RISC Software GmbH
www.risc-software.at

INSIGHTS³

Das Fachmagazin der RISC Software GmbH
zu aktuellen Forschungs- und Entwicklungsthemen.

Data Science und Prescriptive Analytics

Software-Reengineering

Industrial AI

Projektmanagement

Green IT

INHALT

Liebe Leserin, lieber Leser,

mit dieser dritten Ausgabe unseres Magazins setzen wir unsere Reise fort, Ihnen spannende Einblicke in unsere vielfältigen Arbeitsgebiete zu geben. Nach den ersten beiden Ausgaben, in denen wir Ihnen bereits zentrale Themenfelder wie agile Softwareentwicklung, Data Science, Prescriptive Analytics und intelligente Transportsysteme vorgestellt haben, widmen wir uns diesmal besonders den Zukunftsfragen, die unsere Kund*innen, Partner*innen – und uns selbst – intensiv beschäftigen.

Die Beiträge dieser Ausgabe spannen den Bogen von der weiteren Automatisierung in der Produktion über die Sicherung von Datenqualität und den Einsatz von Machine Learning bis hin zu innovativen Softwarelösungen und nachhaltiger IT. Dabei zeigen wir, wie moderne Methoden helfen, komplexe Herausforderungen zu bewältigen – sei es bei der Stabilisierung von Stromnetzen, in der mathematischen Modellierung von Produktionsprozessen oder bei der Transformation bestehender Softwaresysteme.

Was uns als RISC Software GmbH auszeichnet, ist die enge Verbindung von wissenschaftlicher Forschung und praktischer Umsetzung. Unsere Expert*innen bringen interdisziplinäres Know-how ein, das in erfolgreichen Projekten mit unseren Kund*innen kontinuierlich gewachsen ist. Dieses Zusammenspiel ermöglicht nicht nur technologische Innovationen, sondern schafft auch langfristige Partnerschaften, die weit über einzelne Projekte hinaus Bestand haben.

Die digitale Transformation bleibt dabei ein zentrales Leitmotiv – doch wir wissen: Sie gelingt nur, wenn Menschen, Prozesse und Technologien zusammengedacht werden.
Genau hier sehen wir unsere Rolle: als Impulsgeber, als Partner und als Vermittler von Wissen.
So unterschiedlich die Themen dieser Ausgabe auch sind, machen sie dennoch deutlich, dass Fortschritt nur dann gelingt, wenn Technologie, Forschung und Praxis Hand in Hand gehen.
Unser Anspruch ist es, praxisnahe Lösungen zu entwickeln, die Unternehmen stärken und zugleich den Weg in eine nachhaltige digitale Zukunft ebnen.

Wir laden Sie ein, in dieser Ausgabe neue Impulse zu entdecken, Einblicke in unsere Forschungs- und Entwicklungsarbeit zu gewinnen und Anregungen für Ihre eigenen Vorhaben mitzunehmen.

Viel Freude beim Lesen!

Wolfgang Freiseisen
CEO Software GmbH

Industrial AI: Von den Rohdaten zur einen effizienteren Produktionslandschaft

von Dr. Roxana-Maria Holom, MSc und Dr. Evans Doe Ocansey

Der Industriesektor durchläuft, wie auch andere Bereiche, derzeit eine Phase der digitalen Transformation. Das bedeutet, dass Fertigungsunternehmen an verschiedenen Digitalisierungsaktivitäten beteiligt sind [1]. Innerhalb dieses Kontextes spielen industrielle Daten und die Art und Weise, wie sie verarbeitet, visualisiert und genutzt werden, eine wesentliche Rolle.

Anwendungsperspektiven von Industrial AI für Fertigungsunternehmen

Sich ausschließlich auf Technologie zu verlassen, erzeugt keinen geschäftlichen Mehrwert, wenn die Probleme der Branche nicht gründlich untersucht werden. Es gibt viele Möglichkeiten, wie industrielle KI zur digitalen Transformation der Fertigung beitragen kann. Einige der ansprechendsten Bereiche, in denen sie eingesetzt werden kann, sind: Prozessanwendungen zur Produktivitätsverbesserung (d.h. intelligente Produktion), Produkt- und Serviceanwendungen, Erkenntnisanwendungen zur Wissensentdeckung (d.h. Ermittlung der Grundursache, Decision-Making) [2].

Konkrete Beispiele, die zu den oben genannten Kategorien passen, sind zwei unserer laufenden EU-Projekte: "Customizable AI-based in-line process monitoring platform for achieving zero-defect manufacturing in the PV industry" (Platform-Zero) und "Data and Metadata for advanced Digitalization of Manufacturing Industrial Lines" (metaFacturing). Das Projekt Platform-Zero zielt darauf ab, die Produktionsqualität von Photovoltaikanlagen zu verbessern und gleichzeitig die Herstellungskosten durch eine Null-Fehler-Fertigung zu senken. Erreicht wird dies durch die Anwendung zerstörungsfreier Prüfmethoden und -technologien zur frühzeitigen Erkennung, Korrektur und Vermeidung kritischer Produktionsfehler. Die Daten werden in Echtzeit ausgewertet, um den Produktionsprozess zu optimieren und die Produktqualität zu verbessern. Das Projekt metaFacturing fokussiert sich auf die Schaffung einer digitalisierten Werkzeugkette für die Produktion von Metallteilen (Gießen und Schweißen). Zur Gewinnung von Prozesseinblicken, zur Verbesserung der Effizienz des Produktionsprozesses (z.B.: Optimierung der Prozessparameter) sowie der Produktqualität (z.B.: Fehlerreduzierung) werden vertrauenswürdige KI- und Hybridmethoden analysiert und implementiert.

Um ein umfassendes Verständnis des Themas zu erlangen, werden im Folgenden mehrere Schlüsselaspekte betrachtet, die in vier Kategorien eingeteilt sind. Zunächst werden die Probleme und Bedürfnisse der Fertigungsunternehmen erörtert. Darüber hinaus gehen wir auf den Mehrwert ein, den industrielle KI als Lösung für ihre Probleme bieten kann, aber auch auf die Herausforderungen, die sich bei der Anwendung von industrieller KI ergeben (siehe Abbildung 1).

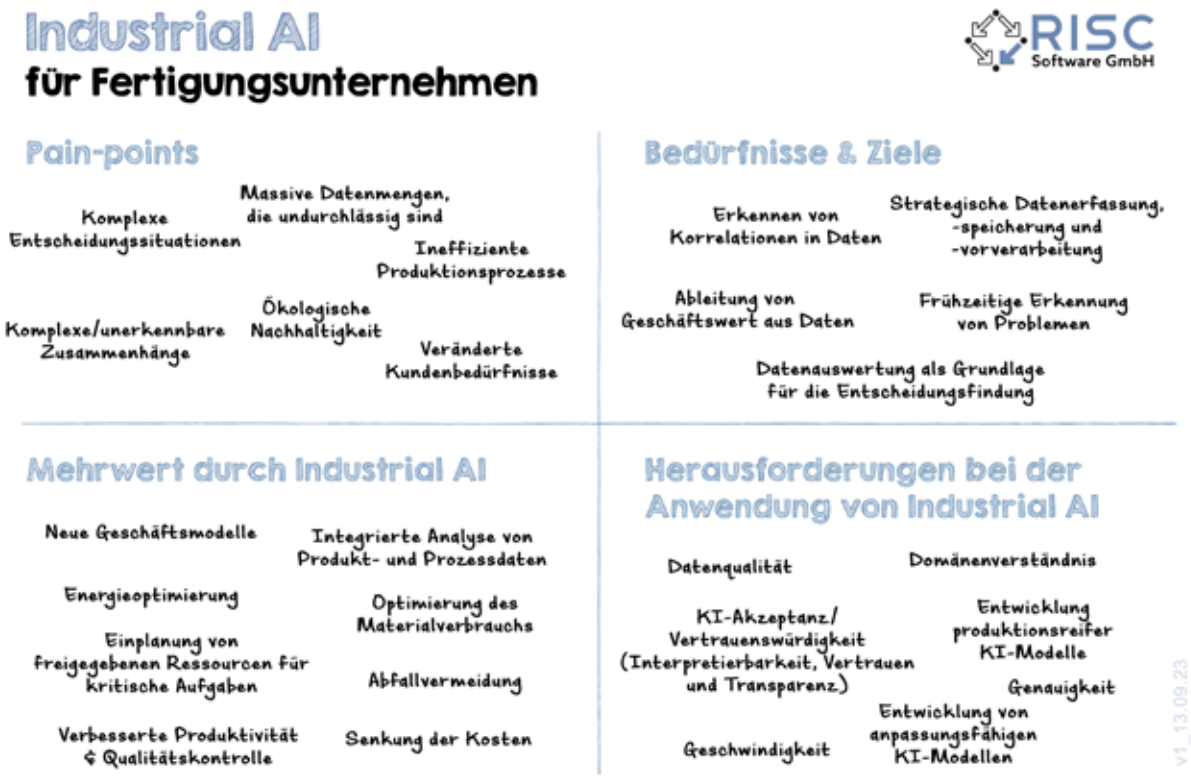


Abbildung 1: Kernpunkte der Anwendungsperspektiven von Industrial AI für Fertigungsunternehmen

Pain-Points

Im Folgenden betrachten wir einige der zentralen Pain-Points, mit denen Produzierende konfrontiert sind, wenn es darum geht, industrielle KI erfolgreich einzuführen und zu nutzen:

Massive Datenmengen, die undurchlässig sind

Die moderne Fertigung generiert heute eine Vielzahl von Daten durch Einsatz von technischen Systemen (Sensoren, Kameras). Es ergibt sich eine Mischung aus strukturierten und unstrukturierten Daten, die oft so komplex und umfangreich sind, dass es schwierig ist, darin klare Muster und Erkenntnisse zu identifizieren. Außerdem ist es schwer zu sagen, welche Daten für die weiteren Analysen relevant sind und gespeichert werden sollten. Ein konkretes Beispiel ist die Produktion von Halbleitern. Enorme Mengen von Sensordaten, Prozessdaten und Qualitätsdaten werden erzeugt. Diese Daten müssen analysiert werden, um Abweichungen oder Fehler in den Produktionsprozessen zu erkennen. Aufgrund der Vielzahl von Datenquellen und -formaten wird eine Herausforderung sein, die relevanten Informationen zu extrahieren und zu interpretieren.

Komplexe/unerkennbare Zusammenhänge

Industrielle Prozesse sind oft durch vielfältige Prozessparameter und Wechselwirkungen geprägt, die es herausfordernd machen, versteckte Zusammenhänge zwischen den Daten zu erkennen. Nehmen wir als Beispiel die Herstellung von Gussteilen: In diesem Kontext sind die wechselseitigen Zusammenhänge der Prozessparameter – wie die Schmelztemperatur, Gießformtemperatur, Geschwindigkeit der ersten Phase und Geschwindigkeit der zweiten Phase – komplex, nicht linear und widersprüchlich [3].

Komplexe Entscheidungssituationen; Ineffizient Produktionsprozesse

Die Entscheidungsfindung in der Fertigung erfordert die Berücksichtigung zahlreicher Faktoren und Einschränkungen. Wie wir im vorigen Abschnitt gesehen haben, ist die Bestimmung des optimalen Toleranzfensters für Parameter keine einfache Aufgabe. Bleiben wir im gleichen Kontext – dem Gießen von Metallteilen – so stellen wir fest, dass diese Entscheidung Auswirkungen auf die Prozesseffizienz hat (d.h.: Metallteile werden von der Gießmaschine automatisch als Ausschuss qualifiziert, wenn die Messungen außerhalb des Toleranzfensters liegen).

Ökologische Nachhaltigkeit

Die Fertigungsindustrie steht vor der Herausforderung, umweltfreundliche Praktiken zu implementieren, um Ressourcenverbrauch und Emissionen zu reduzieren. Z.B.: Die Reduzierung des Wasser- und Energieverbrauchs in der Textilproduktion zur Minimierung des ökologischen Fußabdrucks.

Veränderte Kundenbedürfnisse

Kundenanforderungen ändern sich ständig, und Hersteller müssen agil sein, um Produkte anzupassen und den Marktanforderungen gerecht zu werden. Die Wahl der Werkstoffe wirkt sich erheblich auf die Qualität der Produkte aus. Der Übergang von herkömmlichem Stahl zu fortschrittlichen Leichtbauwerkstoffen wie Kohlefaserverbundwerkstoffen für Karosseriebleche beispielsweise bringt aufgrund der einzigartigen Eigenschaften von Kohlenstoff komplexe Produktionsprozesse mit sich. ›

Bedürfnisse & Ziele

Erkennen von Korrelationen in Daten (Ermittlung der Grundursache)

Durch die Ermittlung der Grundursache eines Problems können Hersteller gezielte Lösungen implementieren, um ein erneutes Auftreten des Problems zu verhindern. Dazu muss man tief in die Daten eindringen, um Korrelationen und damit die zugrunde liegenden Faktoren zu entdecken, die wahrscheinlich für bestimmte Probleme oder Anomalien verantwortlich sind.

Nehmen wir zum Beispiel ein Szenario in der Elektronikfertigung, bei dem eine bestimmte Charge von Produkten bei Qualitätsprüfungen immer wieder durchfällt. Durch eine Ursachenanalyse könnte herausgefunden werden, dass eine bestimmte Maschinenkomponente höchstwahrscheinlich nicht korrekt kalibriert ist.

Frühzeitige Erkennung von Problemen

Durch fortgeschrittene Analyse von Produktionsdaten können Probleme frühzeitig erkannt werden, noch bevor sie zu ernsthaften Fehlern oder Ausfällen führen. In der Energieerzeugung könnten ungewöhnliche Abweichungen im Stromverbrauch eines Generators auf ein potenzielles Problem hinweisen, das behoben werden muss, um einen Ausfall zu vermeiden.

Strategische Datenerfassung, -speicherung und -vorverarbeitung

Unternehmen verlagern ihren Schwerpunkt von der Anhäufung von Massendaten auf das strategische industrielle Datenmanagement. Die Optimierung des Datenbedarfs und der Datenverarbeitung steht auch im Einklang mit den Zielen der Europäischen Kommission. Eine intelligente Datenauswahl und -aufbereitung verringert die Notwendigkeit, große Datenmengen und/oder große KI-Modelle zu sammeln, zu speichern, zu verarbeiten und zu übertragen und damit den Energieverbrauch zu senken [4].

Datenauswertung als Grundlage für die Entscheidungsfindung; Ableitung von Geschäftswert aus Daten

Hersteller nutzen Datenanalysen, um fundierte Entscheidungen zu treffen. Ein Chemiewerk kann unmittelbare Erkenntnisse aus nahtlos integrierten Industriedaten gewinnen, die sich vom Edge bis zur Cloud erstrecken. Dies kann durch die Fusion verschiedener Datenquellen erreicht werden, was eine agile Entscheidungsfindung im gesamten Unternehmen fördert. Bei komplexen Entscheidungsfindungen können diese Daten in Form von Modellen auch in Optimierungsmodelle integriert werden und so die Verantwortlichen bei Planungsproblemen unterstützen.

Mehrwert durch Industrial AI

Innovative Ansätze und der intelligente Einsatz von industrieller KI sind erforderlich, um den Anforderungen der modernen Industrie gerecht zu werden. Im Gegensatz zu industriellen KI-Modellen werden allgemeine KI-Modelle auf der Grundlage umfangreicher Anlagendaten trainiert, die häufig nicht das gesamte Spektrum möglicher Betriebsabläufe abdecken. Dies liegt daran, dass allgemeine KI-Modelle keine Bedingungen für unterschiedliche Zwecke (z. B. Sicherheit, Design) oder Bedingungen, die durch physikalische und chemische Gesetze vorgegeben sind, berücksichtigen.

Verbesserte Produktivität & Qualitätskontrolle

Industrielle KI trägt zur Verfeinerung des Qualitätssicherungsprozesses bei, indem sie den Prozess automatisiert und Defekte frühzeitig erkennt. Dadurch steigert sich die gesamte Produktions- und Produktqualität.

Neue Geschäftsmodelle

Industrielle KI ermöglicht die Umgestaltung von Arbeitsprozessen und die Schaffung neuer Geschäftsmodelle, die auf datengetriebener Innovation beruhen.

Höhere Effizienz

Die Anwendung industrieller KI führt zur Optimierung des Energieverbrauchs, zur effizienten Nutzung von Materialien, zur Reduzierung von Abfall und zur Senkung der Kosten. Zusätzlich ermöglicht sie die strategische Zuweisung freigegebener Ressourcen für kritische Aufgaben.

Integrierte Analyse von Produkt- und Prozessdaten

Industrielle KI ermöglicht eine nahtlose Integration und Analyse von Daten aus Produktions- und Prozessabläufen. Dies ermöglicht fundierte Entscheidungen, um sowohl die Produktqualität als auch die Effizienz der Produktionslinie zu steigern.

Herausforderungen bei der Anwendung von Industrial AI

Der Schlüssel zu einer erfolgreichen industriellen KI-Anwendung liegt in der Umwandlung von Rohdaten in intelligente Erkenntnisse für eine schnelle Entscheidungsfindung. Von den Feinheiten der Datenverwaltung und -integration bis hin zu den Komplexitäten der Anpassung von KI-Modellen an reale Produktionsumgebungen müssen Hersteller die folgenden Herausforderungen proaktiv angehen.

Datenqualität

Obwohl die Datenumgebung in der Industrie heutzutage eine Big-Data-Umgebung ist, gibt es eine Mischung aus strukturierten und unstrukturierten Daten, die von minderer Qualität sein können (z. B.: unausgewogene Daten, fehlende Datenpunkte, ungenaue Sensormessungen, Datendrift, inkonsistente Formate, begrenzter Umfang usw.).

Entwicklung produktionsreifer KI-Modelle

Es fehlt ein systematischer Ansatz zur effizienten Entwicklung von KI-Modellen, die für den Einsatz in der Industrie bzw. für die Integration in den Produktionsprozess bereit sind. Neben Herausforderungen wie Datenkomplexität und -qualität, mangelndem Fachwissen und Interpretierbarkeit der Modelle, müssen bei der Integration von KI-Modellen in bestehende Produktionssysteme auch Kompatibilitätsprobleme und Ressourcenbeschränkungen berücksichtigt werden.

KI-Akzeptanz/Vertrauenswürdigkeit (Interpretierbarkeit, Vertrauen und Transparenz)

Die Glaubwürdigkeit von KI-Systemen in der Industrie kann beeinträchtigt werden, wenn die Genauigkeit nicht annähernd perfekt ist, da diese Systeme kritische Sicherheits-, Zuverlässigkeits- und Betriebsfragen angehen könnten. Jedes Versagen der KI könnte negative wirtschaftliche und/oder sicherheitstechnische Auswirkungen haben und vom Einsatz von KI-Systemen abhalten. Durch die Einhaltung der Anforderungen an vertrauenswürdige KI (d.h.: in Anlehnung an "The Assessment List for Trustworthy Artificial Intelligence" (ALTAI) [5]) werden die Datenanalyseergebnisse nachvollziehbar (u.a. interpretierbar und transparent) gemacht.

Genauigkeit & Geschwindigkeit

Produktionsprozesse erfordern schnelle Entscheidungen und die produzierten Werkstücke können teuer sein, daher müssen KI-Anwendungen schnell reagieren, um Verschwendung und andere Folgen zu vermeiden. Im Gegensatz zu anderen KI-Systemen (z. B. Empfehlungssystemen) ist bei industriellen KI-Systemen außerdem eine sehr geringe Toleranz gegenüber falsch positiven und negativen Ergebnissen erforderlich, damit sie in der Produktion eingesetzt werden können.

Domänenverständnis & Entwicklung von anpassungsfähigen KI-Modellen

Die Einbeziehung von Fachwissen ist ein Muss, um den Unterschied zwischen allgemeiner KI und industrieller KI deutlich zu machen. Die Dateningenieur*innen und Datenwissenschaftler*innen müssen mit den Domänenexpert*innen zusammenarbeiten und Fachwissen in den Modellierungsprozess einbeziehen. Und um ›

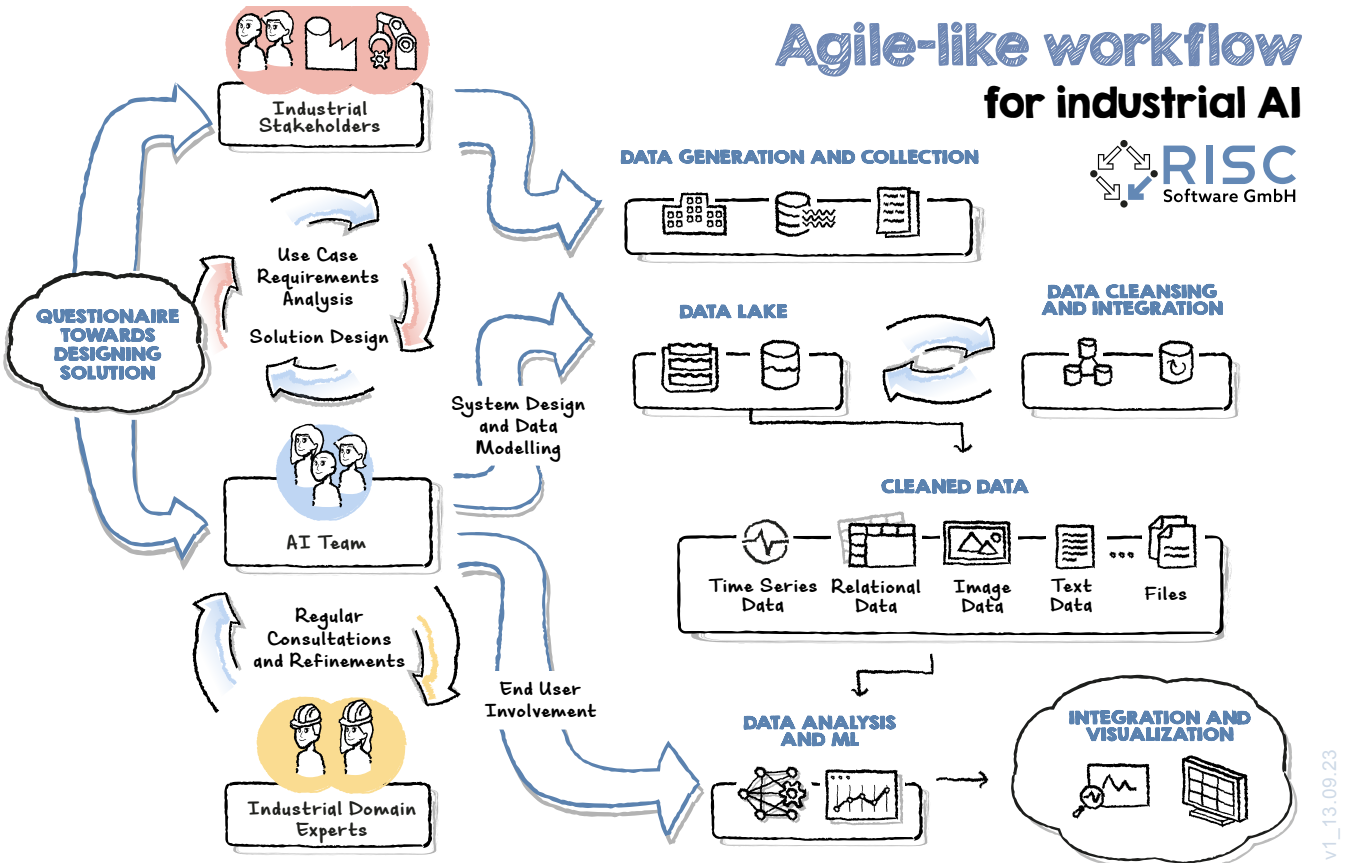


Abbildung 2: Agiler Workflow für industrielle KI-Lösungen

die Einbeziehung des Fachwissens zu maximieren, müssen die entwickelten Modelle adaptiv lernen und die Erkenntnisse der Fachleute als Wissen akkumulieren.

