lassen sich bei nicht vorhandenen Höhendaten die Daten des ML-Modells nutzen und so Lücken in der Datenübertragung ersetzen. Im **Bild 13** sind bis ca. 10:10 Uhr die blauen Realdaten zu erkennen, hinterlegt mit den violetten Prädiktionsdaten, die dann ab 10:10 Uhr weiterhin die Höhe des Schneidauslegers bestimmen und damit für eine unterbrechungsfreie Schneidzykluserkennung sorgen. Ähnlich wie im vorherigen Anwendungsbeispiel wurde das ML-Modell mit unterschiedlichen Maschinenparametern trainiert. Ziel des Modells ist die Approximation der Schneidauslegerhöhe basierend auf diesen Maschinenparametern. Wie im **Bild 14** dargestellt, kann mit dem ML-Modell die Höhe des Schneidauslegers erfolgreich bestimmt werden.

## 6 Fazit und Ausblick

Obwohl sich die Industrie in puncto Digitalisierung noch nicht am Ende des Wegs befindet, belegen die

vorgestellten Beispiele auf sehr eindrucksvolle Weise das bereits heute realisierbare Potenzial von Data-Science-Ansätzen im Bergbau – sowohl auf der Anwender- als auch auf der Herstellerseite. Insbesondere für Unternehmen, die erfolgreich heuristische Ansätze wie Lean-Methodiken nutzen und damit an Optimierungsgrenzen stoßen, ergeben sich viele Möglichkeiten, Prozessabweichungen und -auffälligkeiten zu erkennen und zu quantifizieren, um dann mit den bewährten Ansätzen aus der Lean-Welt Veränderungen herbeizuführen. Best Practices können hart belegt werden und das Potenzial von Verhaltensveränderungen bzw. einleitenden Maßnahmen kann im Vorfeld bestimmt und in der Umsetzung überwacht werden. Neben der Nutzung explorativer Visualisierungen und manuell-regelbasierter Modellierungen werden aber auch die Voraussetzungen für die Anwendung von Machine-Learning-Ansätzen in realen, operativen Anwendungen immer besser. Wertvolle Unterstützung kann auf vielfältige Weise geleistet werden. So können beispielsweise mit Modellen des „unsupervised Learning“ Abweichungen vom Normalfall ohne vorheriges Einlernen von „Schlecht-Fällen“ detektiert werden oder auch der Ausfall von Sensordaten durch eine alternative Menge korrelierender Daten kompensiert werden. Neue Geschäftsmodelle, die auf Verfügbarkeit und Output basieren, können damit weitaus robuster als bisher ausgestaltet werden. Das Potenzial auf diesem Gebiet ist groß, wird aktuell jedoch nur in sehr geringem Umfang ausgeschöpft. Mehr und mehr Betreiber und Hersteller beginnen jedoch, in diesen neuen Kategorien zu denken und die damit verbundenen Chancen einzufordern. Für die einzelnen Unternehmen bedeutet dies auch, Kompetenzen in der Operationalisierung von Data-Science-Algorithmen und -Verfahren aufzubauen und damit einhergehend ihre Dateninfrastruktur weiter auf- und auszubauen. Ziel muss es sein, Data Science in Unternehmensstruktur- und -denken fest zu verankern – analog zum Methodenset aus der Lean-Welt wie KVP (Kontinuierlicher Verbesserungsprozess) und FMEA (Failure Mode and Effects Analysis bzw. Fehlermöglichkeits- und -einflussanalyse). Das Zusammenspiel aus wachsender Kompetenz und Akzeptanz wirkt zudem selbstverstärkend – nicht zuletzt auch in der Anpassung und Erweiterung existierender Geschäftsmodelle.

## 7 Quellenverzeichnis

- [1] Bartnitzki, T. (2016): OPC UA - Ein Kommunikationsstandard für Bergbau 4.0. In: Tagungsband zum AKIDA 2016 / Nienhaus, Karl (editor) (pp. 37-48). Institut für Maschinentechnik der Rohstoffindustrie.
- [2] Suci, M.; Kowitz, S. (2019): Digitizing Raw Material Mining – End-to-End Integration into an IIoT Platform for the Analysis of Machine Data. In: Mining Report Glückauf 155 (2019) No. 4.
- [3] Drüppel, E. (2010): Entwicklung eines Konzeptes für die schneidende Gewinnung im Steinsalz. Hochschul-