Ein systematischer Ansatz für Industrial AI

Wie wir im vorangegangenen Abschnitt gesehen haben ergibt, sich durch die zahlreichen Herausforderungen meist ein beträchtlicher zeitlicher Aufwand bis aussagekräftige Ergebnisse aus der Produktion vorhanden sind. Manchmal wird dieses Ziel nicht erreicht, weil es zu komplex ist und der Fokus verloren geht. Deshalb verfolgt die RISC Software GmbH einen agilen Ansatz (mit starker Einbindung der tatsächlichen Akteure) bei KI-basierter Datenanalyse im industriellen Bereich.

Abbildung 2 veranschaulicht den vorgeschlagenen agilen Workflow, der für ein industrielles KI-Projekt geeignet ist. Der Workflow beginnt mit einem kontinuierlichen Diskurs zwischen den industriellen Stakeholdern und dem Team, das die KI-Lösung entwickelt (im Workflow als AI-Team bezeichnet). Nach detaillierten Analysen durch die Beteiligten werden die Anwendungsfälle und deren Anforderungen festgelegt. Das AI-Team entwirft nach weiterem Dialog mit den industriellen Stakeholdern geeignete Lösungen für die Anwendungsfälle. Um die geeignete Datenumgebung modellieren zu können, erstellt das AI-Team eine Reihe von Fragebögen, die von den Datenlieferanten ausgefüllt werden müssen. Diese Fragebögen bilden die Grundlage für die Anforderungen an die Datenaufbereitung und -integration. Anschließend liefern die Datenlieferanten verschiedene Arten von Daten, die auf den Anforderungen des Anwendungsfalls basieren. Die Datentechniker*innen des AI-Teams verarbeiten, transformieren und laden diese Daten dann in einen Data Lake. Dieser Data-Engineering-Prozess stützt sich nicht nur auf die Beiträge der Datenwissenschaftler*innen des AI-Teams, sondern auch auf die Beiträge der industriellen Stakeholder, wie z. B. Domänenexpert*innen oder Prozessingenieur*innen. Die im Data Lake verfügbaren Daten werden von den Datenwissenschaftler*innen des AI-Teams gründlich bereinigt und für die Analyse vorbereitet. Sie führen explorative Datenanalysen durch und arbeiten mit Domänenexpert*innen zusammen, um den Datenanalyseprozess weiter zu verfeinern. Die vorverarbeiteten Daten werden anschließend verwendet, um KI-Modelle gemäß den Spezifikationen des Anwendungsfalls zu trainieren. Die Ergebnisse dieser KI-Modelle werden anschließend gemeinsam mit Domänenexpert*innen geprüft, um sie für die Produktion geeignet zu machen.

Der entscheidende Aspekt dieses Ansatzes liegt in der aktiven Beteiligung von Domänenexpert*innen während des gesamten Design- und Implementierungszyklus der KI-Lösungen.

Autor*innen

Dr. Roxana-Maria Holom, MSc
Data Science Project Manager & Researcher

Dr. Evans Doe Ocansey
Data Scientist

Referenzen

[1] Lázaro, O. et al.: "Model-Based Engineering and Semantic Interoperability for Trusted Digital Twins Big Data Connection Across the Product Lifecycle". In: Curry, E., Auer, S., Berre, A.J., Metzger, A., Perez, M.S., Zillner, S. (eds) Technologies and Applications for Big Data Value. Springer, 2022.
[2] Deloitte: "AI Enablement on the Way to Smart Manufacturing", Deloitte Survey on AI Adoption in Manufacturing, 2020.

[3] Ducic, N. et al.: "Casting Process Improvement by the Application of Artificial Intelligence", In Appl. Sci. 2022, 12, 3264. https://doi.org/10.3390/app12073264.

[4] European Commission: Horizon Europe – Work Programme 2023-2024, Digital, Industry and Space. European Commission Decision C(2023) 2178 of 31 March 2023.

[5] High-Level Expert Group on Artificial Intelligence (AI HLEG): The Assessment List for Trustworthy Artificial Intelligence (ALTAI), July 2020, Ethics guidelines for trustworthy AI | Shaping Europe's digital future (europa.eu).

[6] AspenTech: "The future starts with Industrial AI", MIT Technology Review, https://www.technologyreview.com/2021/06/28/1026960/the-future-starts-with-industrial-ai/, 2021.

| Fazit

Industrial AI – Die Verbindung von Fachwissen und Datenwissenschaft

Die Entwicklung von KI-Lösungen, die für Fertigungsprozesse wertvoll sind, setzt voraus, dass sie bewusst mit dem spezifischen Fachwissen der Branche angereichert werden [6]. Dies ist entscheidend für die Erzielung von Vorteilen durch KI. Industrielle KI erreicht dies durch die Kombination von Datenwissenschaft, KI und industriellem Fachwissen. Im Rahmen eines systematischen industriellen KI-Workflows werden daher Algorithmen für maschinelles Lernen entwickelt, implementiert und eingesetzt, die auf die spezifischen industriellen Anwendungen zugeschnitten sind. ◆

Datenqualität in der Praxis

von DI Paul Heinzlreiter

Eines der zentralen Ziele des Datenengineerings ist die Aufbereitung von Datensätzen entsprechend den Anforderungen der Nutzer*innen oder der nachfolgenden Prozessschritte. Die Verwendung von Daten kann von der Modellschulung im Bereich des maschinellen Lernens bis hin zu verbesserten internen Unternehmensberichten auf Basis einer integrierten Datenbank reichen. Die Sicherstellung einer ausreichenden Datenqualität ist in allen Fällen zentral. Während die verschiedenen grundlegenden Aspekte der Datenqualität und ihre Bedeutung für Unternehmen in einem früheren Artikel untersucht wurden, stellt dieser Artikel Beispiele für Datenqualitätsprobleme aus der Praxis vor und diskutiert mögliche Lösungen.

Datenformate

Ein Datenfehler stellt immer eine Abweichung von einem Zielwert dar. Das bedeutet, dass mögliche Datenfehler stark von Typ und Format der verfügbaren Daten abhängig sind. Im Wesentlichen muss hier zwischen strukturierten und unstrukturierten Daten unterschieden werden. Unstrukturierte Daten – insbesondere Textdaten – folgen in der Regel keinem Schema, was bedeutet, dass ein Datenfehler nur in seltenen Fällen maschinell erkannt werden kann.

In Textdateien ist ein typisches Beispiel der falsche Lokalisierungsstandard von Gleitkommawerten aufgrund inkonsistenter Verwendung des Dezimaltrennzeichens:

```
1.23
6.4532
7,564
-0.2
```

In diesem Beispiel wird das falsche Dezimaltrennzeichen in der dritten Zeile verwendet, in diesem Fall das Komma, welches in deutschsprachigen Ländern üblich ist. Welches Dezimaltrennzeichen das richtige ist, wird in der Regel durch externe Zusatzinformationen oder durch Bestimmung der Mehrheit innerhalb der gegebenen Daten festgelegt.

Unstrukturierte Textdaten

Mögliche Datenfehler in Textdateien:

- Undefinierte oder abweichende Zeichensätze:
  Ein Zeichensatz beschreibt die Zuordnung von Zeichen (a, b, ä, €, ...) zu ihrer binären Darstellung im Speicher. Ist diese nicht korrekt definiert oder unbekannt, führt dies zu einer falschen Darstellung und Verarbeitung von Sonderzeichen wie deutschen Umlauten.
- Kodierung von Zeilenumbrüchen:
  Ein Zeilenumbruch wird zwischen den Betriebssystemen Microsoft Windows, Apple MacOS und GNU/Linux unterschiedlich dargestellt:
  - In Windows werden hierfür zwei Zeichen verwendet: Eine Sequenz aus Carriage Return (ASCII-Code 13) und Line Feed (ASCII-Code 10).
  - In MacOS wird nur Carriage Return verwendet.
  - In GNU/Linux wird nur Line Feed verwendet.
- Unterschiedliche Lokalisierung der Daten wie deutsche und englische Dezimaltrennzeichen

Binärformate und strukturierte Textdateien

Im Gegensatz dazu basieren strukturierte Daten auf einem Schema, das das Datenformat, die Struktur der Daten sowie die Datentypen und Wertebereiche der enthaltenen Datenwerte enthält. Daten-Schemata können je nach Datenformat explizit oder implizit sein und beschreiben beispielsweise tabellarische Daten pro Spalte:

- Datentyp
- Zellen können Nullwerte enthalten
- Gültigkeitsbereich für numerische Werte
- Format für Zeichenkettenwerte (z.B. Datum und Zeitstempel)

In jedem Fall ermöglicht ein Daten-Schema die Validierung des Inhalts eines Datensatzes oder die Überprüfung auf Fehler. Da es maschinell leichter überprüfbar ist, konzentriert sich dieser Artikel auf strukturierte Daten, wie sie in einem industriellen Umfeld auftreten. Aus der Sicht der Datenvalidierung können strukturierte Daten in zwei grobe Klassen unterteilt werden. Diese unterscheiden sich darin, ob das Format bereits das Daten-Schema bereitstellt:

- Binäre Datenformate mit Schema in den Metadaten bereitgestellt: Beispiele sind Speicherformate kommerzieller Programme wie Microsoft Excel, sowie Bilddateien, standardisierte binäre Protokolle wie OPC UA oder Protobuf, aber auch offene BigData-Formate wie Apache Parquet. Eine weitere sehr typische Klasse von Speicherlösungen, die in diese Kategorie fallen, sind relationale Datenbanken wie Microsoft SQL Server, PostgreSQL oder MySQL.
- Strukturierte Textdateien ohne Schema-Information im Datenformat:
  - Komma-separierte Werte (CSV)
  - XML-Dateien
  - JSON-Dateien

Fehlerursache in strukturierten Textdaten

Während XML- oder JSON-Dateien selten syntaktische Fehler enthalten, da sie in der Regel programmatisch erzeugt werden, treten Datenfehler häufiger in CSV-Dateien auf, da diese oft manuell gepflegt werden (z.B. in Microsoft Excel). Typische Ursachen für Datenformatinkonsistenzen in CSV-Dateien sind, dass es keine explizite Spezifikation des Formats gibt und Fehler beim manuellen Übertragen des Formats von vorherigen Zeilen auftreten können. Typische Beispiele sind:

- Inkonsistente Verwendung von Anführungszeichen für Zeichenkettenfelder
- Unterschiedliche Lokalisierung (z.B. Punkt oder Komma als Dezimaltrennzeichen)
- Leere Spalten und unterschiedliche Anzahl von Spalten pro Zeile
- Unterschiedliche Zeichenkettenrepräsentation von Zeitstempeln, Datum- und Zeitfeldern
- Numerische Werte unter Anführungszeichen
- Schwankende Genauigkeit für numerische Einträge von Integer zu Double

Abweichungen in der Datenrepräsentation können nicht nur durch menschliche Fehler bei der manuellen Dateneingabe auftreten, sondern auch durch Prozessänderungen bei der automatisierten Datengenerierung. Insbesondere bei CSV- und JSON-Dateien ist es oft schwierig, den Typ eines Dateneintrags zu bestimmen, besonders wenn die Quelldaten nicht konsistent befüllt sind. Dieselben Fehlerkategorien können hier auftreten wie bei der manuellen Datenübertragung.

Kategorien von Datenfehlern in strukturierten Daten

Je nach Art der strukturierten Daten können verschiedene Kategorien von Datenfehlern auftreten:

Verstoß gegen die Datensyntax

Diese Fehlerkategorie nimmt im Vergleich zu den folgenden eine Sonderrolle ein, da sie in der Regel nur in strukturierten Textdateien auftreten kann, da Binärdateien fast ausnahmslos algorithmisch erzeugt werden und somit normalerweise syntaktisch korrekt sind.

Ein Syntaxfehler tritt auf, wenn die Textdatei nicht der vorgegebenen Syntax des gewünschten Dateiformats folgt. Beispiele hierfür sind:

- fehlende schließende Tags in XML- oder HTML-Dateien
- falsche Anzahl von Spalten in einer CSV-Datei
- falsches Format eines Datums- oder Zeitstempels in einer Textdatei
- fehlende, überzählige oder falsche Anführungszeichen
- Falsche Lokalisierung wie z. B. Komma statt Punkt als Dezimaltrennzeichen

Fehlerhafte Datentypen

Dieser Fehler tritt auf, wenn ein zu überprüfendes Feld einen falschen Datentyp hat. Typische Beispiele sind:

- Text in einem Feld, in dem numerische Werte erwartet werden.
- Angabe zu weniger strengen Datentypen in Binärformaten, z.B. die Definition eines Textfeldes, in dem semantisch ein Fließkommawert erwartet wird.

Fehlende Daten

Datenschemata erlauben es oft, Datenfelder als optional zu kennzeichnen, so dass es möglich ist, dass Datenfelder auch in strukturierten Binärdateien leer bleiben. Diese sind jedoch für die darauf aufbauende Anwendungssemantik notwendig, wie z.B. Felder, die als Fremdschlüssel zur Verknüpfung von Tabellen verwendet werden sollen. Wenn Daten aus einem strukturierten Textdatenformat eingelesen werden sollen, kommt es viel häufiger als bei Binärdaten vor, dass Daten fehlen. Ein klassisches Beispiel hierfür ist eine fehlende Spalte in einer CSV-Datei.

Fehlende Metainformationen

Ein typisches Beispiel für fehlende Metainformationen ist die Angabe einer Uhrzeit ohne Zeitzone. Die Speicherung der Ortszeit ohne Angabe der Zeitzone kann je nach Art der Speicherung sogar zu Datenverlusten führen, da es bei der Umstellung ›

von Sommer- auf Winterzeit zu doppelten Zeitstempeln für verschiedene Zeitpunkte kommt. Ein weiteres Beispiel für fehlende Metainformationen ist die Angabe eines Datentyps, wenn dieser nicht eindeutig aus dem Datenelement abgeleitet werden kann. Ein Beispiel hierfür ist die folgende Darstellung in einer CSV-Datei:

```
... ;10.3352;
...
```

Ein solches Feld wird normalerweise als Fließkommawert interpretiert und gespeichert. Es gibt jedoch unterschiedliche Datentypen für einfache oder doppelte Genauigkeit. Welcher Datentyp zu wählen ist, hängt davon ab, welche Wertebereiche (z.B. Minimal- und Maximalwerte) in der Gesamtdatenmenge enthalten sind. Wenn alle Daten bereits vorhanden sind, können diese programmatisch abgeleitet werden. Werden die Daten jedoch erst nach und nach geliefert, ist es sicherer, sich für den Datentyp mit dem größeren Wertebereich zu entscheiden. Der Nachteil dabei ist natürlich der doppelte Speicherbedarf für das Datenfeld.

Verstoß gegen den semantischen Geltungsbereich

Diese Fehler beschreiben Werte, die außerhalb ihres Gültigkeitsbereiches liegen, obwohl sie einen gültigen Wert für ihren Datentyp haben. Ein Beispiel hierfür sind Außentemperaturen von über 100° Celsius in Mitteleuropa.

Falsche Reihenfolge

Diese Fehlerkategorie beschreibt die Speicherung von Daten in falscher Reihenfolge. Dies kann z.B. eine Zeitreihe von Sensorwerten sein, die nicht in aufsteigender Reihenfolge nach Zeitstempel sortiert gespeichert wurde. Solange der Zeitstempel für jeden Wert vorhanden ist, kann ein solcher Datensatz noch korrekt eingelesen werden, aber oft werden Zeitstempel nicht explizit gespeichert, um Speicherplatz zu sparen, wenn die Sensorwerte durch regelmäßige Stichproben ermittelt wurden. In einem solchen Fall reichen die Startzeit und der zeitliche Abstand zwischen zwei Messpunkten aus, um alle Zeitstempel zu ermitteln – wenn die Reihenfolge der Speicherung stimmt. Ein weiteres Beispiel, bei dem die Reihenfolge der Speicherung für die Semantik entscheidend ist, ist die sequentielle Speicherung von Messdatenpunkten in einer Textdatei, die auf einem regelmäßigen Gitter angeordnet sind und deren Positionierung auf dem Gitter sich implizit aus Startposition und Schrittweite entlang der Achsen des Koordinatensystems ergibt.

Formatänderungen für kontinuierlich gelieferte Daten

In der Praxis ist dies eines der größten Probleme mit der Datenqualität. Wenn die Daten in einem einheitlichen Format geliefert werden, kann die Verarbeitung der Daten an dieses Format angepasst werden und auch gewisse wiederkehrende Schwankungen in der Datenqualität auffangen. Ändert sich jedoch das Format der gelieferten Daten abrupt, muss die Datenverarbeitung in der Regel angepasst werden. Typischerweise handelt es sich dabei um wegfallende oder hinzukommende Datenfelder, Änderungen der Datentypen oder des Datenformats. Aus Sicht der Datenqualität gibt es im Grunde keinen Unterschied zwischen Daten, die in einem Block geliefert werden und ein uneinheitliches Format haben, und Daten, die im Laufe der Zeit als Datenstrom geliefert werden und ihr Format mit der Zeit ändern. Der Unterschied für den Datenempfänger besteht jedoch darin, dass man bei bereits vorhandenen Datenfehlern den Datenimport sofort darauf abstimmen kann, während bei kontinuierlicher Datenlieferung Änderungen oft unerwartet auftreten.

```
timestamp;temperature_heater;temperature_boiler;pressure_boiler;rpm;power_dynamo;power_heating;valve_aperture;water_level
2019-07-20T11:38:03;26.093750;48.555557;193.544373;0.000000;-0.001262;0.0;0.0;190
2019-07-20T11:38:04;26.093750;48.555557;180.865280;0.000000;-0.001262;0.0;0.0;190
2019-07-20T11:38:05;26.093750;47.416672;193.544373;0.000000;-0.001262;0.0;0.0;190
...
2019-07-20T11:38:58;26.093750;48.555557;206.114639;0.000000;-0.001262;0.0;0.0;190
2019-07-20T11:38:59;26.093750;47.416672;206.114639;0.000000;-0.001262;0.0;0.0;190
2019-07-20T11:39:00;26.093750;48.555557;206.114639;12.000000;-0.001262;446.973846;0.0;190
2019-07-20T11:39:01;26.093750;49.694443;193.489960;12.000000;-0.001262;442.720520;0.0;190
2019-07-20T11:39:02;26.093750;50.833328;206.060226;0.000000;-0.001262;446.973846;0.0;190
2019-07-20T11:39:03;26.093750;49.694443;193.435562;0.000000;-0.001262;446.973846;0.0;190
...
2019-07-20T11:46:09;35.774303;279.750000;1494.212524;0.000000;-0.006040;459.733795;0.0;190
2019-07-20T11:46:10;35.774303;276.333313;1494.212524;0.000000;-0.006702;459.733795;0.25;190
2019-07-20T11:46:11;35.774303;279.750000;1519.461914;0.000000;-0.006702;459.733795;0.25;
2019-07-20T11:46:12;35.774303;279.750000;1519.516235;0.000000;-0.006702;459.733795;0.25;
```

Beispieldatensatz

Typische strukturierte Daten aus dem industriellen Umfeld sind Zeitreihen von Sensordaten. Als Beispiel kann hier die Zeitreihe von Messdaten einer Wärmekraftmaschine dienen, die in Auszügen dargestellt ist. Diese Daten wurden direkt aus dem Betrieb der Maschine über Sensoren entnommen und von einem Programm, das auf einem Raspberry Pi Minicomputer läuft, in der CSV-Datei gespeichert, was in Bezug auf die Datenqualität als durchaus repräsentativ für Industriedaten angesehen werden kann. In dieser CSV-Datei sind einige der oben gezeigten Datenfehler offensichtlich:

- Der Zeitstempel in der ersten Spalte enthält keine Zeitzone
- In der Spalte power_dynamo werden negative Werte angezeigt
- In den letzten beiden Zeilen fehlt der Wert für water_level

Methoden der Fehlersuche bei Daten

Ein naheliegender – und guter – Ansatz zur Behebung von Datenqualitätsmängeln besteht darin, eine verbesserte Version der Daten vom Datenlieferanten anzufordern. In der Praxis ist dieser Ansatz jedoch oft nicht durchführbar. Wenn zum Beispiel in einer Produktionslinie fehlerhafte Sensordaten aufgezeichnet wurden, weil ein Sensor defekt ist, ist es oft nicht möglich oder zumindest sehr kostenintensiv, die Datenaufzeichnung zu wiederholen. Während der reale monetäre Wert der erhobenen Daten für die Projektbeteiligten zu Beginn oft nicht abschätzbar ist, lassen sich die direkt anfallenden Kosten für eine Wiederholung einer Messung – z.B. aufgrund einer Produktionsunterbrechung – sehr schnell beziffern. Darüber hinaus kann auch der Austausch eines defekten Sensors durch die oft notwendige Einschaltung von Fremdfirmen zu einer erheblichen Verzögerung der geplanten Datenanalysen führen. Ein typisches Szenario ist hier die Sammlung von ausreichenden Trainingsdaten für Machine-Learning-Modelle, die oft Monate dauern kann. Hier kann eine möglicherweise mehrwöchige Verzögerung durch den Austausch eines Sensors den gesamten Projektablauf gefährden, ohne dass der tatsächliche Nutzen eines solchen Eingriffs im Vorfeld klar ist. Aus diesen Gründen ist die algorithmische Behandlung des Datenfehlers oft die insgesamt günstigste Lösung.

Algorithmische Wiederherstellung von Datenfehlern

Leider gibt es keine allgemein anwendbaren Methoden, um Quelldaten immer in das gewünschte Zielformat zu bringen. Generell lässt sich aber sagen, dass die umfassende Verfügbarkeit von Metainformationen oder die Verwendung eines strukturierten Binärformats für die Quelldaten den Aufwand für die Datenvalidierung stark reduziert. Der gewünschte Typ eines Datenfeldes ist bereits bekannt, so dass die Daten nicht fehlerhaft gespeichert werden können. Der häufigste Fehler bei Binärdaten sind daher fehlende Daten, wenn sich das zugrunde liegende Schema geändert hat oder ein Datenfeld als optional angegeben wurde, obwohl es für die Anwendungslogik benötigt wird. Generell lässt sich sagen, dass Datenfehler bei strukturierten Binärdaten in der Regel auf Fehler im Datenschema oder auf eine ungeplante Änderung desselben zurückgeführt werden können.

Werden strukturierte Textdaten als Datenquelle verwendet, kommen – wie oben beschrieben – weitere Klassen von möglichen Fehlern hinzu. ▸

Explizite Schemainformationen in Textdaten

Bei strukturierten Textdateien können jedoch Schemainformationen per Konvention aufgenommen werden, z. B. in die Kopfzeile der CSV-Datei, die die Namen der Spalten enthalten kann. Als Erweiterung dieser üblichen Methodik kann man die Kopfzeile um die Datentypinformationen erweitern, um sicherzustellen, dass der richtige Zieldatentyp verwendet wird:

```
timestamp:java.sql.Timestamp;pressure_boiler:java.lang.Double;rpm:java.lang.Double;valve_aperture:java.lang.Double;water_level:java.lang.Integer
```

Im obigen Beispiel werden die entsprechenden Java-Datentypen angegeben, wobei es natürlich vom Zieldatenspeichersystem abhängt, welche Datentypen zur Speicherung zur Verfügung stehen. In der Regel ist es jedoch ausreichend, den Datentyp für eine Datenbank oder Programmiersprache eindeutig zu definieren, da dann die Konvertierung für andere Zielsysteme automatisch erfolgen kann.

Automatisierte Datenfehlerbehebung

Wie kann man im Rahmen eines automatisierten Prozesses auf Datenfehler reagieren? Wenn Daten als Teil eines ETL-Prozesses (Extrahieren – Transformieren – Laden) unter Verwendung eines bestimmten Schemas gespeichert werden sollen und ein Datensatz die Anforderungen des Datenschemas nicht erfüllt, besteht die einfachste Methode darin, diesen Datensatz zu verwerfen. Dies kann für einige Anwendungsfälle – wie z. B. große Datensätze für das Training von KI-Modellen – angemessen sein, aber im Allgemeinen besteht das Ziel darin, einen Datensatz so umzuwandeln, dass er im vorgesehenen Schema gespeichert werden kann. Die folgenden Methoden können verwendet werden, um dies automatisch zu tun:

- **Schema-Evolution oder optionale Typfelder:** Schemaevolution beschreibt die Möglichkeit der Versionierung eines Schemas, wobei Daten, die mit einer früheren Version des Datenschemas gespeichert wurden, mit dem neuen Schema verarbeitbar bleiben. Eine Schemaevolution kann das Hinzufügen, Entfernen und die Typkonvertierung von Datenfeldern beinhalten. Ein gutes Hilfsmittel hierfür sind optionale Datenfelder, die es beispielsweise ermöglichen, neue Felder hinzuzufügen und dennoch vorhandene alte Daten korrekt zu verarbeiten. Optionale Datenfelder sind auch eine gute Möglichkeit, leere Datenfelder korrekt zu speichern, ohne dass der gesamte Datensatz verworfen werden muss.
- **Implizite Typkonvertierung:** Wenn ein Quelldatentyp automatisch und ohne Genauigkeitsverlust in den Zieldatentyp konvertiert werden kann, kann dies im ETL-Prozess automatisch geschehen:
- **Dateninterpolation für fehlende Werte in Zeitreihen:** Dies ist ein offensichtlicher Vorgang, aber es hängt stark von der beabsichtigten Verwendung der Daten ab, ob ein solcher Vorgang zulässig ist.

Wenn Datenfehler nicht automatisch korrigiert werden können, ist es sinnvoll, die Rohdaten aufzubewahren und z.B. eine Meldung zu senden, damit der Fehler untersucht und der ETL-Prozess – ggf. nach manueller Korrektur – abgeschlossen werden kann. Handelt es sich nicht um einen einmaligen Fehler, wird der ETL-Prozess in der Regel im Zuge der Untersuchung und Behebung des Problems angepasst, um den Fehler in Zukunft zu beseitigen. Dies gilt insbesondere für Fehler, die auf eine Änderung des Datenquellenformats zurückzuführen sind.

Vermeidung von Datenverlusten

Wenn die Quelldaten kontinuierlich über einen Streaming-Prozess geliefert werden, ist es besonders wichtig, die Rohdaten zunächst zu speichern, bevor sie weiterverarbeitet werden. Dies kann verhindern, dass die Daten verloren gehen, wenn die Datenverarbeitung zu einem späteren Zeitpunkt im ETL-Prozess fehlschlägt. Nachdem ein Fehler behoben oder der ETL-Prozess nach einer Formatänderung angepasst wurde, können die gespeicherten Rohdaten erneut verarbeitet werden. Meistens verbrauchen die Rohdaten mehr Speicherplatz, insbesondere wenn sie als Textdateien geliefert werden, im Vergleich zu einer späteren strukturierten und komprimierten Speicherung. Daher ist es oft ratsam, die erfolgreich verarbeiteten Rohdaten nach der Validierung zu löschen. Um Speicherplatz zu sparen, können Rohdaten natürlich auch mit Standardalgorithmen komprimiert werden, was vor allem bei Textdaten zu erheblichen Speicherplatzeinsparungen führt.

Die Rolle der Datenqualität bei der Projektplanung

Vor Beginn eines Data-Science- oder Data-Engineering-Projekts ist es für alle Beteiligten oft schwierig, die Qualität der einzubeziehenden Daten einzuschätzen. Dies liegt oft daran, dass die Daten bereits über einen gewissen Zeitraum gesammelt, aber noch nicht operativ genutzt werden, weil sie zum Beispiel noch nicht in ausreichender Menge vorliegen.

Darüber hinaus stellt die Anhebung der Datenqualität auf ein für die Projektziele erforderliches Niveau oft einen erheblichen Teil des Projektaufwands dar, der ohne Kenntnis der Daten oder ihrer Qualität nur schwer abzuschätzen ist. Um diesem Problem zu begegnen, kann beispielsweise eine Vorprojektphase zur gemeinsamen Klärung der Ausgangssituation eingeplant oder ein agiler Ansatz gewählt werden, der ein schrittweises gemeinsames Vorgehen mit flexibler Definition von Meilensteinen ermöglicht.

Die RISC Software GmbH ist mit ihrer über zehnjährigen Expertise im Bereich Data Engineering ein zuverlässiger Beratungs- und Implementierungspartner, unabhängig vom Anwendungsbereich. ◆

Autor

DI Paul Heinzlreiter
Senior Data Engineer

Abra CaTabRa: Daten automatisiert analysieren, validieren und damit Machine Learning Modelle trainieren

von Sophie Kaltenleithner, MSc

Daten werden mittlerweile in fast allen Lebensbereichen gesammelt – seien es die gekauften Produkte beim Online-Shopping, Bewegungs- und Ernährungsinformationen in Fitness-Apps oder Maschinendaten beim Produktionsprozess. Häufiges Ziel davon: Automatisch Vorhersagen zu treffen: Welchen Zielgruppen soll mein Produkt vorgeschlagen werden? Welche Gewichtsabnahme kann ich erwarten, wenn ich täglich eine Runde laufe? Wann muss ich die Verschleißteile meiner Maschinen tauschen, um möglichst kurze Stillstandszeiten zu haben?

Damit solche Vorhersagen möglich werden, bedarf es komplexer Analysetätigkeiten und technischer Expertise. Nicht immer kann dieser Aufwand in Projekten investiert werden. CaTabRa schafft hier Abhilfe: CaTabRa ist ein Open-Source-Tool zur Automatisierung von Schritten in der Analyse von tabellarischen Daten und der Entwicklung von Vorhersagemodellen. Es eignet sich sowohl für Domänenexpert*innen ohne technischem Know-how, als auch für Data-Scientists, die effizient Informationen aus ihren Daten gewinnen möchten. Statistische Auswertungen, Training von Machine Learning Modellen, Erklärung von Modellentscheidungen, Validierung von Inputdaten. All das ist mit geringem Zutun erledigt!

Anwendungsbeispiel aus der Medizin: Covid-19 Detektion in Bluttests

Kann man COVID19 anhand von Werten eines Standard-Bluttests diagnostizieren? Mit diesem Thema beschäftigten sich Forscher*innen von JKU, KUK MedCampus III und RISC Software GmbH im Jahr 2022 [1] . Ziel war es, Covid-19 Infektionen aus routinemäßig durchgeführten Labortests nachzuweisen, um so eine große Anzahl von Patient*innen schnell und ohne Mehraufwand testen zu können. Wie ähnliche Ergebnisse allein mithilfe von CaTabRa generiert werden können, wird im Folgenden demonstriert.

CaTabRa arbeitet auf tabellarischen Daten. Zeilen sind dabei einzelne Stichproben (Samples) und Spalten deren Charakteristika (Features). In unserem Beispiel sind die Samples Patienten und die Features ihre Blutwerte. Zusätzlich muss der Zielwert definiert sein. Das kann ein numerischer Wert sein (Regression) oder – wie hier – ein kategorialer (Klassifikation): „infiziert" und „nicht infiziert".

Ein typischer Workflow besteht aus Anwendung der folgenden vier Schritte, die über Kommandozeilen-Befehle aufgerufen werden können:

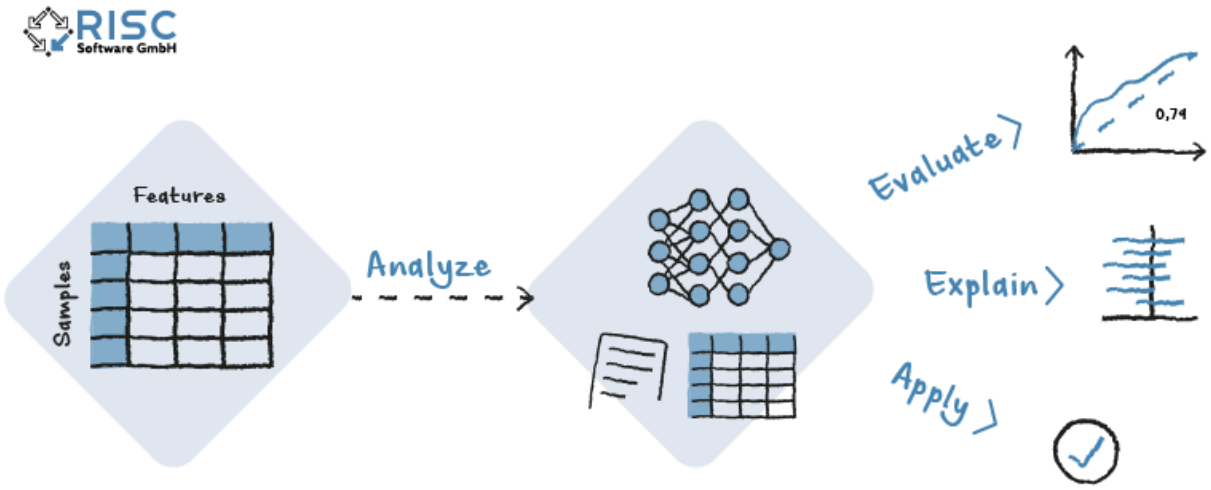


Abbildung 1: Der typische Workflow bei Anwendung von CaTabRa besteht aus den vier Schritten Analyze, Evaluate, Explain und Apply.

1) Analyze Erstellt Statistiken und trainiert Vorhersagemodelle.

```
python -m catabra analyze covid_data.h5 --classify label --split_split --group patient_internal_id --ignore case_internal_id timestamp month --time 120 --out resoult_dir
```

2) Evaluate Evaluiert die Modelle auf einem Testdatensatz, um ihre Qualität zu überprüfen.

```
python -m catabra evaluate result_dir --on covid_data.h5 --split_split --bootstrapping-repetitions 1000
```

3) Explain Generiert Erklärungen für Modellentscheidungen in Form von Feature-Importance Scores.

```
python -m catabra explain result_dir --on covid_data.h5 --split_split
```

4) Apply Trifft Vorhersagen für neue Samples durch Anwenden der zuvor trainierten Modelle.

```
python -m catabra apply result_dir -on new_data.h5
```

	feature	count	mean	std	min	25%
0	age	127115	54.147479	26.023760	0.0	35.663462
1	_HAEMATOKRIT	122970	37.119061	6.790249	1.3	32.600000
2	_MCH	121631	29.757381	2.619814	14.0	28.400000
3	_HAEMOGLOBIN	121614	12.487140	2.455372	3.1	10.800000
4	_LEUKOZYTEN	121608	8.695341	6.134663	0.0	5.980000

	feature	age	_HAEMATOKRIT	_MCH	_HAEMOGLOBIN	_LEUKOZYTEN
0	age	1.000000	-0.111429	0.089873	-0.164331	-0.042021
1	_HAEMATOKRIT	-0.111429	1.000000	0.173223	0.963975	0.017756
2	_MCH	0.089873	0.173223	1.000000	0.263880	-0.013019
3	_HAEMOGLOBIN	-0.164331	0.963975	0.263880	1.000000	0.006117
4	_LEUKOZYTEN	-0.042021	0.017756	-0.013019	0.006117	1.000000

Abbildung 2: Exemplarischer Auszug aus den Covid-19 Daten.

Datenanalyse und Training

Im ersten Schritt generiert Analyze deskriptive Statistiken, die einen besseren Überblick über die Daten geben sollen. Diese werden pro Feature berechnet. Je nach Datentyp sind das etwa die Anzahl an Einträgen im Datensatz, Extremwerte, Mittelwerte, Korrelationen mit anderen Spalten etc. Die Tabellen in Abbildung 2 zeigt dies exemplarisch für ausgewählte Features der Covid-19 Daten.

Im zweiten Schritt wird ein Modell trainiert, das den definierten Zielwert vorhersagt – also hier ob eine Covid-19 Infektion vorliegt. Die Qualität von Machine Learning Modellen hängt stark von den verwendeten Algorithmen und deren Konfigurationen ab. Diese können im Vorfeld nicht ohne Weiteres festgestellt werden. CaTabRa setzt deshalb auf State-of-the-Art AutoML Methoden, um schnell und ohne großen manuellen Aufwand die richtige Konfiguration zu finden. AutoML steht für „Automated Machine Learning". Dabei werden komplizierte Optimierungsverfahren eingesetzt, um sich schrittweise der besten Lösung anzunähern – ganz ohne manuellen Aufwand.

Nach Beendigung des Trainings kann über die Evaluate Funktionalität die Qualität des Modells überprüft werden. Dabei werden detaillierte Performance-Reports und entsprechende Visualisierungen erstellt. Die Auswertung wird mithilfe eines Teil des Covid-19 Datensatzes durchgeführt, der nicht zum Trainieren verwendet wurde. So kann geschätzt werden, wie gut das Modell mit neuen Daten umgehen kann. Untenstehende Abbildung zeigt Beispiele der so erhaltenen Grafiken. Diese visualisieren bestimmte Qualitäts-Metriken in Abhängigkeit der Modell-Vorhersagen. Der erhaltene ROC-AUC Wert – ein Qualitätsmaß für Klassifikationsprobleme – ist vergleichbar gut wie der in der originalen Publikation.

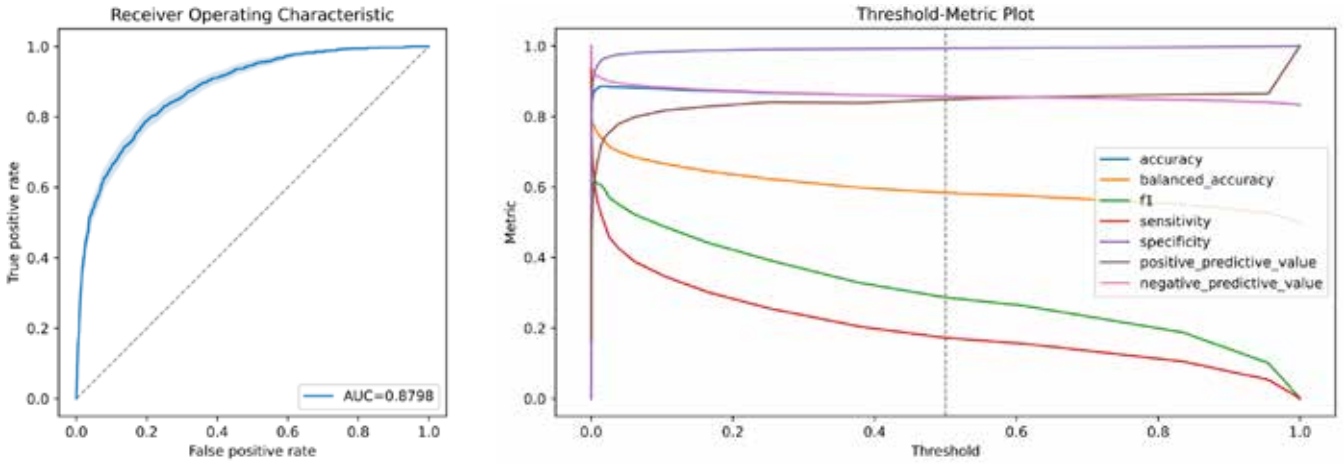


Abbildung 3: Beispielhafte Grafiken zur Modell-Evaluierung
die von CaTabRa generiert werden. Oben: ROC-Kurve; Unten: Metrik-Werte in Abhängigkeit des Decision-Thresholds.

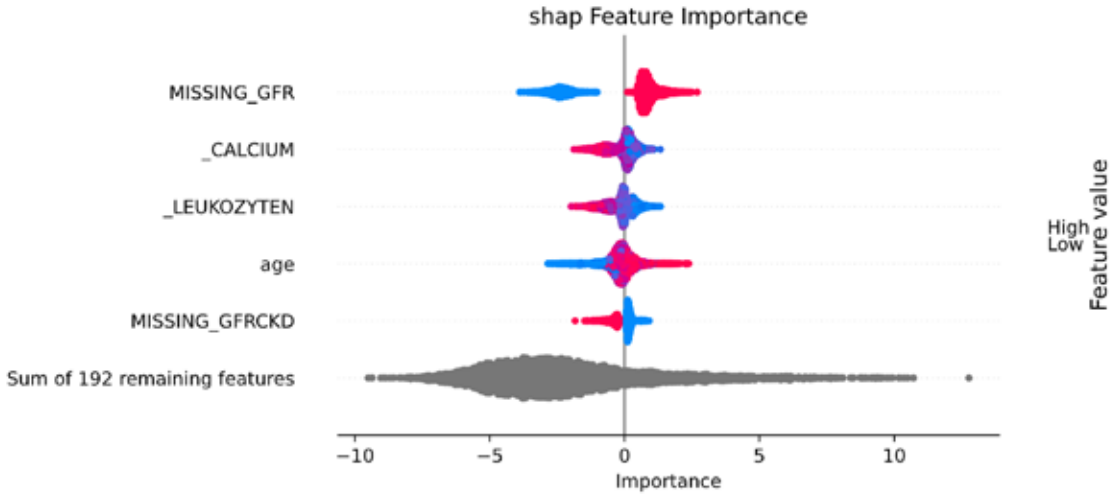


Abbildung 4: Feature-Importance Plot für die Covid-Daten basierend auf SHAP-Werten.

Modellerklärung – Explainable AI

Entscheidungen von Black-Box Machine Learning Modellen sind für Menschen nur schwer nachvollziehbar. Gerade in der Medizin ist es allerdings wichtig, den Modellen nicht blind zu vertrauen. Oft ist etwa ein ungewollter Bias in den Daten vorhanden, der die Modelle dazu veranlasst, falsche Schlussfolgerungen zu ziehen. Wären in den Trainingsdaten etwa zufälligerweise mehr Männer als Frauen an Covid-19 erkrankt gewesen, könnte es sein, dass das Geschlecht der Patient*innen zu stark in die Entscheidung miteinfließt.

CaTabRa erlaubt deshalb mithilfe der Explain Funktion die Wichtigkeit von einzelnen Features festzustellen. Standardmäßig wird dafür SHAP verwendet. Wie diese Methode im Detail funktioniert, lesen Sie im Fachbeitrag Explainable AI. Zusätzlich zu den reinen Berechnungen werden auch hier wieder automatisch aussagekräftige Visualisierungen erstellt.

Die Abbildung 4 zeigt die Feature-Importance Scores der Covid-Daten für die fünf wichtigsten Features. Ein Punkt korrespondiert dabei mit einem Sample, wobei die Farbe den Feature-Wert darstellt (blau: niedrig, rot: hoch). Die Position auf der x-Achse zeigt, wie ein Feature für ein bestimmtes Sample das Ergebnis beeinflusst. Beispielsweise deutet das Fehlen von Messungen der glomerulären Filtrationsrate („MISSING_GFR"; ein Parameter, der v.a. die Nierenfunktion misst) tendenziell auf eine Covid-Infektion hin, und auch hohes Alter scheint für den verwendeten Datensatz ein Indikator zu sein. Insgesamt achtet das Vorhersagemodell jedoch auf viele verschiedene Features, anstatt die Entscheidung an ein paar wenigen Features festzumachen.

Erkennung von ungültigem Input – Out-of-Distribution Detection

Modelle des maschinellen Lernens gehen im Allgemeinen davon aus, dass neu vorherzusagende Daten der Verteilung der ursprünglichen Trainingsdaten entsprechen. In der Realität kommt es allerdings häufig zu so genannten „Domain Shifts", d.h. einer Änderung der Datenverteilung. Der Grund dafür kann vieles sein: Der Trainingsdatensatz war zu wenig repräsentativ, Messverfahren haben sich geändert, Charakteristika ändern sich über die Zeit etc. Jedenfalls sind die Modellentscheidungen in solchen Fällen nicht mehr vertrauenswürdig. CaTabRa trainiert deshalb Out-of-Distribution (OOD) Detektoren um zu überprüfen wie sehr sich ein gewisser Input von den Trainingsdaten unterscheidet. Sie werden bei Aufruf von Apply (also der Vorhersage-Funktionalität) automatisch angewendet. So wissen Anwender*innen wann sie Modell-Vorhersagen besser hinterfragen sollten.

Bei den Covid-19-Daten konnte festgestellt werden, dass Modelle, die nur auf Daten zu Beginn der Pandemie trainiert wurden, zu späteren Zeitpunkten schlechtere Vorhersagen treffen. Das könnte daran liegen, dass der Virus sich allgemein stärker in der Gesellschaft verbreitet hat, aber auch daran, dass neue Mutationen aufgetreten sind. Wenn zum Modelltraining nur Daten der ersten zehn Pandemiemonate verwendet werden und die generierten OOD-Detektoren danach auf Daten angewendet werden die auch die Monate elf und zwölf enthalten, sind bei 81 von insgesamt 95 kontinuierlichen Features geänderte Verteilungen feststellbar.

Fazit – Bei der Vorhersage von Covid-19 ist Vorsicht geboten

Was aus den Ergebnissen der originalen Publikation hervorgeht und sich auch bei Anwendung von CaTabRa wieder zeigt: Bluttests sind ein relativ guter Indikator dafür ob eine Person an Covid-19 erkrankt ist. Allerdings gilt dies nur solange wie man sich sicher sein kann, dass sich an den Eigenschaften des Virus und dessen Verbreitung nicht zuviel ändert. Wie die Meisten aber mitbekommen haben dürften, ist dies in der Realität nicht der Fall. Eine rasche Verbreitung des Virus, Lockdowns und Mutationen könnten alle zu einer geänderten Verteilung führen. Es ist deshalb ratsam die Qualtiät von Machine-Learning Modellen kontinuierlich auf aktuellen Daten zu überprüfen und gegebenenfalls neu zu trainieren. ◆

Vorteile durch den Einsatz von CaTabRa

Es macht die Auswertung von Daten einfacher und effizienter – man kann schnell und einfach einen Einblick in die Daten gewinnen, um bspw. festzustellen, ob der Einsatz von Machine Learning Methoden Sinn macht.

Es erstellt ansprechende Visualisierungen, die man als solche direkt in Publikationen verwenden kann.