- bibliothek der Rheinisch-Westfälischen Technischen Hochschule Aachen.
- [4] Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P. (1996): From Data Mining to Knowledge Discovery in Databases. In: AI Magazine, 17(3), 37.
- [5] Provost, F.; Fawcett, T. (2017): Data Science für Unternehmen. mitp Verlag, Ferchen. ISBN: 978-3-95845-546-7.
- [6] Marz, N.; Warren, J. (2015): Big data: principles and best practices of scalable real-time data systems. Manning Verlag, Shelter Island, New York, USA.
- [7] Diebold, F. X. (2020): On the Origin(s) of the Term “Big Data“. In: arXiv e-prints, arXiv:2008.05835.
- [8] Laney, D. (2001). 3D Data Management: Controlling Data Volume, Velocity, and Variety. Technical Report, META Group.
- [9] White, T. (2015). Hadoop: The Definitive Guide. 2. Auflage. O'Reilly Media, Sebastopol, Kalifornien, USA. ISBN: 978-1-49190-163-2.
- [10] Chambers, B.; Zaharia, M. (2018): Spark: The Definitive Guide: Big Data Processing Made Simple. 1. Auflage. O'Reilly Media, Sebastopol, Kalifornien, USA. ISBN: 9781491912218.

### Prof. Dr.-Ing. Dipl.-Wirt. Ing. Andreas Merchiers

hat ein Maschinenbaustudium sowie ein wirtschaftswissenschaftliches Zusatzstudium absolviert. Schwerpunktthemen seiner beruflichen Laufbahn bildeten stets die Herausforderungen produzierender Unternehmen, wie Lean-Gedanken entlang der Wertschöpfungskette, Produktions- und Logistikstrukturen sowie Industrie-4.0-Anwendungen im Maschinen- und Bergbau. Seine akademische Ausbildung schloss Andreas Merchiers an der RWTH Aachen mit der Promotion zum Dr.-Ing. ab und ist heute Professor an der Hochschule Bochum. Er lehrt und forscht zu Produktionsmanagement / Industrie 4.0 und Technischer Investitionsplanung. Zusätzlich ist er als Industrieberater tätig.



**Kontakt:** andreas.merchiers@hs-bochum.de

### Prof. Dr. rer. nat. Henrik Blunck

ist Professor für praktische Informatik an der Hochschule Bochum. Er ist diplomierter Mathematiker und hat an der WWU Münster in Informatik promoviert. Seine Forschungsgebiete umfassen Data Science und Big Data Analytics, insbesondere in den Domänen Mobility Data, Context-aware Consumer Software und Industrie-4.0-Anwendungen. In diesen Gebieten forscht er von Datenakquise und Algorithmen-Entwicklung bis hin zu real-weltlichen und markttauglichen Produkten und Errungenschaften in sich entwickelnden Gesellschafts- und Geschäftsfeldern. Zusätzlich ist er als Industrieberater tätig.



**Kontakt:** henrik.blunck@hs-bochum.de

### Arne Köller, M.Sc.,

studierte Informatik mit Schwerpunkt Intelligente Algorithmen an der Hochschule Bochum. Schwerpunkte seiner wissenschaftlichen und praktischen Arbeit waren Maschinenautonomie und Sensorverfahren in der Rohstoffindustrie. Nach dem Studium folgte eine Beschäftigung bei der Eickhoff Bergbautechnik GmbH u. a. im Bereich Data Engineering. Aktuell arbeitet er als wissenschaftlicher Mitarbeiter im Bereich Maschinenautonomie und Kommunikationstechnik in der Rohstoffindustrie am Institute for Advanced Mining Technologies der RWTH Aachen.



**Kontakt:** akoeller@amt.rwth-aachen.de

### Dr.-Ing. Christian Gierga

hat nach seinem Bergbaustudium an der RWTH Aachen am Institut für Aufbereitung promoviert. Seine berufliche Laufbahn führte ihn anschließend im Jahr 2000 zur HAZEMAG & EPR GmbH in Dülmen, zunächst in die Projektierung und dann in den Vertrieb, dessen Leitung er 2004 übernahm. Im Jahr 2007 wechselte er zur Eickhoff Bergbautechnik GmbH wo er nunmehr verantwortlich für Service und Produktion ist.



**Kontakt:** kontakt@eickhoff-bochum.de