Im Gegensatz zu ähnlichen Cloud-Lösungen müssen keine sensiblen Daten hochgeladen werden, alles passiert lokal.

Es wird auf Flexibilität gesetzt: CaTabRa lässt sich einfach erweitern, sodass der Prozess durch eigene Methoden angepasst werden kann. Zusätzlich wird eine Vielzahl von Konfigurationen Out-of-the-Box angeboten.

CaTabRa ist gleichzeitig auch eine Python Bibliothek, die die einzelnen Features, sowie Methoden zur Datenaufbereitung über Programmierschnittstellen zu Verfügung stellt.

Wer CaTabRa ausprobieren möchte, findet es auf GitHub. https://github.com/risc-mi/catabra

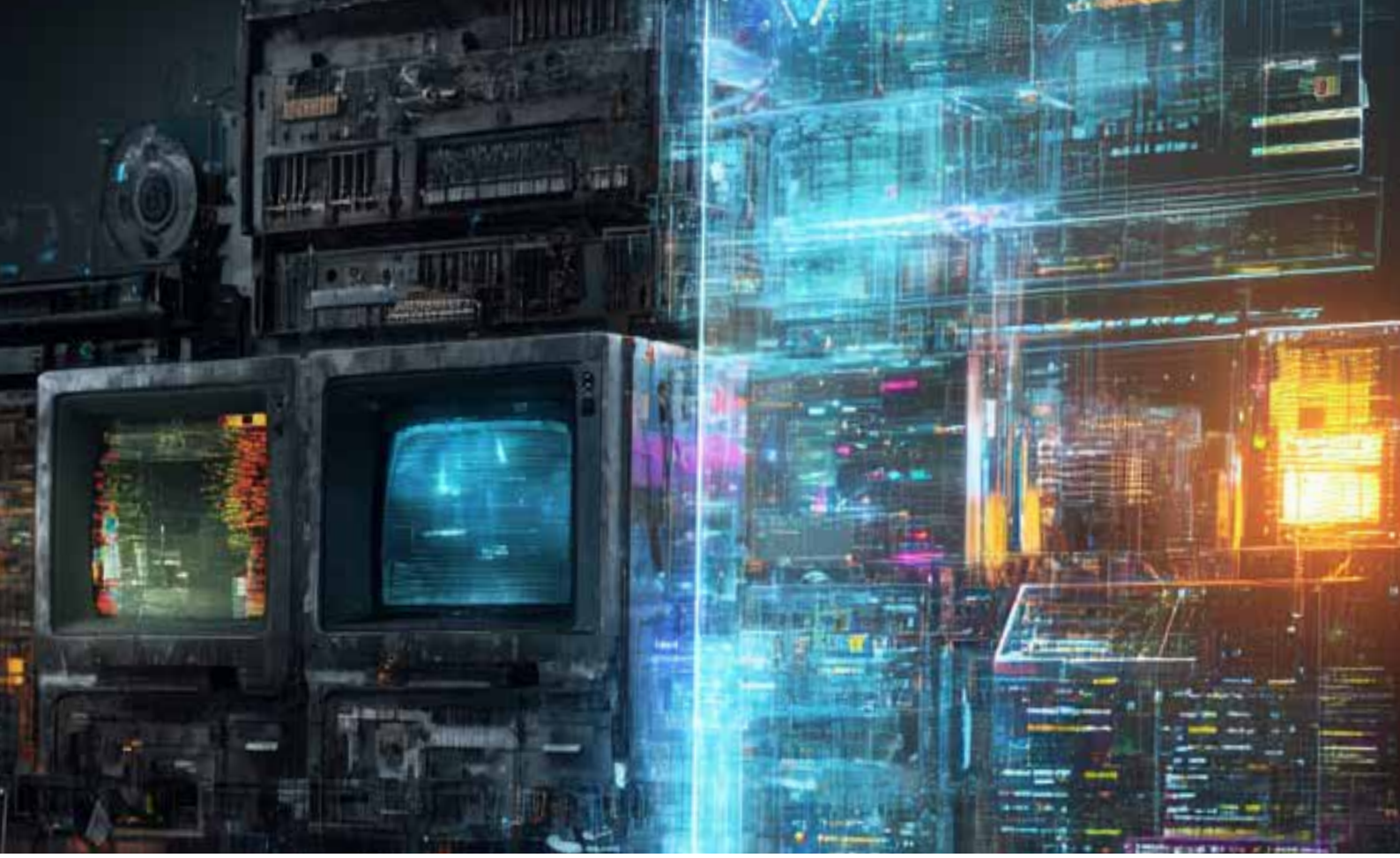
Quellen

[1] T. Roland et al., 'Domain Shifts in Machine Learning Based Covid-19 Diagnosis From Blood Tests', J Med Syst, vol. 46, no. 5, p. 23, Mar. 2022, doi: 10.1007/s10916-022-01807-1.

Autorin

Sophie Kaltenleithner, MSc
Researcher & Developer


PROBLEM
SOLVING
AUTOMAT
Ai
ARTIFICIAL
INTELLIGENCE
MACHINE
LEARNING
ALGORITHM



Reengineering: Ingenieurslösung erstrahlt durch Web-Portierung in neuem Glanz

WARUM BRAUCHT ES AUCH FÜR INGENIEURSLÖSUNGEN EINE WEB-VERSION?

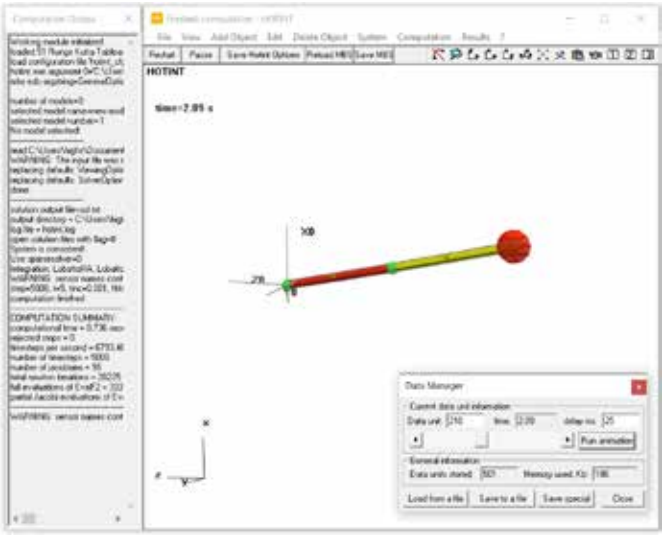
von Martin Hochstrasser MSc, DI (FH)Josef Jank und DI (FH) Alexander Leutgeb

Ein bekanntes Sprichwort sagt: „Nichts ist so beständig wie der Wandel" (Heraklit von Ephesus). Das gilt natürlich auch für Software und insbesondere für dessen Benutzeroberflächen. Diese Anwendungen verschieben sich zunehmend in Richtung Web-Technologien, weil sie dadurch ohne Installation für einen breiten Anwender:innenkreis auf allen Plattformen zur Verfügung stehen. Auch erfolgreiche Desktop-Anwendungen müssen sich diesem Modernisierungsdruck stellen, da sie sonst Gefahr laufen, nicht weiter verwendet zu werden, und damit die Ressourcen für Weiterentwicklungen verlieren. Bei älteren Softwaresystemen münden solche neuen Anforderungen häufig in einer vollständigen Neuentwicklung der Software. Viele sehen es als Chance, um sich von Altlasten befreien und auf die bevorzugte Plattform und Programmiersprache umsteigen zu können. Aufwände und Risiko werden in der anfänglichen Euphorie gerne unterschätzt und viele dieser Projekte überschreiten bei Weitem die anfänglich angenommenen Kosten.

Für eine technisch und wirtschaftlich erfolgreiche Umsetzung von solchen Vorhaben ist daher eine gut überlegte Gesamtstrategie wichtig. Das Ziel sollte eine minimalinvasive Lösung sein, wo nur jene Teile geändert werden, die tatsächlich davon betroffen sind. Für alle anderen Teile gilt: „Never change a running system". Damit werden unnötige Aufwände und Fehlerquellen vermieden. Die Kund*innen erhalten frühzeitig eine einsatzfähige Software, können basierend darauf laufend Feedback geben und damit die Weiterentwicklung in die richtige Richtung lenken. Außerdem wird die Phase der parallelen Wartung der bestehenden Software und Neuentwicklung kurz gehalten. Durch die Modernisierung bzw. das Reengineering ist die Software technologisch fitter für die Zukunft und durch den potenziell breiteren Anwender*innenkreis können zukünftige Investitionen besser argumentiert werden. Durch diesen Reengineering-Ansatz und eine agile Vorgehensweise können die Aufwände gering gehalten und ein gutes Kosten-Nutzen-Verhältnis erreicht werden.

Use-Case: Reengineering der Desktop Anwendung HOTINT für Mehrkörpersimulation

HOTINT ist ein freies Softwarepaket für die Modellierung, Simulation und Optimierung von mechatronischen Systemen, insbesondere von flexiblen Mehrkörpersystemen. Es umfasst Solver für statische, dynamische und modale Analysen, ein modulares objektorientiertes C++-Systemframework, eine umfassende Elementbibliothek und eine grafische Benutzeroberfläche mit Werkzeugen zur Visualisierung und Nachbearbeitung. HOTINT wird seit mehr als 25 Jahren kontinuierlich weiterentwickelt und derzeit von der Linz Center of Mechatronics GmbH (LCM) im Rahmen mehrerer wissenschaftlicher und industrieller Projekte sowohl im Open- als auch im Closed-Source-Bereich weiterentwickelt. Der Schwerpunkt liegt heute auf der Modellierung, Simulation und Optimierung komplexer mechatronischer Systeme (inkl. Steuerung), d.h. allgemeiner mechanischer Systeme in Kombination mit elektrischen, magnetischen oder hydraulischen Komponenten (z.B. Sensoren und Aktoren). Parametrisierte Modell-Setups können über die HOTINT-Skriptsprache oder direkt innerhalb des C++-Modellierungsrahmens implementiert werden. Weitere wichtige Merkmale sind die Unterstützung von Schnittstellen, z.B. eine vielseitige TCP/IP-Schnittstelle, die die Kopplung mit anderen Simulationswerkzeugen ermöglicht (z.B. Co-Simulation mit MATLAB/Simulink oder die Kopplung mit einem Partikelsimulator zur Analyse von Fluid-Struktur- und Partikel-Struktur-Interaktion), sowie die Integration in das Optimierungsframework SyMSpace. Die Software ist als C++ Anwendung realisiert und verwendet zur Umsetzung der Benutzeroberfläche die MFC von Microsoft. Für die Visualisierung der 3D-Darstellung wird die Programmierschnittstelle OpenGL verwendet.



Screenshot 1: alte Desktopanwendung HOTINT

In nur drei Monaten zum möglichst hohen Kundennutzen

Bereits im Zuge der ersten Besprechungen wurden gemeinsam mit LCM die **Rahmenbedingungen** und primären **Ziele** abgesteckt. Generell bestand das übergeordnete Ziel darin, für die bestehende Desktop-Anwendung eine Web-Oberfläche zu realisieren. Relativ unklar war aber für alle Beteiligten, ob die angedachten Lösungsideen im geplanten Zeitraum von 3 Monaten umsetzbar wären, oder ob man an nicht vorhersehbare Grenzen oder Probleme stoßen würde. Daher wurde auch auf eine präzise Definition der finalen Ergebnisse verzichtet, sondern das Risiko durch **Timeboxing** (definierte inkrementelle Produktentwicklung mit festgeleger Zeitdauer) begrenzt. Für das Team bestand die Aufgabe darin, in der verfügbaren Zeit einen möglichst **hohen Kund*innennutzen** zur generieren – im Idealfall den aktuellen Funktionsumfang auch vollständig im Web-Client zur Verfügung zu stellen. Ganz im Sinne des agilen Vorgehens wurde das Risiko zusätzlich durch **zweiwöchige Sprints** mit gemeinsamen **Reviews** und der darauf basierenden **Planung** für die nachfolgenden Sprints minimiert. Der erste Sprint der jeweiligen Phase hatte dabei **Ähnlichkeiten mit Spikes**, wo die grundsätzliche Machbarkeit und der Aufwand für eine technische Story bestimmt wird. Basierend darauf wurde in den folgenden Sprints die konkrete Realisierung vorangetrieben.

Im Nachhinein hat sich dieser **leichtgewichtige Prozess** für das Projekt gut **bewährt**, aber dies muss nicht zwangsweise für andere Projekte gelten. Falls die Kund*innen wenig Erfahrungen im Bereich Agilität und Softwareentwicklung mitbringen, kann natürlich wesentlich mehr Überzeugungsarbeit oder ein formalerer Prozess notwendig sein. Im aktuellen Fall hatte das gut eingespielte Team bereits langjährige Erfahrung im Bereich der 3D-Visualisierung und konnte damit aus der Erfahrung heraus die richtigen Entscheidungen treffen. Gerade bei technisch anspruchsvollen Themen ist dieser Aspekt wichtiger, als ein möglichst gut ausgeklügelter Prozess (ohne dessen Bedeutung kleinreden zu wollen). Für weiterführende Informationen zum Thema möchten wir auf unsere Fachbeiträge Agile vs. klassische Softwareentwicklung, Software-Reengineering: Wann wird das Alt-System zum Problem? und Modernisierung von Software verweisen, da diese Ansätze und Ideen auch hier angewendet wurden.

Mögliche Lösungsansätze

Ausgangsbasis war eine klassische Desktop-Anwendung, die in C++ basierend auf den Microsoft Foundation Classes (MFC) realisiert war. Für die 3D-Darstellung wurde OpenGL verwendet. Das Ziel bestand darin, die Anwendung zusätzlich über einen Web-Browser bedienbar zu machen. Prinzipiell gibt es für eine Portierung der nativen HOTINT C++ Anwendung in eine Webanwendung folgende zwei Möglichkeiten:

Eine reine Client-seitige Webanwendung, die mit Webtechnologien vollständig neu implementiert werden müsste ohne einen Quellcode wiederzuverwenden oder alternativ eine semi-automatisierte Portierung der C++ Anwendung in eine Webanwendung unter Verwendung von WebAssembly und Emscripten.

Als zweiter Ansatz kann eine Client/Server-basierte Webanwendung umgesetzt werden, wobei HOTINT als C++ Server läuft und mit der client-seitigen Webanwendung kommuniziert (Kommunikationsprotokoll für strukturierten Datenaustausch: Parameter, Ergebnisdaten, 3D-Visualisierung, Maus-Interaktion). Die 3D-Visualisierungen können dabei entweder in Form strukturierten Daten übertragen (komplette Übertragung oder nur von sich geänderten

Teilen) oder direkt in Form eines Bildes (Image Streaming).

Eine rein client-seitige Webanwendung wurde vorab ausgeschlossen, da eine vollständige Neuimplementierung den Projektrahmen gesprengt hätte. Die semi-automatisierte Portierung war aufgrund der Verwendung von Closed-Source Bibliotheken von Intel, Microsoft und weiteren nicht möglich. Eine weitere Möglichekit wäre die Realisierung auf Basis von ParaView, da dort bereits eine Client/Server basierte Web-Anwendung (Link zu ParaView Webanwendung) angeboten wird. Dabei gibt es zwei Möglichkeiten, wie die 3D-Visualisierung erfolgen kann:

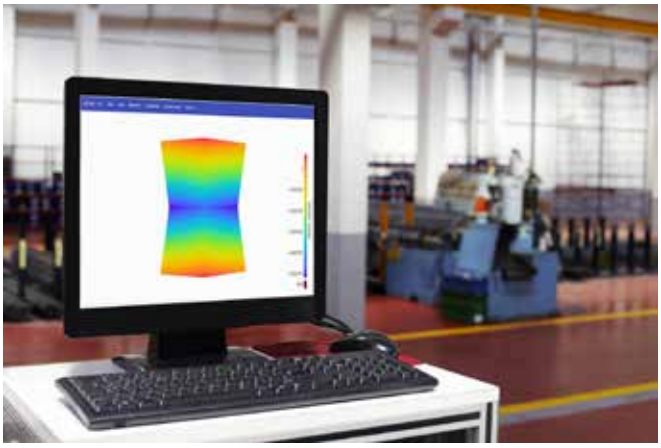
- Rein client-seitiges Rendern der 3D-Szene
- Server-seitiges Rendern der 3D-Szene und Übertragen des Bildes zum Client (Image Stream).

Es besteht auch die Möglichkeit, beide Varianten zu benutzen und im Client eine Überlagerung durchzuführen (hybride 3D-Visualisierung). ParaView verwendet zur Visualisierung die Bibliothek VTK. Die notwendige Infrastruktur für 3D-Visualisierung im Web auf Basis einer Client/Server Architektur wird bereits von VTK zur Verfügung gestellt (Link zu vtkWeb). Dabei kann die 3D-Visualisierung für den Client rein am Server (Image Stream), rein am Client oder auch auf beiden (hybrid) berechnet werden. Die Entscheidung für eine Variante hängt dabei von der Komplexität der zu übertragenden 3D-Szene und dem damit verbundenen Datenaufkommen ab.

Umsetzungskonzept

Für den gewählten Lösungsansatz, in Form einer ClientFür den gewählten Lösungsansatz, in Form einer Client/Server-Architektur, musste das Rendern der 3D-Szenen adaptiert werden. Im Lösungsansatz bestand die Idee darin, das clientseitige Rendering ausschließlich mithilfe der VTK-Bibliothek zu realisieren. Im Zuge der Umsetzung stellte sich aber heraus, dass die interne Rendering-Schnittstelle nicht ohne großen Aufwand auf eine abstrakte Beschreibung für VTK umgestellt werden konnte. Da der Simulationskern im Zuge der Animation in jedem Zeitschritt die 3D-Visualisierung aktualisiert, wäre die Abbildung auf eine andere Rendering API mit Performanz-Einbußen verbunden gewesen.

Daher wurde folgende Vorgehensweise gewählt: Die Verwendung von OpenGL wurde unverändert beibehalten, aber anstelle der Ausgabe am Bildschirm wird ein Bild generiert (Off-Screen Rendering) und vom Server zum Client übertragen. Dieser Ansatz bietet folgende Vorteile:Server-Architektur, musste das Rendern der 3D-Szenen adaptiert werden. Im Lösungsansatz bestand die Idee darin, das clientseitige Rendering ausschließlich mithilfe der VTK-Bibliothek zu realisieren. Im Zuge der Umsetzung stellte sich aber heraus, dass die interne Rendering-Schnittstelle nicht ohne großen Aufwand auf eine abstrakte Beschreibung für VTK umgestellt werden konnte. Da der Simulationskern im Zuge der Animation in jedem Zeitschritt die 3D-Visualisierung aktualisiert, wäre die Abbildung auf eine andere Rendering API mit Performanz-Einbußen verbunden gewesen.



Daher wurde folgende Vorgehensweise gewählt: Die Verwendung von OpenGL wurde unverändert beibehalten, aber anstelle der Ausgabe am Bildschirm wird ein Bild generiert (Off-Screen Rendering) und vom Server zum Client übertragen. Dieser Ansatz bietet folgende Vorteile:

- Grundlegendes Rendern in HOTINT unterliegt keinen großen Änderungen.
- Erweiterungen und Anpassungen in den Elementroutinen von HOTINT können in bekannter Art und Weise durchgeführt werden.
- Die Änderungen betreffen nur das neue Web-Backend selbst, wodurch die Wartbarkeit von HOTINT unverändert erhalten bleibt.

Da eine rein bildbasierte Übertragung von 3D-Ansichten nicht für alle Anwendungsfälle aufgrund der Kommunikationsbandbreite zielführend war, wurde für bestimmte Anwendungsfälle die Übertragung von kompletten Modellen konzipiert. Im Falle von FEM (Finite-Element-Method) Modellen inklusive Berechnungsergebnissen werden diese nur einmal vom Server zum Client in Form einer VTK-Szenenbeschreibung übertragen. Dadurch kann im Client interaktiv einerseits die 3D-Ansicht verändert und andererseits die Visualisierung unterschiedlicher Ergebnissgrößen ausgewählt werden, ohne dass Daten erneut vom Server übertragen werden müssen.

Für die Anbindung der Web-Clients wurde ein Backend-For-Frontend (BFF) auf Basis von Node.js realisiert. Dieser Ansatz wurde gewählt, da die Integration eines vollständigen Webservers im Widerspruch zur minimalinvasiven Anpassung der bestehenden Anwendung gewesen wäre und darüber hinaus Node.js gute Unterstützung für gRPC und GraphQL bietet. Das Backend stellt dem BFF die Daten über gRPC in generischer Form zur Verfügung und dieses wiederum liefert die Daten über GraphQL weiter an das Frontend. Das BFF übernimmt einerseits die Abbildung der unterschiedlichen Protokolle und andererseits können auch intelligente Caching Mechanismen implementiert werden. Durch die Verwendung des BFF können die Erweiterungen im Backend (HOTINT) gering gehalten werden. Die folgende Abbildung zeigt das Umsetzungskonzept.

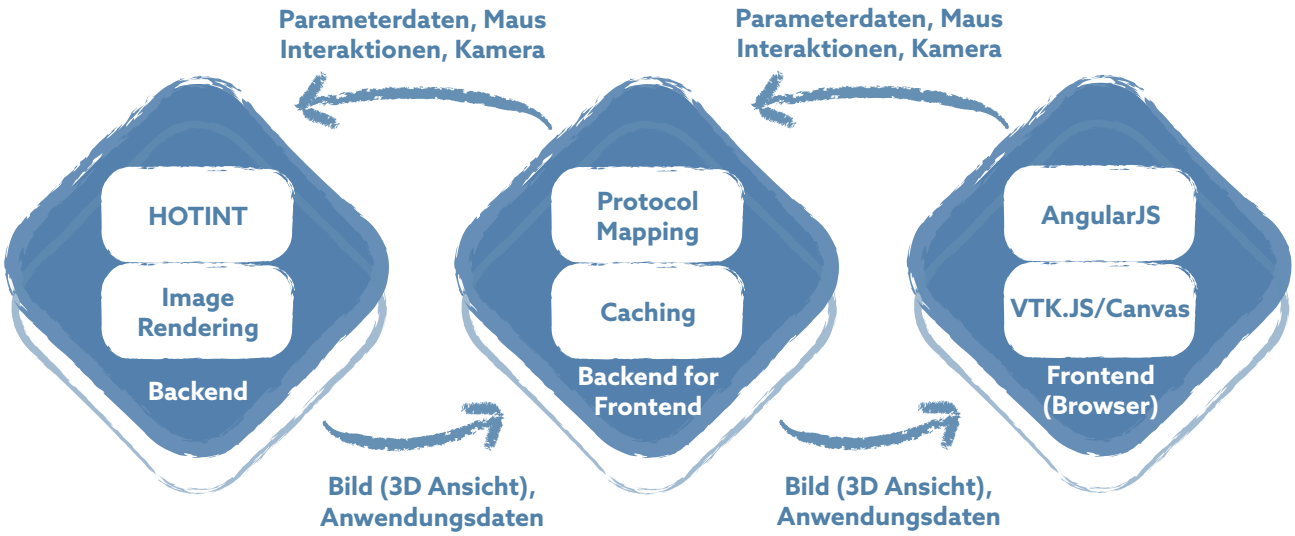


Abbildung 1: Umsetzungskonzept

Technische Umsetzung und Implementierung

Neue Erkenntnisse im Zuge der Implementierung hätten Änderungen und Anpassungen des geplanten Umsetzungskonzepts erforderlich machen können. Im konkreten Fall war dies überraschenderweise nicht erforderlich – das Konzept konnte ohne größere Abweichungen wie geplant implementiert werden. In HOTINT waren letztendlich nur folgende Änderungen notwendig:

- OpenGL Off-Screen Rendering
- Export / Import der Parameter-Einstellungen als JSON
- Erweiterung um VTK-Szenengenerierung für Visualisierung von FEM Berechnungsergebnissen

Der bisherige Render-Code mit OpenGL musste nicht angepasst werden. Die Code-Basis von HOTINT hat einen Umfang von ungefähr 200.000 LOC (Lines of Code) mit ~300 Klassen. Die HOTINT-Web-Erweiterung hat ca. 2.000 LOC, während die Anpassungen an HOTINT mit ~500 LOC (0.25 %) sehr gering ausgefallen sind.

Das **Backend** wurde wie geplant mit gRPC implementiert. Die Einzelbilder für die bild-basierte Übertragung der 3D-Darstellung werden mit Offscreen-Rendering erstellt. Dafür wurden die Bibliotheken GLFW , glbinding und glm verwendet. Um das Datenaufkommen bei der Übertragung der Einzelbilder zu senken wurden diese mittels TurboJPEG zu JPG Bildern komprimiert.

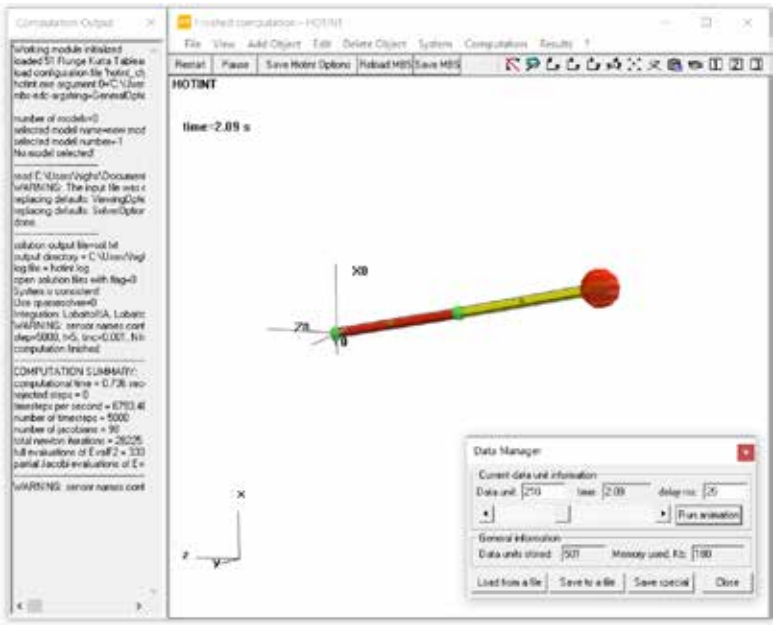
Das **Backend-For-Frontend** (BFF) wurde mit Node.js von Grund auf neu entwickelt. Für die Kommunikation mit dem Backend / HOTINT kam gRPC-JS zum Einsatz. Als GraphQL-Framework wurde der Apollo Server verwendet, wobei zur Steuerung der Simulation GraphQL Mutations und zur Übertragung der Bilder / Nachrichten GraphQL Subscriptions herangezogen werden.

Für das **Frontend** wurde Angular in Kombination mit Material Design verwendet. Für die Interaktion mit der 3D-Ansicht kommt auch bei der Bildübertragung vtk.js zum Einsatz – mit dem Vorteil, dass Kamerainteraktionen nicht manuell neu-implementiert werden mussten.

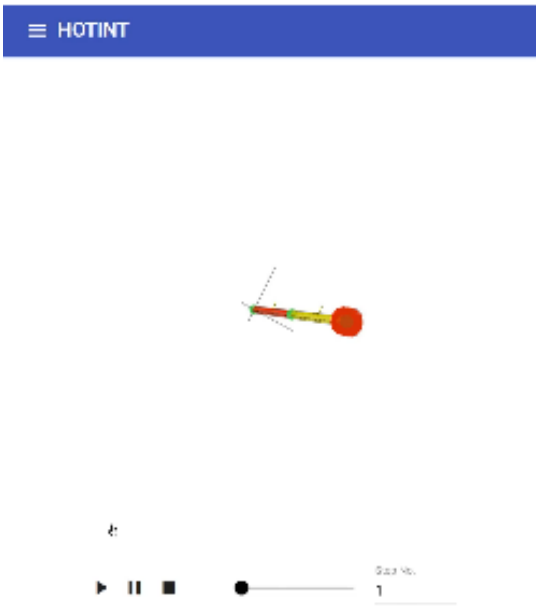
Die Interaktion mit der Simulation entspricht grob folgendem Ablauf:

- Vorbereiten der Simulationsumgebung (Auswahl der Projekt-Datei).
- Starten des Simulationslaufs.
- Während des Simulationslaufs generiert HOTINT laufend neue Zeitschritte und Nachrichten. Mithilfe des Backends und dem BFF werden diese zum Frontend kommuniziert.
- Das Frontend entscheidet, wann neue Bilder benötigt werden oder das Picking (Selektion von Elementen) gestartet werden soll. Es kann hier auch jederzeit auf vorherige Zeitschritte zurückgegriffen werden.
- Fertigstellung oder Stoppen der Simulation.

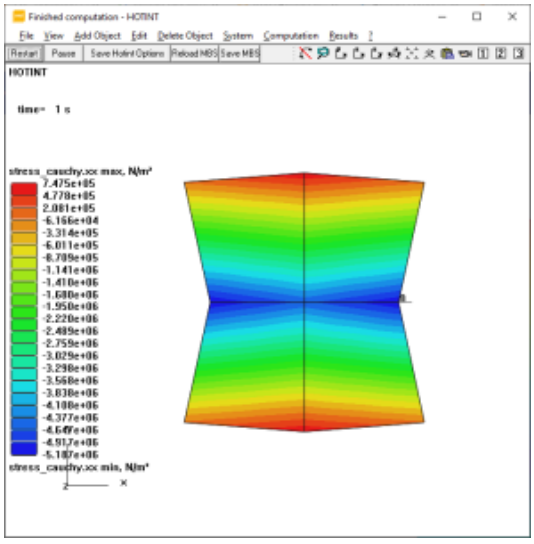
In nachfolgender Darstellung erfolgt die **Gegenüberstellung** der bisherigen Desktop-Anwendung und der neuen Web-Anwendung:



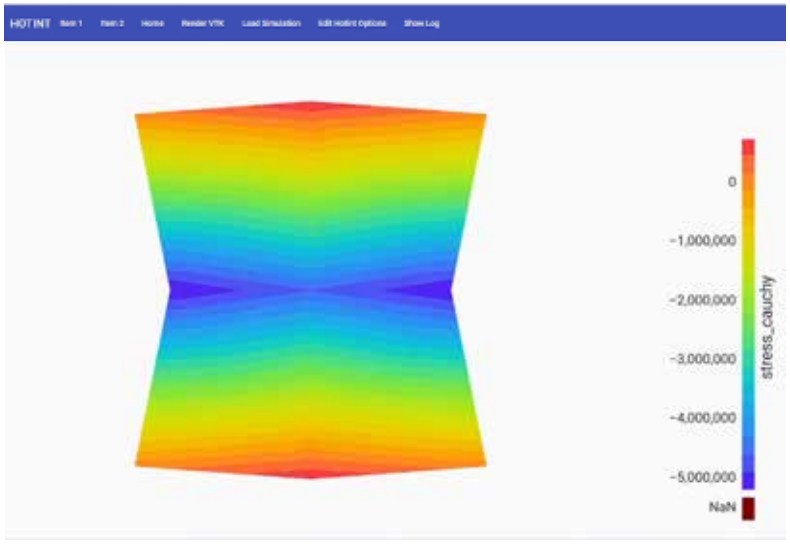
Screenshot 1: alte Desktopanwendung



Screenshot 2: neue Webversion



Screenshot 1: alte Desktopanwendung



Screenshot 2: neue Webversion

Autoren

Martin Hochstrasser, MSc
Software Developer

DI (FH) Josef Jank, MSc
Senior Software Architect & Project Manager

DI (FH) Alexander Leutgeb
Head of Unit Industrial Software Applications

| Fazit

Wie eingangs erwähnt, wird ein Großteil der Benutzer*innenoberflächen heute mit Web-Technologien realisiert, weil diese für einen breiten Anwender*innenkreis einen niederschwelligen Zugang ermöglichen. Mit diesem Erwartungsdruck sind daher nun auch klassische Desktop-Anwendungen konfrontiert. In diesem Zusammenhang stellt sich die Frage einer entsprechenden Vorgehensweise: vollständige Neuentwicklung oder schrittweise Modernisierung (Reengineering). Die Entscheidung muss natürlich je Projekt individuell getroffen werden und hängt natürlich stark vom Umfang und der Qualität der vorhandenen Code-Basis ab. Es ist aber nicht so, dass eine Neuentwicklung generell die bessere Lösung ist. Im vorgestellten Projekt haben sich der gewählte minimalinvasive Reengineering-Ansatz und die agile Vorgehensweise bewährt. Die Erwartungen im Hinblick auf die Ergebnisse und Aufwände konnten damit sogar übertroffen werden, da das Projekt innerhalb einer Durchlaufzeit von rund drei Monaten und einem Aufwand von ca. 500 Personenstunden abgeschlossen werden konnte.

In Summe waren für den Erfolg zusammenfassend folgende Faktoren ausschlaggebend:

- Die **agile** Vorgehensweise und die regelmäßige gemeinsame Abstimmung der Zwischenergebnisse und der weiteren Vorgangsweise.
- Der gewählte **minimalinvasive** Lösungsansatz.
- **Zukunftssicherheit**, weil bestehender Code und Erweiterungen unabhängig voneinander gewartet und weiterentwickelt werden können. ◆

(R)Evolution der Sprachmodelle – ChatGPT

ANTWORTEN AUF DIE WICHTIGSTEN FRAGEN ZU DEN KÜNSTLICHEN INTELLIGENZEN CHATGPT, BARD UND CO.

von Sandra Wartner, MSc

In den letzten Wochen drehte sich alles um die neuen Künstlichen Intelligenzen (KIs) der großen Player. Seit der Veröffentlichung von OpenAIs Chatbot-Revolution ChatGPT Ende November 2022 (vgl. Seite von OpenAI) befindet sich das Sprachmodell in einer für jeden zugänglichen Forschungs- bzw. Feedbackphase und hat damit das öffentliche Interesse geweckt. Mit bereits über 100 Mio. User*innen im Jänner zeigten sich die vielseitigen Funktionen und Anwendungsszenarien, welche die KI zu bieten hat. Sie hilft bei alltäglichen und beruflichen Schreibaufgaben wie Einkaufslisten oder kreativen Marketingtexten, liefert Vorschläge zur Planung von Geburtstagspartys, schreibt Gedichte und Songtexte oder hilft bei Programmieraufgaben.

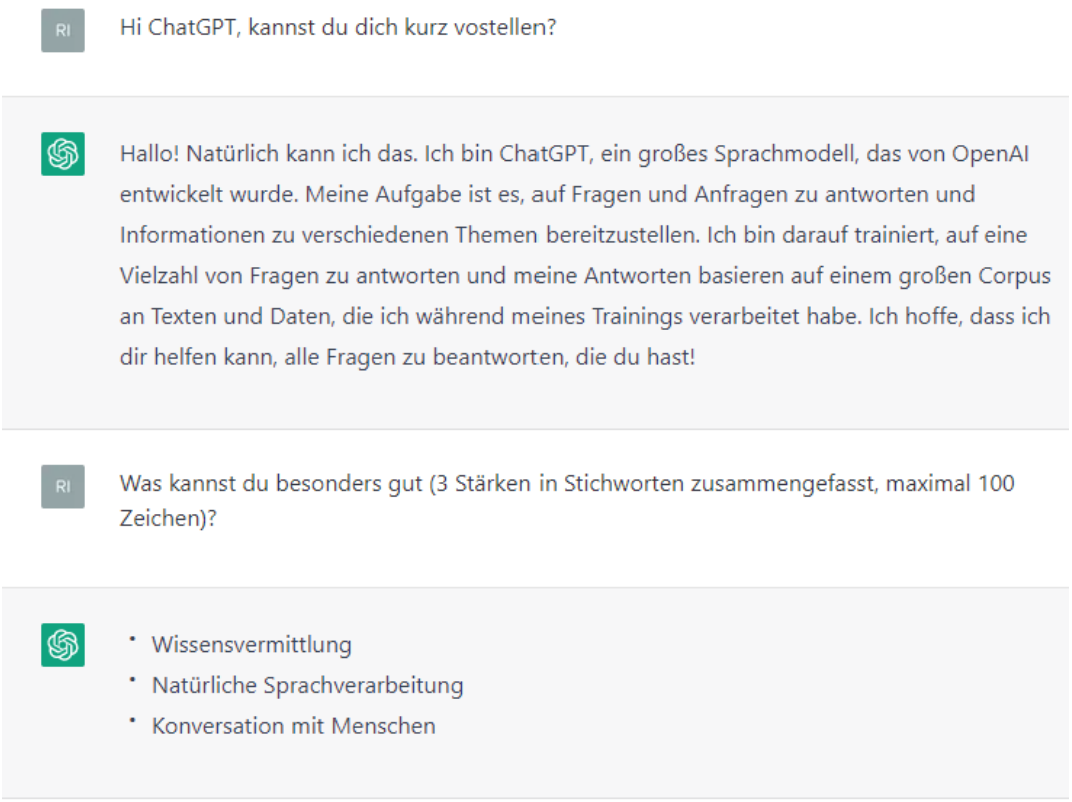
Als Antwort auf die Veröffentlichung von ChatGPT kündigte Anfang Februar auch Google die eigene Lösung Bard an. Auch China will als weiterer Konkurrent mitmischen und plant mit Ernie von Baidu im März ein Ende der Testphase. Wie sich die Sprachmodelle in unseren Alltag einfügen und uns unterstützen werden, ist noch nicht ganz klar. Dass ein Umdenken erforderlich ist und sich auch die Berufswelt dementsprechend anpassen wird, lässt sich ebenfalls nicht abstreiten.

Doch was steckt eigentlich hinter der Technologie rund um die neuen Künstlichen Intelligenzen?

Ein weiterer Meilenstein für die KI-Landschaft

Die dahinterliegende Technologie ist Natural Language Processing, kurz NLP **(vgl. einen unserer Fachbeiträge zu NLP)** und ermöglicht es Maschinen, menschliche Sprache zu verarbeiten und das Wissen in Form von (Sprach-)Modellen abzubilden. Einen wesentlichen Meilenstein dazu lieferte die Veröffentlichung der Transformer-Architektur (siehe auch [1], **bzw. einen weiteren Fachbeitrag**) im Jahr 2017, vorangetrieben durch die Fortschritte im Bereich des Deep Learning, der Verfügbarkeit großer Mengen an Trainingsdaten sowie der gestiegenen Rechenleistung. Um Sprachrepräsentationen zu lernen, werden dem Sprachmodell in der Pre-Trainingsphase riesige Mengen an Textdaten zur Verfügung gestellt. Zu diesem Zeitpunkt hat das Modell noch keine Informationen zu konkreten Aufgaben wie bspw. Texte übersetzen oder Informationen auswerten, und muss diese in der nachfolgenden Finetuning-Phase mit Hilfe eines annotierten Datensets erlernen.

Das Nachahmen natürlicher Konversationen in Form von konversationeller KI (conversational AI) konnten wir über virtuelle Assistent*innen wie bspw. Siri und Alexa bereits in den letzten Jahren miterleben, die neue Generation schafft es allerdings noch besser, menschliche Interaktionen zu simulieren. Wer in den letzten Jahren die Medien aufmerksam mitverfolgt hat, wird bereits immer wieder über News zu den ChatGPT-Vorgängermodellen GPT-2 (2019) und



GPT-3 (2020) gestolpert sein. GPT ist ein Akronym für Generative Pretrained Transformer, und nutzt – wie der Name bereits verrät – die oben genannte Transformer-Architektur, um basierend auf einer textuellen Konversationsaufforderung einen zum Kontext und Thema passenden, menschenähnlichen Text als Antwort zu generieren. Die Modelle wurden von OpenAI, einem US-amerikanischen Unternehmen zur Erforschung von Künstlicher Intelligenz, entwickelt und zählen zu den bisher größten neuronalen Netzwerken.

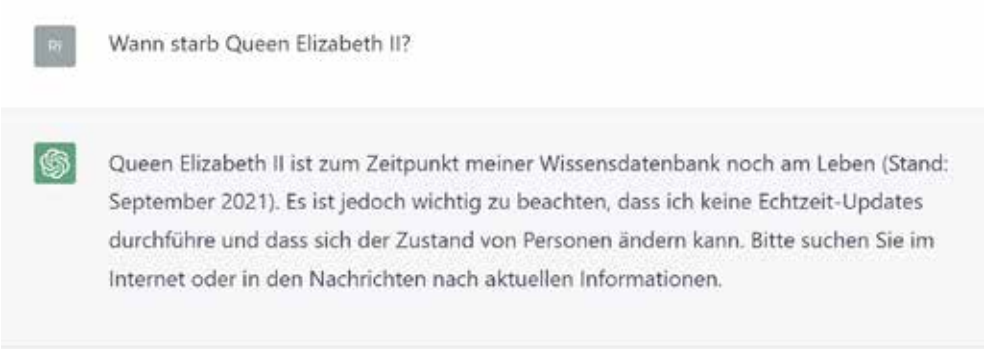
Chat-GPT gehört als finegetunte Version von GPT-3 zur Familie GPT-3.5 und mit 175 Milliarden Parameter zu einem der leistungsstärksten State-of-the-Art Sprachmodelle. Trainiert wurde dieses auf einer riesigen Menge unterschiedlicher Textdaten aus Websites, Artikeln und Büchern, um Aufgaben zu erledigen wie bspw. Texte verfassen, analysieren und zusammenfassen bzw. umschreiben, aber auch zum Schreiben und Analysieren von Programmcode in unterschiedlichen Programmiersprachen. Am besten funktioniert ChatGPT in Englisch, kann aber auch andere Sprachen verarbeiten. Microsoft hat als einer der wichtigsten Geldgeber für die Integration der Technologie in der Suchmaschine Bing eine weitere Milliardeninvestition getätigt **(siehe auch Blogbeitrag von Microsoft)**, und es kann gut sein, dass manche Funktionen zukünftig auch in Outlook oder Word anzutreffen sein werden. Die Nutzung des Bing-Chatbots wurde zu Beginn der Startphase erstmals eingeschränkt, nachdem das Modell unangemessenes Verhalten (u.a. Liebeserklärungen, Drohungen) gezeigt hat.

Auch Google hat inzwischen seinen eigenen Chatbot Bard angekündigt **(siehe auch Blogbeitrag von Google)**. Dieser basiert auf dem Google Sprachmodell LaMDA (Language Model for Dialogue Applications), welches hauptsächlich auf Texten und Dialogen trainiert wurde. Einer der größten Unterschiede liegt aktuell darin, dass Bard – im Gegensatz zu ChatGPT's Wissenslimit bis zum Jahr 2021 – Informationen aus dem Web bezieht. Bard befindet sich aktuell ebenfalls noch in der Testphase, jedoch nur für eine ausgewählte Testgruppe.

Wie kann ich ChatGPT selbst ausprobieren?

Um das Können von ChatGPT selbst zu testen, wird ein OpenAI-Account benötigt **[https://chat.openai.com]**. Über ein Dialogfenster im Browser entsteht eine Art Chat mit dem Modell, wobei man das Verfassen der Antworten in Echtzeit mitverfolgen kann.

Wer neugierig ist, muss sich eventuell etwas gedulden, da es aufgrund der großen Nachfrage immer wieder zu Stabilitäts- und Kapazitätsproblemen kommt. Die Nutzung von ChatGPT ist in der aktuellen Entwicklungsphase kostenlos, es gibt allerdings schon erste Kostenmodelle wie ChatGPT Plus (Link zu ChatGPT Plus), die auch zu Spitzenzeiten einen stabilen Zugriff auf das Modell ermöglichen sollen. Da die eingegebene Daten u.a. zur weiteren Verbesserung bzw. Evaluierung des Modells ausgewertet werden, ist es wichtig, keine sensiblen und persönlichen Daten einzugeben. ▸



Wie zuverlässig sind die Antworten der Modelle und wo liegen ihre Grenzen?

ChatGPT liefert auf den ersten Blick eine beeindruckende Performance. Die Konversation wirkt so (erschreckend) natürlich, als würde man mit einer menschlichen Person chatten. Die Fähigkeiten haben aber auch gewisse Grenzen [siehe Beschreibung von OpenAI: https://openai.com/blog/chatgpt].

Kein übereilter Vertrauensvorschuss

Auch wenn die Modellantworten sehr plausibel klingen, sind diese nicht immer korrekt. ChatGPT neigt dazu, Unwissen zu verschleiern, indem es immer eine Antwort generiert, und rät bei mehrdeutigen Anfragen welche Absicht dahintersteckt. Teilweise frei von ChatGPT erfundene Fakten oder Quellen (Fake News) und die Urheberschaft der Antworten lassen sich äußerst schwer überprüfen. Daher ist es immer notwendig, die Glaubwürdigkeit der zurückgelieferten Informationen kritisch zu hinterfragen und z.B. mit zusätzlichen Recherchen auf Korrektheit zu überprüfen.

Nichts dem Zufall überlassen?

Auch die Formulierung der Eingabe hat einen wesentlichen Einfluss auf die Antwort, die das Modell gibt. Es kann also passieren, dass ChatGPT bei einer bestimmten Frage die Antwort nicht (zufriedenstellend) beantworten kann, bei einer Umformulierung jedoch passend antwortet.

Nicht am Puls der Zeit

Da kein Zugang zu Echtzeitdaten vorliegt und die Trainingsdaten nur bis 2021 reichen, gibt es keine Garantie auf die Aktualität der gelieferten Informationen. (Siehe Abbildung oben)

Verantwortungsvoller Umgang mit KI-Systemen

Da es bei den notwendigen Massen an Trainingsdaten nicht möglich ist, jeden einzelnen Text zu überprüfen, kann ein Sprachmodell u.a. gesellschaftliche oder historische Vorurteile und Stereotypen mitlernen, reproduzieren und damit zahlreiche ethische Probleme hervorrufen. Dieses Problem zeigte sich bereits 2016 bei Tay, einem Chatbot von Microsoft, welcher nach weniger als 24h wieder offline genommen werden musste, da sich dieser in einen rassistischen und sexistischen Chatbot verwandelt hatte (siehe auch Blogbeitrag von The Verge). Das vorsorgliche Treffen von Sicherheitsmaßnahmen beim Einsatz (und soweit möglich auch beim Training) dieser Modelle ist daher unbedingt erforderlich. Auch bei ChatGPT wurden Mechanismen eingebaut, um unangemessene Anfragen in der Regel nicht zu beantworten.

Wer KI-Modelle verwendet, sollte auch ein Verständnis dafür haben, wie das Modell zu dieser Antwort gekommen ist, um es verantwortungsvoll einsetzen zu können (siehe auch Fachbeitrag zu Explainable AI und Vertrauen in KI). OpenAI ist sich dieser Thematik durchaus bewusst und versucht, mehr Transparenz hinsichtlich ihrer Absichten und Fortschritte zu schaffen sowie den Finetuning-Prozess verständlicher und kontrollierbarer zu gestalten [vgl. Blogbeitrag von OpenAI].

Welches Potenzial für Geschäftsanwendungen verbirgt sich hinter den Technologien?

Die aktuellsten Errungenschaften in der Weiterentwicklung der NLP-Technologien öffnen neue Türen und prägen die Zukunft von Künstlicher Intelligenz und ihrer Anwendungen. Um die Technologien verantwortungsvoll und nutzbringend einsetzen zu können, müssen Anpassungen für neue Bereiche und die individuellen Bedürfnisse der einzelnen Branchen vorgenommen werden. Wir, die RISC Software GmbH, beschäftigen uns konkret mit dem nachhaltigen Einsatz von NLP-Technologien in praktischen Anwendungen (siehe auch Fachbeitrag zu Natural Language Understanding) und schaffen individuelle Lösungen für unsere Kund*innen. Die entwickelten KI-Systeme unterstützen und ergänzen dabei die Fähigkeiten der Domänen-Expert*innen bei der Ausführung ihrer anspruchsvollen Tätigkeiten. Transformer-Modelle sind auch in unseren Projekten seit einigen Jahren fester Bestandteil und häufig Teil einer erfolgreichen Problemlösung in den NLP-Projekten. Mit diesen positiven Erfahrungen möchten wir den nächsten Technologie-Schritt mitgehen und die Anwendungsmöglichkeiten und Grenzen der neuen Technologien ausloten.

In den letzten Jahren sind wir immer wieder auf ähnliche Herausforderungen gestoßen. Häufig sind keine oder nicht ausreichend Trainingsdaten vorhanden, die mit hohem personellen Aufwand manuell erstellt und annotiert werden müssen, um Modelle speziell auf eine Aufgabenstellung trainieren zu können (wenn nicht glücklicherweise öffentliche Datensätze vorhanden sind, welche

die Datenlage zufällig ausreichend gut abbilden). Zwar wird dieser Schritt kaum jemals völlig automatisiert durchgeführt werden können, allerdings könnten die neuen Modelle zum Vorannotieren genutzt werden und durch Menschen als Supervisor in einem weit weniger aufwändigen Prozess korrigiert und ergänzt oder für ein schnelles Prototyping genutzt werden. Auch in NLP-Tasks wie bspw. Informationsextraktion könnten Modelle wie ChatGPT (mit einem gewissen Postprocessing- und Integrations-Aufwand) als weitere Komponente eines Modell-Ensembles genutzt werden.

Sprachmodelle wie ChatGPT sind bislang noch nicht ausgereift, und dessen Einsatz muss u.a. hinsichtlich Datenschutz, Kosten-/Nutzen-Faktor, API-Abhängigkeiten sowie Erklärbarkeit und Transparenz gut überdacht und für den individuellen Anwendungsfall entschieden werden.

Es gibt derzeit viele Organisationen und Unternehmen, die mit konkreten Ideen zur Prozessverbesserung oder weiteren Optimierung ihrer Produkte durch NLP-Assistenten an uns herantreten. Die RISC Software GmbH unterstützt und begleitet ihre Kund*innen dabei gerne von der Idee bis hin zur Integration [vgl. mehr zu unseren Kompetenzen im Bereich NLP]. ◆

Referenzen

[1] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

Autorin

Sandra Wartner, MSc
Data Scientist

Stabiles Stromnetz durch Planung und Optimierung

METAHEURISTIKEN FÜR DIE KURZFRISTIGE OPERATIVE BETRIEBSPLANUNG DER STROMVERSORGUNGSSYSTEME

von Ionela Knospe, PhD

Die Elektrifizierung wird oft als die technische Errungenschaft angesehen, die unser Leben am meisten verändert hat. Für die jüngeren Generationen in vielen Teilen der Welt ist die Nutzung elektrischer Energie eine Selbstverständlichkeit und sie können sich ihr Leben ohne sie nicht vorstellen. Aber welche Kraftwerke müssen im Laufe eines Tages eingesetzt werden, sodass zu jedem Zeitpunkt die Last in einem Stromnetz gedeckt ist und dessen Gesamtbetriebskosten minimal sind? Wie wird in der operativen Betriebsplanung die Ungenauigkeit der Prognose für erneuerbare Energiequellen und den Energieverbrauch berücksichtigt? Wie soll bestmöglich der Lade- und Entlade-Zyklus der Energiespeichersysteme im Stromnetz mit dem aktuellen Marktpreis für den Strom koordiniert werden? Diese Fragen sind ein Teil der täglichen operativen Betriebsplanung eines Stromversorgungssystems.

Der Trend zu sauberen und erneuerbaren Energiequellen, wie Wind- und Solarenergie, erfordert von den Stromversorgungssystemen eine verstärkte Integration von Energiespeichersystemen und flexiblen Lasten. Die langfristige Betriebsplanung der Stromversorgungssysteme befasst sich mit dem Ausbau der Übertragungskapazitäten und der Bereitstellung zusätzlicher Kapazitäten. Sie erforscht intensiv die großen Transformationen, die das Energiesystem durchlaufen muss, um bis 2050 null Netto-CO2-Emissionen zu erreichen. Die kurzfristige Betriebsplanung von Stromsystemen hat einen Planungshorizont von wenigen Tagen bis Stunden und befasst sich mit Formulierungen des **Unit Commitment Problems (UCP)**, des **Optimal Power Flow (OPF)** und der **Day-Ahead- und Intraday-Märkte**. Bei dieser Planung treffen Data Science, Optimierung und Simulation auf wunderbare Weise zusammen, um einen optimalen operativen Betriebsplan zu erstellen. Die Methoden, die in der Optimierung verwendet werden, sind sowohl exakte Methoden als auch Metaheuristiken und hybride Verfahren. Metaheuristiken sind Algorithmen zur näherungsweisen Lösung von Optimierungsproblemen. Sie haben sich als gut geeignet erwiesen, um komplexe und schwer lösbare konkrete Problemstellungen zu lösen.

Kurzfristige Betriebsplanung der Stromversorgungssysteme

UCP- und OPF-Problemstellungen

Das Unit Commitment Problem (UCP) und das Optimal Power Flow (OPF) gehören zu den wichtigsten und kritischsten Problemen in der Energiewirtschaft. Das UCP-Problem wird bereits seit den 1940er Jahren ausgiebig erforscht und ist aufgrund der möglichen Betriebskosteneinsparungen immer noch Gegenstand aktiver Forschung. Das OPF-Problem wurde 1962 als eine Erweiterung des Problems der optimalen wirtschaftlichen Einteilung in traditionellen Stromversorgungssystemen eingeführt, indem Gleichungen für den elektrischen Leistungsfluss einbezogen wurden. Es gibt umfangreiche Literatur sowohl über deterministische und stochastische UCP-Formulierungen als auch über OPF-Methoden. Wir weisen hier auf [1] und [3] und die darin enthaltenen Referenzen hin.

Im Folgenden wird ein kurzer Überblick über die Standardformulierungen der beiden Probleme gegeben.

UCP befasst sich mit der optimalen Planung der Stromerzeugung programmierbarer Generatoren im Stromnetz, um die prognostizierte Stromnachfrage zu decken. Die Planung erfolgt unter Berücksichtigung einer Reihe physikalischer und technischer Restriktionen der Erzeuger (minimale/maximale Leistung, maximale Leistungserhöhung und -reduktion, Anfahrkosten und Abschaltkosten). In der Regel ist das Ziel die Minimierung der Gesamtbetriebskosten, aber auch zusätzliche Ziele wie minimale CO2-Emissionen können berücksichtigt werden. Die Lösung des UCP-Problems enthält für jede Erzeugungseinheit und jeden Zeitschritt des Planungshorizonts einen Ein-/Aus-Zustand – d.h. die Entscheidung über den Einsatz der Erzeugungseinheiten und zusätzlich die Leistung. Es gibt verschiedene mathematische Formulierungen des UCP, je nachdem, wie die Übertragungs- und Betriebsrestriktionen der Systeme berücksichtigt werden. Diese Formulierungen führen natürlich zu unterschiedlicher Rechenkomplexität.

OPF befasst sich für einen Zeitschritt im Planungshorizont mit der optimalen Planung der eingesetzten Erzeugungseinheiten, unter Einhaltung zulässiger Leistungsflüsse. Das Modell der AC-Lastflussrechnung (AC-OPF) ist das präziseste für die Modellierung der physikalischen Leistungsflüsse im gesamten Netz, da es Leitungsverluste, Netzparameter sowie Wirk- und Blindleistung berücksichtigt. Es handelt sich hier um ein nichtlineares, nicht-konvexes Problem, das schwer zu lösen ist. Die vereinfachte Form der Lastflussrechnung (DC-OPF) ist eine Annäherung an die AC-OPF, die die Leistungsflüsse in einer linearisierten Form betrachtet, nur die Wirkleistung und keine Leitungsverluste berücksichtigt. Die DC-OPF ist somit konvex und falls die Zielfunktion linear ist, ein lineares Optimierungsproblem, das mit einem kommerziellen oder freien Löser effizient gelöst werden kann.

Mit der zunehmenden Präsenz erneuerbarer Energiequellen in Stromnetzen erhielten auch UCP und OPF eine erhöhte Aufmerksamkeit, um unter anderem die Unsicherheitsfaktoren, welche durch die intermittierende Natur erneuerbarer Energien entstehen, zu erfassen.

Lösungsansätze für die kurzfristige Betriebsplanung der Stromversorgungssysteme

Wenn das Stromnetz mit dem vollständigen Wechselstrommodell in UCP dargestellt wird, dann beinhaltet UCP bereits die OPF-Problemstellung. Wie bereits erwähnt, ist sie selbst nicht konvex und nicht linear, und daher sehr schwierig zu lösen. Darüber hinaus befassen sich typische reale Anwendungen von UCP mit großen Stromsystemen – in Bezug auf die Anzahl der Erzeuger und die Größe des Übertragungsnetzes -, was die Komplexität der UCP-Problemstellung weiter erhöht. Für die kurzfristige Betriebsplanung von Stromversorgungssystemen wird daher UCP häufig ohne die physikalischen Restriktionen der Übertragungsleitungen oder nur in linearisierter Form gelöst. OPF wird dann in diesem Fall in einem zweiten Schritt gelöst.

Formulierungen von UCP als gemischt-ganzzahlige lineare Programmierung (MILP) haben im Laufe der Zeit erheblich an Bedeutung gewonnen. Der Grund dafür sind die dramatischen Verbesserungen der Leistung kommerzieller und kostenloser MILP-Löser. Diese Formulierungen haben den Vorteil, dass sie optimale Lösungen generieren, wenn diese gefunden werden. Eine Schwäche, trotz der erreichten Fortschritte, liegt bei großen Probleminstanzen noch in der benötigten Rechenzeit.

Neben exakten Methoden werden auch Metaheuristiken und hybride und iterative Lösungsverfahren für die Lösung großer Optimierungsprobleme im Bereich der Strom- und Energiesysteme eingesetzt. Genetische Algorithmen, Particle Swarm Optimization, evolutionäre Algorithmen, aber auch Tabu-Search und Simulated Annealing (siehe Wikipedia-Artikel zu Metaheuristik: https://en.wikipedia.org/wiki/Metaheuristic) sind die am häufigsten verwendeten Metaheuristiken in diesem Bereich. Um die Leistung des Algorithmus zu verbessern, müssen sie oft um problemspezifische Operatoren ergänzt werden. In den letzten Jahren hat das Forschungsinteresse an der Integration von maschinellem Lernen in Metaheuristiken stark zugenommen. Diese Kombination wird in verschiedenen Bereichen untersucht, wie Algorithmenauswahl, Initialisierung, Evolution oder Parametereinstellung. Daher ist es von Interesse, ob die Kombination von maschinellem Lernen mit Metaheuristiken auch zu neuen Lösungsansätzen für die kurzfristige Betriebsplanung von Stromversorgungssystemen führen könnte.

Tabu Suche für die kurzfristige Betriebsplanung der Stromversorgungssysteme

Tabu-Suche ist ein iteratives metaheuristisches Verfahren zur Lösung oder Annäherung kombinatorischer Optimierungsprobleme, die lokale Suchmethoden verwendet. Diese lokalen Suchmethoden untersuchen den Lösungsraum, indem sie von einer möglichen Lösung zu einer verbesserten Lösung in ihrer Nachbarschaft wechseln. Die Suche endet, wenn ein Stoppkriterium erfüllt ist (z.B. ein Versuchslimit oder ein Schwellenwert der Bewertung). In den meisten Fällen aber ohne Garantie der Lösungsqualität. Lokale Suchmethoden neigen dazu, in suboptimalen Regionen oder Plateaus hängenzubleiben, wo keine verbesserten Nachbarn verfügbar sind. Um dies zu vermeiden, verwendet die Tabu-Suche eine Speicherstruktur, die den Algorithmus dazu zwingt, neue Bereiche des Suchraums zu untersuchen. Die bereits besuchten Lösungen werden für eine bestimmte Dauer dorthin gespeichert, d.h. als verboten oder „tabu" markiert (vgl. dazu Wikipedia-Artikel zu Tabu-Suche: https://en.wikipedia.org/wiki/Tabu_search).

In einem aktuellen Forschungsprojekt hat die RISC Software GmbH die Anwendung der Tabu-Suche mit einem adaptivem Nachbarschaftsoperator-Selektor für die kurzfristige Betriebsplanung von Stromversorgungssystemen untersucht und für vielversprechend befunden ([4]). Der betrachtete Planungshorizont umfasst einen Tag mit einer Zeitauflösung von 15 Minuten – beide Zeitrahmen sind skalierbar. Stromsysteme mit den folgenden Komponenten und technischen Details wurden analysiert: ›

- **Programmierbare Erzeuger:** minimale und maximale Wirkleistung, Mindestzeitspannen für Erzeugungsphasen und Ruhephasen, maximale Leistungserhöhung und -reduktion, maximale Reservegrenze, Erzeugungskosten, Anfahrkosten, Abschaltkosten und Reservekosten;
- **Energiespeichersysteme:** maximale Kapazität, minimale und maximale Lade- und Entladeleistung, minimale Lade- und Entladezeit, Lade- und Entladeeffizienzfaktoren, Ladezustand;
- **Photovoltaik:** installierte Kapazität pro Einheit;
- **Lasten:** Stromverbrauch;
- **Übertragungsleitungen:** Blindwiderstand, maximaler Leistungsfluss und Übersetzungsverhältnis, wenn die Übertragungsleitung ein Transformator ist; sie werden durch DC-Leistungsflüsse modelliert.

Zusätzlich zu diesen Komponenten sind im Optimierungsmodell zwei Parameter enthalten, Seamless Index und Reservefaktor, welche die Autarkie des Systems ansprechen und steuern (siehe auch [2]).

Die Last und die Photovoltaik-Erzeugung werden durch Prognosemodelle bereitgestellt, deren Datenbasis aus realen Datensätzen und historischen Daten besteht. Die Prognosemodelle basieren auf nicht linearer Regression wie Gradient Boosting unter Verwendung modernster Wolkenmodelle und zusätzlicher Funktionen von Wettervorhersageanbietern.

Energiespeicherarbitrage ist eine Technik, bei der Strom gekauft und gespeichert wird, wenn die Netzstrompreise am günstigsten sind und in den Spitzenzeiten genutzt wird, wenn die Netzstrompreise am höchsten sind. Für alle Energiespeichersysteme im Stromnetz ist die verwendete Planungsstrategie Arbitrage zu erreichen.

Der vorgeschlagene Lösungsansatz für die Day-Ahead-Betriebsplanung von Stromsystemen mit den oben definierten Komponenten basiert auf der Tabu-Suche mit adaptivem Nachbarschaftsoperator-Selektor. Es handelt sich um einen hierarchischen Ansatz, bei dem die Erzeugung einer Lösung drei Schritte umfasst:

- **Zuordnung des Ein-/Aus-Zustandes für die programmierbaren Erzeuger und Energiespeicher über den gesamten Planungshorizont;**
- **Einplanung der Energiespeichersysteme über den gesamten Planungshorizont;**
- **Lösen von DC-OPF für jeden Zeitschritt des Planungshorizonts.**

Die Teilprobleme im zweiten und dritten Schritt sind lineare Programme (LP), die zum Beispiel mit einem open-source LP-Löser und der Optimierungssuite OR-Tools von Google gelöst werden können. Die Tabu-Suche verwendet Nachbarschaftsoperatoren, die auf die aktuelle Problemformulierung zugeschnitten sind. Den Nachbarschaftsoperatoren werden Punkte basierend auf ihrer Leistung zugewiesen, die durch die Verringerung der Zielfunktion gemessen wird. Diese Punktezuweisung hat ein Fading Memory, d.h. je länger ein Erfolg zurückliegt, desto weniger Punkte liefert er für die aktuelle Bewertung. Zusammenfassend kann dieser Ansatz als ein hybrider Ansatz angesehen werden, der eine Kombination aus score-basierter, durchschnittlicher und extremwertbasierter Punktezuweisung darstellt.

Dieser Lösungsansatz für die Day-Ahead-Betriebsplanung von Stromnetzen liefert akzeptable Lösungen in angemessener Rechenzeit. Obwohl nicht unbedingt optimal, stellen sie für große Stromnetze eine wertvolle Alternative zur äquivalenten Formulierung als gemischtes ganzzahliges lineares Programm (MILP) dar,

insbesondere wenn kostenlose oder open-source MILP-Löser verwendet werden. Die Komponenten und technischen Details des Stromnetzes können bei diesem Ansatz leicht erweitert werden, z.B. mit anderen erneuerbaren Energiequellen wie Windparks oder mit flexiblen Lasten. Die Verwendung von metaheuristischen Lösungen für die Planung könnte daher potenziell verschiedene Anwendungen finden, etwa bei der Planung von Mikronetzen oder Energiegemeinschaften.

Referenzen

[1] Abdou, I., Tkiouat, M.: Unit Commitment Problem in Electrical Power System: A Literature Review, International Journal of Electrical and Computer Engineering (IJECE) 8 (3), 1357-1372 (2018).

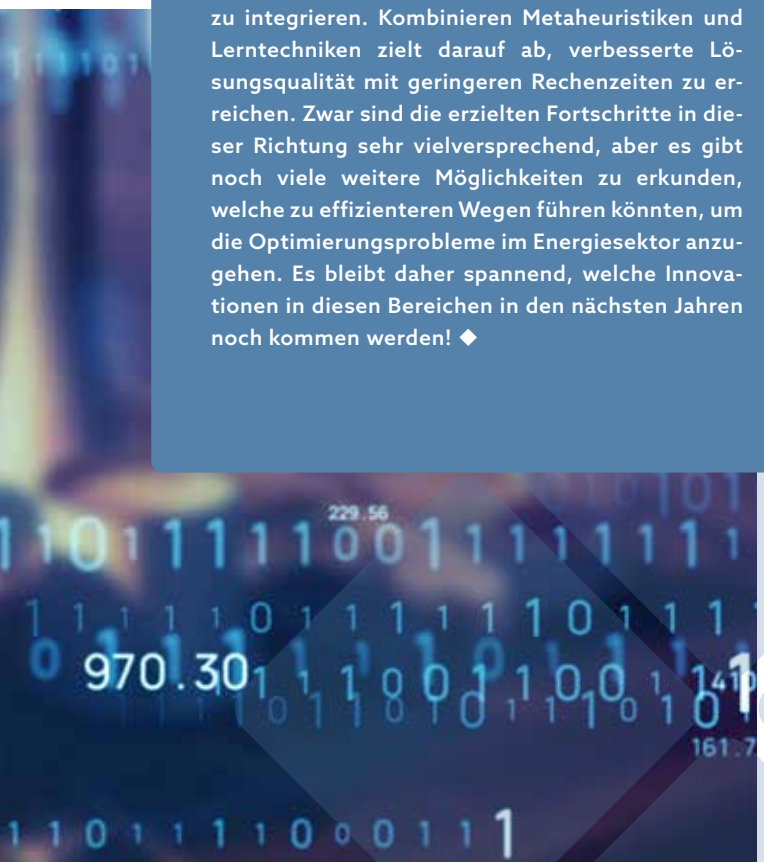
[2] Hosseinnezhad, V., Rafiee, M., Ahmadian, M., Siano, P.: Optimal day-ahead operational planning of microgrids. Energy Conv. Manag. 126, 142-157 (2016).

[3] Khan, B., Singh, P.: Optimal Power Flow Techniques under Characterization of Conventional and Renewable Energy Sources: A Comprehensive Analysis, Journal of Engineering (2017).

[4] Knospe, I., Stainko, R., Gattinger, A., Bögl, M., Rafetseder, K., Falkner, D.: A Tabu Search Approach to the Short-Term Operational Planning of Power Systems, Proceedings International Conference on Operations Research 2022, (akzeptiert).

| Fazit

Der Energiesektor steht derzeit vor großen Umwälzungen, die sich sowohl in der langfristigen als auch in der kurzfristigen Planung von Energiesystemen widerspiegeln. Für die Methoden, die in der Optimierungsprobleme in diesen Bereichen eingesetzt werden, wird derzeit an der Möglichkeit geforscht, Techniken des maschinellen Lernens zu integrieren. Kombinieren Metaheuristiken und Lerntechniken zielt darauf ab, verbesserte Lösungsqualität mit geringeren Rechenzeiten zu erreichen. Zwar sind die erzielten Fortschritte in dieser Richtung sehr vielversprechend, aber es gibt noch viele weitere Möglichkeiten zu erkunden, welche zu effizienteren Wegen führen könnten, um die Optimierungsprobleme im Energiesektor anzugehen. Es bleibt daher spannend, welche Innovationen in diesen Bereichen in den nächsten Jahren noch kommen werden! ◆



Autorin

Ionela Knospe, PhD
Mathematical Optimization Engineer

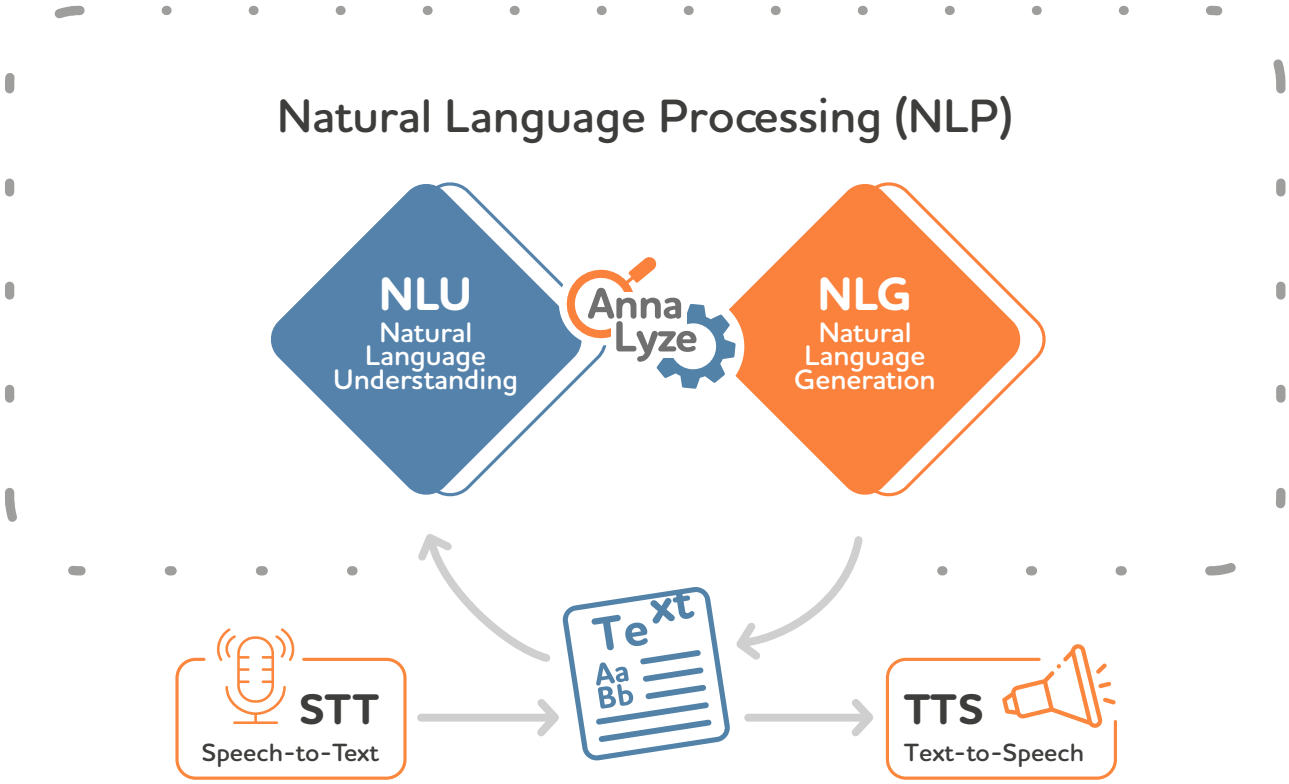
Mit Natural Language Understanding (NLU) vom Textchaos zum Wissensgewinn

WIE NATURAL LANGUAGE UNDERSTANDING AUCH IHREM UNTERNEHMEN HILFT, BESTEHENDE PROZESSE ZU OPTIMIEREN

von Sandra Wartner, MSc

In vielen Unternehmen findet zunehmend eine Verlagerung in Richtung Digitalisierung und Automatisierung statt. Dabei fallen kontinuierlich enorme Mengen an unstrukturierten Daten an, deren Umfang und Komplexität die betroffenen Stakeholder vor einer Auswertung abschrecken lassen, oder das Potenzial in den vorhandenen Daten häufig erst gar nicht erkannt wird. Egal ob Störmeldungen in Fertigungsprozessen analysiert, Arztbriefe strukturiert abgelegt oder Produkte automatisiert vorgeschlagen werden sollen, Natural Language Understanding (NLU) bietet ein breites Spektrum an branchenspezifischen und -übergreifenden Einsatzmöglichkeiten.

Sprache ist omnipräsent und begegnet uns sowohl in unserem Alltag als auch in unserem beruflichen Umfeld in vielen unterschiedlichen Facetten – von Menschen geschrieben, gesprochen und in unterschiedlichen Sprachen kommuniziert, aber auch analysiert, bearbeitet und synthetisiert durch Maschinen. Mit Natural Language Processing (NLP) sind Computer in der Lage, natürliche Sprache automatisiert zu verarbeiten, zu erzeugen und als Schnittstelle zwischen Mensch und Maschine zu fungieren (für mehr Details zum Thema NLP siehe [1]). Als Anwendungsbereich der Künstlichen Intelligenz (KI) kommt NLP immer dann zum Einsatz, wenn monotone Prozesse bzw. häufig wiederkehrende Aufgaben in der Textverarbeitung automatisiert, anschließend optimiert und in ein übergeordnetes Framework eingegliedert werden sollen. Dadurch können in verschiedenen Bereichen Fehler minimiert, Prozesse (teil)automatisiert und Einsparungen (durch verringerten Personalaufwand) erzielt werden. Die RISC Software GmbH unterstützt ihre Kund*innen mit ihrer langjährigen, praktischen Erfahrung, wenn es um die Entwicklung



von individuell zugeschnittenen, KI-gestützten Lösungen geht, u.a. auch im Bereich Natural Language Understanding (NLU), einem Teilbereich des Natural Language Processing.

Natural Language Understanding (NLU) konzentriert sich auf die Extraktion von Informationen aus geschriebenem Text und damit auf das Erwerben von Textverständnis hinsichtlich eines bestimmten Teilaspekts. Dabei spielen v.a. Syntax (grammatikalische Struktur) und Semantik (Bedeutung von Wörtern) eine wesentliche Rolle. Beispiele hierfür sind:

- **Informationsextraktion**, z.B. das Erkennen von Personen, Orten oder anderen Schlüsselwörtern in Texten (z.B. Named Entity Recognition (NER)),
  - **Use-Case "Newsadoo"**: „Newsadoo – Alle News zu deinen Interessen" – ermöglicht Benutzer*innen den Zugriff auf Newsartikel zahlreicher Quellen und bietet relevante sowie nach Interessen personalisierte Nachrichten. Im Hintergrund findet mittels NLP eine Transformation von unstrukturierten Textdaten in strukturierte, auswertbare Inhalte statt.
  - **Use-Case "FLOWgoesS2T"**: Sprachnachrichten zum aktuellen Verkehrsgeschehen werden in geschriebene Texte überführt, in denen anschließend mittels NLP wichtige Informationen wie Straßen, Ortsangaben, Fahrtrichtungen und Ereignisse automatisch erkannt und strukturiert gespeichert werden. Dies dient zur Unterstützung der Redakteur*innen bei der Bearbeitung von übermittelten Sprachnachrichten, um verkehrsrelevante Ereignisse rasch identifizieren zu können.
- **Klassifizierung von Text in vordefinierte Kategorien**
  - **Use-Case "ACT4":** In einer Ausbaustufe der bestehenden Plattform-Lösung ACT4 der Compliance 2b GmbH entwickelt die RISC Software GmbH gemeinsam mit dem Unternehmen eine vertrauenswürdige KI-Komponente, welche einerseits Hinweisgebende bei der Abgabe der Meldung unterstützt und andererseits den zuständigen Sachbearbeiter*innen eine effizientere und weniger fehleranfällige Abwicklung der Meldungen ermöglichen soll. Das System soll dabei automatisiert Informationen (z.B. Hinweiskategorie oder Rollen der beteiligten Personen) aus den textuellen Hinweisen ableiten und diese mit bereits strukturell erfassten Daten in Form einer Plausibilitätsprüfung abgleichen.
- **Stimmungs- und Meinungsanalyse (Sentimentanalyse)**
  - **Use-Case "Intelligente Twitter Analyse":** Stehen positive Emotionen in Tweets über aktiennotierte Unternehmen mit deren Aktienkursentwicklung in Zusammenhang? Mittels Sentimentanalyse kann ein Text hinsichtlich Stimmung (positiv, negativ etc.) analysiert und dahingehend evaluiert werden, wie viel Information tatsächlich zwischen den Zeilen steckt.

Am Anfang steht der Datenberg… und was nun?

Die ersten Schritte sind fast immer die schwersten. Nachfolgende (bestimmt nicht vollständige) Checkliste bietet einen Überblick über die relevantesten Fragestellungen, die jedes Projektteam vor der konkreten Planung bzw. Umsetzung von NLU- bzw. KI-Systemen im Allgemeinen klären sollte. ▸

Ist die Problemstellung ausreichend gut formuliert?

- Welchen Anforderungen muss das KI-System genügen, um nutzbringend im operativen Betrieb eingesetzt werden zu können?
- Sind die erwarteten Ergebnisse klar definiert?
- Haben alle Stakeholder die gleichen Erwartungen?

Ist die Art des zu lösenden Problems bekannt bzw. klar abgegrenzt (z.B. Klassifikation von Wörtern oder Dokumenten, Sentimentanalyse)?

- Falls nein, kann das Problem in mehreren Teilproblemen gelöst werden, die sich klar abgrenzen lassen?

Kann ich das Problem anhand der vorhandenen Datenbasis lösen?

- Falls nein, gibt es Möglichkeiten diese Daten zu bekommen, z.B. durch das Verwenden von Daten aus anderen/öffentlichen Quellen, oder durch Sammeln eigener Daten?

Ist die Datenqualität ausreichend „gut"?

- Die Datenqualität ergibt sich aus dem Zusammenspiel unterschiedlicher Kriterien, die abhängig vom Use-Case sind (siehe [2]).
- Falls die Datenqualität nicht ausreichend ist – welche Maßnahmen können gesetzt werden, um diese zu verbessern? Gibt es die Möglichkeit, langfristig eine robuste(re) Datenstrategie im Unternehmen zu etablieren?

Ist eine Ground Truth (korrekt annotierte Beispiele) vorhanden?

- Falls nein, kann diese erstellt werden? Sind Ressourcen verfügbar bzw. ist technisches/domänen-spezifisches Know-how vorhanden, um diese zu annotieren?

Wie bewerte ich, ob eine Lösung „gut genug" funktioniert? Wie kann ich Fehler „messen"?

- Es braucht einerseits Metriken für die Genauigkeit der Modelle selbst, und andererseits Bewertungsstrategien, ob und welcher Mehrwert durch den Einsatz der Lösung erzeugt wird, z.B. eine gewisse prozentuelle Erhöhung einer oder mehrerer KPI's des Unternehmens.

Gibt es bereits Lösungsansätze zu ähnlichen Problemstellungen oder hat das Projekt einen hohen Innovationsgrad? Wie risikotolerant ist meine Organisation?

- Bei hohem Innovationsgrad und vielen Risikofaktoren können auch Fördermöglichkeiten genutzt werden, um das Projekt dennoch, aber mit geringerem Risiko umsetzen zu können (siehe [3]).
- Wenn die Risikofaktoren (noch) unbekannt oder unklar sind, kann eine Machbarkeitsstudie helfen, diese einzuschätzen (siehe [4]).

Wie kann ich ein vertrauenswürdiges KI-System schaffen?

- Welche Bereiche sind für meinen Use-Case relevant, z.B. Nachvollziehbarkeit, Fairness, technische Robustheit (siehe [5])?
- Kann ich Methoden aus dem Bereich Explainable AI nutzen, um meine Black-Box zu durchleuchten (siehe [6])?

Wie bringe ich dem KI-System bei, was es tun soll?

Um von den Rohdaten zu einer erfolgreich umgesetzten NLU-Komponente zu kommen, sind einige Schritte notwendig. Die konkreten Maßnahmen unterscheiden sich zwar von einem Projekt zum nächsten, die grundlegende Vorgehensweise folgt allerdings dem in Abbildung 1 dargestellten Schema.

Datenbasis

Die vorhandenen Rohdaten können in vielen verschiedenen Formaten vorliegen, z.B. als Textfelder in Datenbanken, Inhalte von Webseiten, Textdateien oder Text in Bildern bzw. Scans. Sind Texte in (komplex-)strukturierten PDFs oder Webseiten enthalten, können relevante Inhalte mit einem gewissen Aufwand extrahiert werden. Bei Scans von Dokumenten kommt die Methode Optical Character Recognition (OCR) zum Einsatz, welche Texte in einem zweidimensionalen Bild erkennt und mit deren Position für die weitere Verarbeitung ablegt. Bei Bildern mit strukturierten, maschinengeschriebenen Texten (z.B. Scans oder Fotos von analogen Dokumenten) erzielen OCR-Systeme bereits sehr gute Ergebnisse, bei Fotos (z.B. von Straßenschildern) oder handgeschriebenen Texten stellt dieser Schritt häufig eine Herausforderung dar. Auch Audiodateien können mittels Speech-To-Text-Technologien in geschriebenen Text transkribiert werden. Je nach Qualität der Aufnahme, Sprache und Dialekt kann auch dies erheblichen Aufwand bedeuten, bis die Texte für die weitere Verarbeitung in ausreichend guter Qualität verfügbar sind.

Datenaufbereitung

Als nächstes müssen die Texte für die weitere Verarbeitung aufbereitet werden. Dieser Schritt erfordert je nach Anwendungsfall bspw. bestimmte Satzzeichen und/oder überschüssige Leerzeichen zu entfernen oder Texte in Kleinschreibung zu konvertieren. Dadurch gehen zwar manche Informationen verloren, allerdings erleichtert dies sowohl die manuelle als auch die maschinelle Verwertung der Texte durch KI-Modelle erheblich. Ein weiterer essenzieller Schritt ist die Tokenisierung der Texte. Da Computer mit Wörtern nicht "rechnen" können, wird jedem Wort eine eindeutige Zahl zugewiesen, und alle Texte in dieses einheitliche Zahlenschema konvertiert.

Sprachmodelle

Moderne, deep-learning-basierte Sprachmodelle werden selbstüberwacht auf umfangreichen Textdatenbanken wie etwa BookCorpus vortrainiert. Ein sehr häufig verwendeter Ansatz ist dabei das sogenannte Masked Language Modelling, bei dem zufällige Satzteile (z.B. Wörter) geschwärzt werden und das Modell versucht, den Lückentext möglichst nahe zum Originaltext wieder zu befüllen. Damit das Modell ein gutes Verständnis für die Strukturen natürlicher Sprache aufbauen kann, sind Millionen an Beispielen und viele Iterationen dieses Ratespiels notwendig. Da dieser Prozess sehr ressourcenintensiv ist (hohe Rechenleistung und Kosten), werden diese meist von großen Organisationen wie bspw. Google oder Facebook vortrainiert und – dankenswerterweise – anderen Entwickler*innen öffentlich verfügbar gemacht.

Finetuning

Über das Prinzip des sogenannten Transfer-Learnings können vortrainierte Modelle ihr Sprachverständnis nun nutzen, um mit geringeren Datenmengen die Lösung konkreter Aufgaben (wie bspw. weiter oben bereits erläutert NER, Textklassifikation oder Sentimentanalyse) zu erlernen. Für dieses Finetuning sind je nach Komplexität der Aufgabenstellung einige hunderte bis tausende Beispieldaten notwendig.

Evaluierung

Die Qualität dieser Modelle wird anschließend über bereitgestellte Test- bzw. Validierungsdaten quantitativ bewertet. Je nach Aufgabe und Ziel werden dabei unterschiedliche Metriken herangezogen. Somit kann es notwendig werden, Modelle anhand mehrerer Metriken zu bewerten und zu vergleichen.

Produktiveinsatz

Die Vorhersagen der Modelle auf neuen Daten (Inferenz) liefern Ergebnisse entsprechend der Struktur aus den Beispieldaten und können damit in den Unternehmens-Workflow eingebunden werden. ›

Aktuelle Trends und Herausforderungen: Wenn KI's lernen wie Menschen zu schreiben, zeichnen und kommunizieren

In den letzten Jahren dreht sich im NLU-Bereich fast alles um die sogenannten Transformer-Modelle. Dabei handelt es sich um eine spezielle Architektur von künstlichen neuronalen Netzen, die besonders geeignet für den Umgang mit Textdaten ist (siehe auch [7]). Besondere Aufmerksamkeit erregte in den letzten Monaten beispielsweise Google's Language Model for Dialogue Applications – kurz: LaMDA (siehe [8]). Dieses Modell ist darauf trainiert, sich im Dialog möglichst menschlich zu verhalten, und diese Fähigkeit konnte das Modell bereits in mehreren "Interviews" beweisen (siehe [9]). Auch die von OpenAI entwickelten DALL·E Modelle (siehe [10]) können (unter anderem) zu einem Eingabetext passende Bilder erzeugen. Das Modell basiert auf der GPT-3 Architektur (siehe [11]), welche zuvor bereits durch ihre Fähigkeit, neue Texte in bisher unerreichter Qualität zu generieren, überzeugen konnte. Ein vereinfachtes, DALL·E nachempfundenes Modell ist unter craiyon.com öffentlich verfügbar: Was für durchschnittliche Internetuser*innen eine lustige Spielerei ist, kann auch zahlreiche produktive Anwendungen finden.

Die größte Herausforderung bei der Verwendung dieser neuen Modelle in innovativen Forschungsprojekten sind die für die jeweilige Aufgabe verfügbaren Daten. Für erfolgreiches Finetuning eines vortrainierten Modells auf eine neue Aufgabe sind entsprechende Daten notwendig, die dem Modell vorzeigen, was es zu tun hat. Diese Daten müssen auch in ausreichender Menge vorhanden sein und den festgelegten Datenqualitätskriterien entsprechen. Im Weiteren stellt auch die Auswahl des vortrainierten Modells eine Herausforderung dar. Um die besten Ergebnisse zu erzielen, ist eine Literaturrecherche und das Testen und Evaluieren unterschiedlicher Modelle unumgänglich.

Bei all diesen spannenden, neuen Innovationen den Überblick zu behalten, ist nicht immer einfach. Allerdings sollten hier auch nicht immer nur die neuesten Trends beachtet werden. Manche Aufgaben können auch mit älteren Methoden oder (in Kombination) mit ausgeklügelten regelbasierten Systemen gelöst werden, die teilweise effizienter in der Verwendung sind und auch die Nachvollziehbarkeit von Modellentscheidungen ad-hoc ermöglichen. Es lohnt sich daher definitiv, für einen ersten Prototyp auch bereits langfristig etablierte Methoden auszutesten.

| Fazit

Menschliche Sprache ist erstaunlich komplex und vielseitig. NLU-Lösungen verstehen und interpretieren sprachlich vermittelte Inhalte immer besser und der rasante Fortschritt wird immer beeindruckender. Fast täglich erhöht sich die Anzahl der öffentlich verfügbaren Modelle, und gleichzeitig zeigt sich, wie vielfältig diese bereits eingesetzt werden können. Mit der zunehmenden Digitalisierung sowie der Menge an routinemäßigen Abläufen steckt noch immer viel ungenutztes Potenzial in den unstrukturierten Textdaten der Unternehmen, um deren Prozesse und Produkte mit NLP-Lösungen auf das nächste Level zu heben. Wenn auch Sie am Einsatz solcher Technologien in Ihrem Unternehmen interessiert sind, unterstützen wir Sie gerne bei der Planung und Umsetzung von NLP-Projekten (https://www.risc-software.at/analyze-nlp/). ◆

Referenzen

[1] Wartner, Sandra (2021): „OK Google: Was ist Natural Language Processing?" – Wie Maschinen die menschliche Sprache lesen, entschlüsseln und verstehen (ris.w4.at/fachbeitrag-natural-language-processing-1)

[2] Wartner, Sandra (2021): Vom Informationsfluss zum Informationsgehalt – warum sich sauberes Daten(qualitäts)management auszahlt (ris.w4.at/fachbeitrag-datenqualitaet)

[3] Hochleitner, Christina (2021): Förderungen mit laufender Einreichmöglichkeit (https://www.risc-software.at/foerderungen-mit-laufender-einreichmoeglichkeit/)

[4] Wartner, Sandra (2021): Warum auch eine gute Idee eine Machbarkeitsstudie braucht (ris.w4.at/fachbeitrag-warum-auch-eine-gute-idee-eine-machbarkeitsstudie-braucht)

[5] Wartner, Sandra (2021): Wie wir vertrauenswürdige KI-Systeme schaffen und nutzen (ris.w4.at/fachbeitrag-vertrauen-in-die-kuenstliche-intelligenz)

[6] Jaeger, Anna-Sophie (2022): Explainable Artificial Intelligence (XAI) – Wie Machine Learning Vorhersagen interpretierbar(er) werden (ris.w4.at/fachbeitrag-explainable-artificial-intelligence)

[7] Wartner, Sandra (2022): Transformer-Modelle erobern Natural Language Processing (ris.w4.at/fachbeitrag-transformer-modelle-erobern-natural-language-processing)

[8] Thoppilan, Romal, et al. "Lamda: Language models for dialog applications." arXiv preprint arXiv:2201.08239 (2022).

[9] Lemoine, Blake (2022): "Is LaMDA Sentient? – an Interview" (https://cajundiscordian.medium.com/is-lamda-sentient-an-interview-ea64d916d917)

[10] OpenAI (2022): https://openai.com/dall-e-2/

[11] Brown, Tom, et al. "Language models are few-shot learners." Advances in neural information processing systems 33 (2020): 1877-1901.

Autorin

Sandra Wartner, MSc
Data Scientist

Warum auch eine gute Idee eine Machbarkeitsstudie braucht

DAMIT DEM ERFOLGREICHEN PROJEKTSTART NICHTS MEHR IM WEGE STEHT, UND DIE IDEE NICHT NUR EINE IDEE BLEIBT, SOLLTE EINE MACHBARKEITSSTUDIE (FEASIBILITY STUDY) VOR DER UMSETZUNG DURCHGEFÜHRT WERDEN.

von Sandra Wartner, MSc

Zu Beginn steht meist eine gute Idee. Dann soll diese alsbald in die Tat umgesetzt werden und ein Projekt gestartet werden. Sich vorschnell in eine Projektumsetzung zu stürzen ist allerdings nur in Ausnahmefällen eine gute Idee. Warum? Weil im Vorfeld unbedachte Einflussfaktoren schnell zum Scheitern eines Projekts führen können. Das ist dann nicht nur schlecht für das Geschäft, sondern auch schade um die gute Idee (und die Arbeit, die man in deren Umsetzung schon gesteckt hat). Vor dem Start ist es also wichtig, die Idee auf Umsetzbarkeit zu prüfen, mögliche Risikofaktoren zu erkennen und zu bewerten und dabei sowohl wirtschaftliche Einflussfaktoren wie etwa Finanzen und Personalressourcen als auch zeitliche Abläufe zu berücksichtigen. Wir stellen im nachfolgenden Text die Machbarkeitsstudie (Feasibility Study) als wichtiges Instrument vor jedem Projektstart vor, beschreiben ihre Ziele und zeigen einige praktische Methoden und Tools zur effizienten Durchführung. Damit dem erfolgreichen Projektstart nichts mehr im Wege steht – und die Idee nicht nur eine Idee bleibt.

Eine Machbarkeitsstudie (auch Projektstudie oder engl. feasibility study) schafft eine Entscheidungsgrundlage, ob bzw. wie ein Projekt durchgeführt werden kann und wird daher in der Initialisierungsphase vom Auftragnehmer bzw. der Auftragnehmerin durchgeführt. Das Ergebnis ist ein umfassender Bericht zur Durchführbarkeit eines Projektes, der neben rechtlichen und wirtschaftlichen Aspekten auch die Organisation und Zeitplanung, vor allem aber die technische Machbarkeit beleuchtet. Die Studie schließt damit in der Praxis meist eine umfassende Anforderungs- und auch Risikoanalyse mit ein. In welchem Umfang und mit welchen Methoden eine Machbarkeitsstudie durchgeführt wird, ist in der Praxis individuell und von Projekt zu Projekt (bzw. Projektidee) unterschiedlich.

Die Durchführung einer Machbarkeitsstudie ist jedenfalls dann besonders wichtig, wenn die Projektidee erst vage ist, es viele Unklarheiten gibt, nicht alle Beteiligten das gleiche Verständnis haben, die Ziele noch nicht klar sind oder einfach die technische Umsetzbarkeit fraglich ist. Auftraggeber*innen formulieren in der Anfangsphase meistens, was sie wollen, haben aber vor allem bei komplexen Projekten oftmals noch keine oder eine sehr unklare Vorstellung von der konkreten technischen Umsetzung, den zu verwendenden Technologien und generell der Machbarkeit.

Im Rahmen einer Anforderungsanalyse wird dann von Auftraggeber*in und Auftragnehmer*in gemeinsam erarbeitet, was der*die Auftraggeber*in wirklich braucht. Auf dieser Basis kann dann die Machbarkeit des Vorhabens geprüft werden. Verzichtet man zu Beginn auf eine Machbarkeitsstudie, ist die Gefahr groß, dass es spätestens in der Realisierungsphase zu unerwarteten Einschränkungen durch nicht bedachte Risiken kommt und Probleme auftreten.

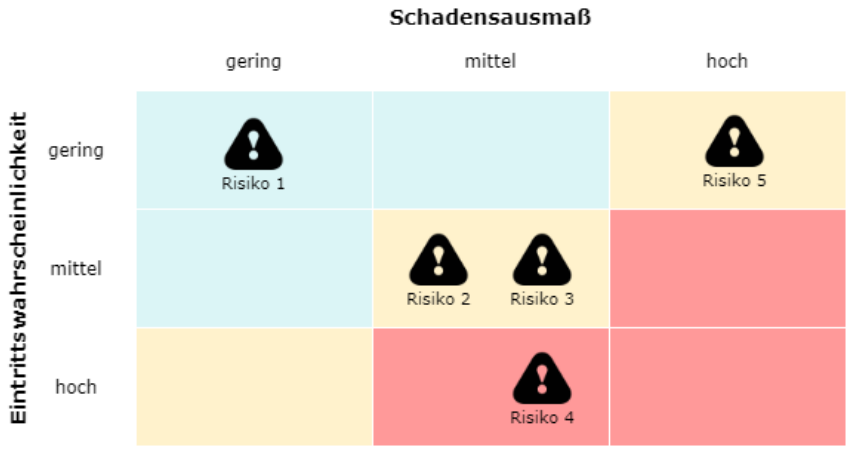
Mit dem passenden Methodenset zum Ziel

Die Inhalte bzw. Lösungsansätze einer Machbarkeitsstudie sind nicht in einer Norm festgelegt. Das ist auch gut so, denn jedes Projektvorhaben steht vor unterschiedlichen An- und Herausforderungen. Aus dem Pool an Möglichkeiten kann damit individuell auf die bestehenden Bedürfnisse und Rahmenbedingungen zugeschnitten ein passendes Set an Methoden ausgewählt werden. Im Vorfeld sollte man sich bereits Gedanken über ein paar wesentliche Voraussetzungen machen, ohne deren Erfüllung eine Machbarkeitsstudie nur wenig Sinn macht: Dies sind u.a. klar definierte Projektziele (SMART), strategische Ziele und Nicht-Ziele, Projekt-Grenzen sowie obligatorische und optionale Ergebnisse. Denn nicht alles, was technisch machbar ist, ist gleichzeitig auch zielführend oder zufriedenstellend für die Kund*innen. In Machbarkeitsstudien häufig verwendete Methoden schließen beispielsweise die folgenden ein:

In Machbarkeitsstudien häufig verwendete Methoden schließen beispielsweise die folgenden ein:

Reifegradanalyse
Anforderungsanalyse
Business Model Canvas
Risikoanalyse
SWOT-Analyse
Kosten-Nutzen-Analyse

Risikoanalyse



Ein kurzer Realitätscheck: Ein Projekt läuft im seltensten Fall reibungslos ab und positive Themen werden von Mitarbeiter*innen viel lieber bearbeitet als das Absichern gegen negative Zukunftsszenarien. Dennoch ist die Risikoanalyse ein unverzichtbarer Bestandteil einer Machbarkeitsanalyse, um für das Unerwartete gerüstet zu sein. Ziel der Risikoanalyse ist es, mehr Transparenz und Bewusstsein für das Projekt potenziell gefährdende Einflussfaktoren zu schaffen. Durch die vorzeitige Auseinandersetzung mit diesen Risiken kann sich das Projektteam vorbereiten und auch bereits präventive Maßnahmen treffen. Bei der Risikoanalyse sind drei Begriffe voneinander abzugrenzen:

- Eine **Ursache** ist ein existierender Faktor, der potenziell zu einem Risiko führen kann.
- Ein **Risiko** ist eine Unsicherheit, die mit einer gewissen Wahrscheinlichkeit eintritt und beim Eintreten zu einem Problem wird.
- Eine **Auswirkung** ist eine negative Konsequenz, wenn ein Risiko eintritt.

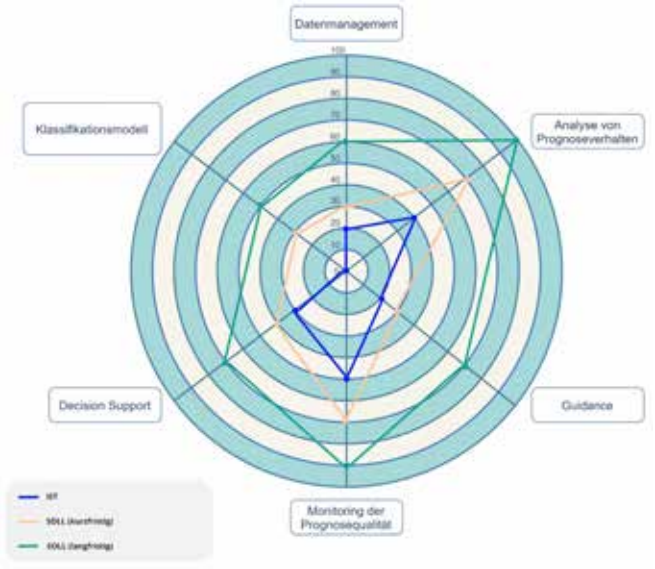
Risiken können in unterschiedlichen Bereichen wie bspw. in der Finanzierung, Planung oder dem Projektumfeld vorhanden sein. Nur wenn Risiken frühzeitig identifiziert, quantifiziert und Maßnahmen abgeleitet werden, kann das Schadensausmaß (z.B. Ressourcenprobleme, Mehrkosten, Zeitverzögerungen, Scheitern des Projekts) reduziert werden. Das Ergebnis ist eine Risikomatrix, die Informationen zu den identifizierten Risiken sowie das Schadensausmaß und auch die (geschätzte) Eintrittswahrscheinlichkeit beinhaltet. Da Risiken sich über die Zeit auch verändern können, sollte anstatt einer einmaligen Einschätzung aktives Risikomonitoring betrieben werden. ▸

Business Model Canvas

Ein sehr hilfreiches Framework zur strategischen Planung und Entwicklung bietet das Business Model Canvas. Der Nutzen liegt v.a. in der kompakten und visuellen Darstellung des Geschäftsmodells in einem skalierbaren System. Dies kann als Diskussionsbasis genutzt werden, um ein gemeinsames Verständnis aufzubauen und Abhängigkeiten auf einen Blick zu erkennen. Einerseits werden die zur Umsetzung notwendigen Mittel dargestellt (linke Hälfte), zum anderen werden Werte für Kundinnen und Kunden bzw. Unternehmen dokumentiert (rechte Hälfte). Die weitere Gliederung erfolgt in neun Elemente:

- **Nutzenversprechen/Mehrwert:** Was ist unser Nutzenversprechen den Kundinnen und Kunden gegenüber?
- **Kundensegmente:** Wer ist die Zielgruppe?
- **Kanäle:** Wie erreiche und informiere ich potenzielle Kundinnen und Kunden über das Angebot?
- **Kundenbeziehungen:** Wie soll die Beziehung zu Kundinnen und Kunden aussehen (von der Akquise bis hin zur Stammkundenpflege)?
- **Einnahmequellen:** Wie gestaltet sich das Preismodell, mit dem wir unser Geld verdienen?
- **Wichtige Ressourcen:** Welche Ressourcen (z.B. finanziell, personell, (im)materiell, technisch) sind notwendig?
- **Wichtige Aktivitäten:** Welche Tätigkeiten sind notwendig, um den Erfolg des Geschäftsmodells kontinuierlich gewährleisten oder erhöhen zu können?
- **Wichtige Partner:** Wer sind unsere strategischen Partner*innen?
- **Kostenstruktur:** Welche erfolgskritischen Kostenpunkte und Ausgaben müssen berücksichtigt werden?

Reifegradanalyse



Die Reifegradanalyse (engl. maturity analysis) ist ein praxisorientiertes Verfahren zur Zustandsermittlung einer bestimmten Gruppe von Faktoren bzw. Fähigkeiten. Dabei kann einerseits der aktuelle Reifegrad (IST) erhoben werden, und andererseits können kurzfristige oder langfristige Ziele (SOLL) als zukünftige Reifegrade im selben Diagramm dargestellt werden. Diese Analyse macht v.a. dann Sinn, wenn ein bestehendes Produkt bzw. System erweitert werden und die weitere Ausbaustufe klar definiert werden soll. Ein klarer Vorteil ergibt sich aus der anschaulichen Darstellung durch die visuell aufbereiteten Ergebnisse und den uneingeschränkten Freiheitsgrad an Faktoren. So erhält man auf einen Blick eine gute Übersicht über unterschiedliche Ausbaustufen und gleichzeitig eine gute Vergleichsbasis (IST vs. SOLL). Für die identifizierten Entwicklungspotenziale und Schwächen können dadurch gezielt Verbesserungsmaßnahmen abgeleitet und dadurch der Reifegrad erhöht werden.

Die Umsetzung eines solchen Maturity Charts erfolgt häufig über ein Radardiagramm. Dabei wird in jeder Stufe für jeden Faktor eine Punktzahl zw. 0 und 100 (vollkommen unausgereift bzw. voll ausgereift) vergeben, welcher den Reifegrad dieses Faktors in dieser Stufe beschreibt und im Anschluss innerhalb einer Stufe über Linien miteinander verbunden werden. Es ist allerdings nicht als negativ zu werten, wenn bestimmte Faktoren nie einen Reifegrad von 100 erreichen. Die tatsächlich notwendigen Reifegrade hängen von den vorliegenden Anforderungen zu bestimmten Zeitpunkten ab.

Prozessoptimierung mit dem richtigen Werkzeug

Zur bestmöglichen Einbindung und Beteiligung aller Stakeholder kann ein starker Fokus auf Interaktivität den Prozess zur gemeinsamen und erfolgreichen Ausarbeitung der Inhalte unterstützen. Damit wird sichergestellt, dass alle Sichtweisen gesammelt, die unterschiedlichen Anforderungen dokumentiert und die dadurch gewonnenen Erkenntnisse in die finale Evaluierung miteinbezogen werden können. So kann eine Machbarkeitsstudie effizient und zielführend durchgeführt werden. Situationsabhängig erweisen sich dabei unterschiedliche Werkzeuge als nützlich wie bspw. ein einfacher Flipchart oder Online-Werkzeuge wie bspw. Collaboard, Microsoft Whiteboard, InVision Freehand oder Miro.

Die RISC Software GmbH konnte mit der virtuellen Online Kollaborations-Plattform Miro (https://miro.com) bereits sehr viele positive Erfahrungen sammeln. Hier können gleichzeitig mehrere Personen in Echtzeit auf einem unendlich großen Online-Whiteboard visuell zusammenarbeiten. Das Tool lässt sich außerdem mit zahlreichen Apps (z.B. Google Drive, Video-Chat, etc.) erweitern, welche das Arbeiten im Team noch weiter vereinfachen und effizienter gestalten. Auch im Zuge einer Machbarkeitsstudie eignet sich ein solches Online-Tool hervorragend als Diskussionsbasis und Dokumentationswerkzeug. Anhand eines zuvor konstruierten Leitfadens zur Ausarbeitung einzelner Komponenten können die Inhalte diskutiert und in den Templates des Boards gemeinsam bearbeitet und dokumentiert werden. So wird auch in Covid-19 bzw. Home-Office-Zeiten ein gemeinsames Verständnis über die Durchführbarkeit eines Projektes geschaffen und Herausforderungen frühzeitig erkannt. Dadurch können von Beginn an die richtigen Maßnahmen gesetzt und Ressourcen bestmöglich verplant werden, sodass zeitnah mit einer erfolgreichen Umsetzung der initialen Projektidee begonnen werden kann.

Dieser Fachartikel entstand im Rahmen der FFG Förderung des Projekts "InnoFIT – Innovative Forecast- und Bedarfsanpassung durch die Nutzung von Vertriebsdaten aus neuen Informationstechnologien" (FFG Projektnummer: 867471). ◆

Autorin

Sandra Wartner, MSc
Data Scientist

Mathematische Modellierung von Produktionsproblemen

GAME CHANGER IN DER PRODUKTIONSPLANUNG

von Dr. Roman Stainko

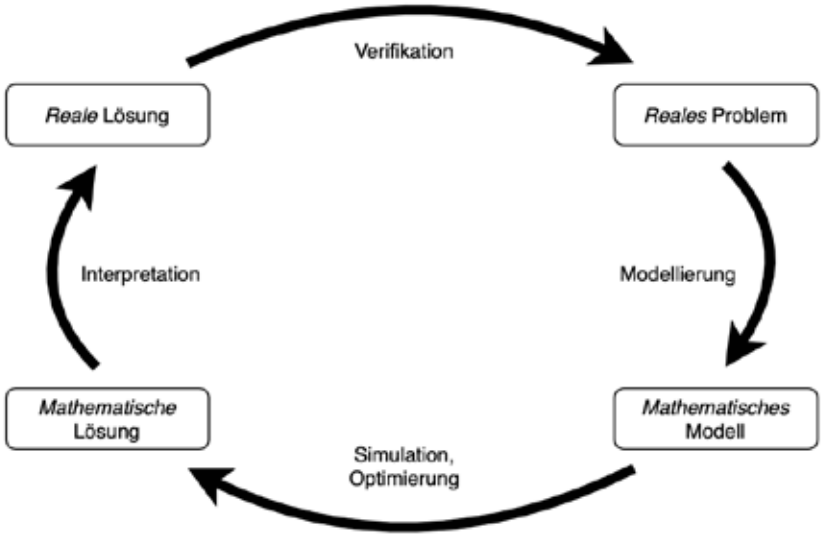
Mathematische Modelle werden überwiegend in der Wissenschaft verwendet, insbesondere der Naturwissenschaften. Klassische Einsatzgebiete sind hier die theoretische Physik oder zur Zeit sehr prominent die Epidemiologie. Aber auch, und hier natürlich besonders nah thematisch verbunden, im Gebiet von Operational Research, mit all den Fragestellungen der Unternehmensplanung und der Logistik.

Über mathematische Modelle und reale Probleme

Auf den ersten Blick mögen die Mathematik und ihre theoretischen Modelle mit der realen Welt wenig gemeinsam haben. Bereits in der Schule wird im selten beliebten Unterrichtsfach Mathematik oft die Frage gestellt: Wozu braucht man denn das? Riskiert man aber einen zweiten gezielten Blick, so lässt sich erkennen, dass mathematische Modelle hinter vielen Prozessen des Alltags und realen Fragestellungen liegen. Bloß wo? Beispielsweise: Die nagenden Überlegungen der Studenten, den kürzesten Weg zwischen den Stationen des pub crawls zu finden. Die Anstrengungen der Junggastronomin, eine ideale Preisgestaltung der Speisekarte vorzunehmen. Das Bestimmen der optimalen Mengen an Verpflegung und Getränken für das Sommerfest. Wie kann der Unternehmer Bruno Bank die Bedingungen seiner Geldgeberin erfüllen? Doch dazu etwas später. All diese Fragestellungen lassen sich mittels mathematischer Modelle beschreiben und folglich mit mathematischen Methoden berechnen und lösen.

Klarerweise werden mathematische Modelle überwiegend in der Wissenschaft verwendet, insbesondere in den Naturwissenschaften. Ganz klassisch und seit jeher in der theoretischen Physik, die mit ihren mathematischen Modellen physikalische Prozesse erklärt. Unter anderem auch in der Epidemiologie, die in letzter Zeit besonders ins Rampenlicht gerückt ist und mit passenden Modellen die Verbreitung von Viren beschreibt. Aber auch, und hier natürlich besonders nah thematisch verbunden, im Gebiet von Operational Research, mit all den Fragestellungen der Unternehmensplanung und der Logistik. Was kann man aber nun unter einem mathematischen Modell verstehen? Ein mathematisches Modell kann als Menge an mathematischen Funktionen, Vorschriften, Gleichungen und Ungleichungen gesehen werden, die in ihrer Gesamtheit eine reale Fragestellung in der Sprache der Mathematik beschreiben. Zielführenderweise sollte das Modell berechenbar sein, sodass man es mit mathematischen Werkzeugen auch berechnen und lösen kann. Für die Modellierung, also der Schritt von der realen Fragestellung zum Modell, gibt es leider kein allgemein gültiges Vorgangsrezept. Ein paar grundlegende Richtlinien helfen aber. Was ist wesentlich, auf was kann verzichtet werden? Es ist wichtig, nur jene Aspekte zu modellieren, die für den Prozess wirklich relevant sind. Bei einem Modell handelt es sich in der Regel immer um eine Vereinfachung der realen Ausgangslage. Wie genau müssen die Antworten sein? Auch lohnt es, sich Gedanken über die verwendeten Skalen und Einheiten (z.B. Ort und Zeit) zu machen. Die Verwendung von passenden Skalen hilft, das Modell in einer berechenbaren sinnvollen Größe zu halten. Und zu guter Letzt, welche Ziele sollen erreicht werden? Es ist oft nicht offensichtlich, von einer realen Fragestellung eine klare Zielvorgabe abzuleiten, für ein gutes Modell ist dies aber unerlässlich.

Die Kunst der mathematischen Modellierung liegt nun also darin, ein möglichst einfaches Modell zu erstellen, welches die ausgehende reale Fragestellung möglichst gut beschreibt. Dabei ist der Prozess der Modellierung kein fixer Ablauf von sequenziellen Arbeitsschritten. Es gibt keine feste Vorgehensweise, keine Rezeptur die immer zum Erfolg führen wird. Vielmehr kann man den Modellierungsprozess schematisch als Kreislauf verstehen, dessen Schritte in der Regel öfters durchlaufen werden müssen.



Ein kleines Beispiel zum Kennenlernen – Bruno Banks Bierbank Garnituren

Bruno Bank hat ein kleines StartUp-Unternehmen ins Leben gerufen. Er will innovative Bierbank-Garnituren produzieren und verkaufen. Allerdings besitzt er keine eigene Werkstatt und muss diese für die Produktion seiner Garnituren pro Monat um 1000 € mieten, unabhängig davon, wie viele Garnituren produziert werden. Die Herstellung einer Garnitur kostet Bruno 25 €. Die Produktion einer Garnitur pro Monat würde Bruno also beispielsweise 1025 € und von 100 Garnituren 3500 € kosten. Mehr als 500 Garnituren pro Monat kann Bruno allerdings nicht herstellen, denn er arbeitet alleine. Für die produzierten Garnituren steht ein Lager zur Verfügung, wobei pro Garnitur durchschnittlich 2 € Lagerkosten pro Monat anfallen. Die fertigen Garnituren können gut gestapelt werden, somit kann von unbegrenztem Lagerraum ausgegangen werden. 10 Garnituren hat Bruno bereits gefertigt und liegen im Lager.

Aus bereits getätigten Anfragen und Bedarfsanalysen konnte Bruno prognostizierte monatliche Absatzzahlen erstellen, die im Sommer die größten Werte aufweisen:

	Jan	Feb	Mär	Apr	Mai	Jun	Jul	Aug	Sep	Okt	Nov	Dez
Bedarf	50	50	50	100	200	200	200	100	100	50	50	50

Tabelle 1: Bedarfsprognose für Bierbank Garnituren für das kommende Jahr

Als Geldgeberin für seine Idee mit den Bierbänken hat Bruno Bank die Investorin Gabi Geld gewinnen können, die ihn mit 37000 € unterstützen wird. Allerdings stellt sie dafür eine Bedingung an seine Geschäftsplanung. Die Produktion muss auf die Bedarfsprognose abgestimmt werden und die sich daraus ergebenden Produktionskosten müssen sich durch das zur Verfügung gestellte Geld abdecken lassen. Aber wie soll er das bloß erreichen? Bruno beschließt ein paar Produktionsszenarien aufzustellen ▸

und deren Kosten zu berechnen. Vielleicht kann er so eine kostenminimale Produktion finden. Zwei Szenarien erscheinen ihm besonders vielversprechend. Einmal soll pro Monat genau die prognostizierte Menge produziert werden (Szenario A), um das Lager nicht zu benötigen. Das andere Mal soll die gesamte jährliche Bedarfsmenge gleich am Jahresanfang produziert werden (Szenario B), um die Werkstatt dann nicht mehr zu benötigen. Für beide Produktionsszenarien berechnet Bruno die voraussichtlichen Kosten und erhält 41760 € für Szenario A und 43900 € für Szenario B. Wobei sich die Kosten aus monatlicher Lagerhaltung, Werkstattmiete und Produktion ergeben. Wie Bruno die Kosten der Werkstattmiete und der Produktion zu berechnen hat, war ihm sofort klar. Bei den Lagerkosten nimmt er schließlich an, dass sich die eingelagerte Menge vom Stand am Monatsanfang zum Stand am Monatsende gleichmäßig verändert *(linearer Verlauf)*.

Szenario A	Jan	Feb	Mär	Apr	Mai	Jun	Jul	Aug	Sep	Okt	Nov	Dez	Summen
Bedarf	50	50	50	100	200	200	200	100	100	50	50	50	1200
Produktion	40	50	50	100	200	200	200	100	100	50	50	50	1190
Prod.-Kosten	1000 €	1250 €	1250 €	2500 €	5000 €	5000 €	5000 €	2500 €	2500 €	1250 €	1250 €	1250 €	29750 €
Wst. Miete	1000 €	1000 €	1000 €	1000 €	1000 €	1000 €	1000 €	1000 €	1000 €	1000 €	1000 €	1000 €	12000 €
Lagerstand	10	0	0	0	0	0	0	0	0	0	0	0	
Lagerkosten	10 €	0	0	0	0	0	0	0	0	0	0	0	10
Monatskosten	2010 €	2250 €	2250 €	3500 €	6000 €	6000 €	6000 €	3500 €	3500 €	2250 €	2250 €	2250 €	**41760 €**

Tabelle 2: Berechnete Kosten für Szenario A (minimale Lagernutzung)

Szenario B	Jan	Feb	Mär	Apr	Mai	Jun	Jul	Aug	Sep	Okt	Nov	Dez	Summen
Bedarf	50	50	50	100	200	200	200	100	100	50	50	50	1200
Produktion	500	500	190	0	0	0	0	0	0	0	0	0	1190
Prod.-Kosten	12500 €	12500 €	1250 €	0	0	0	0	0	0	0	0	0	29750 €
Wst. Miete	1000 €	1000 €	1000 €	0	0	0	0	0	0	0	0	0	3000 €
Lagerstand	460	910	1050	950	750	550	350	250	150	100	50	0	
Lagerkosten	470 €	1370 €	1960 €	2000 €	1700 €	1300 €	900 €	600 €	400 €	250 €	150 €	50 €	11150 €
Monatskosten	12970 €	14870 €	7710 €	2000 €	1700 €	1300 €	900 €	600 €	400 €	250 €	150 €	50 €	**43900 €**

Tabelle 3: Berechnete Kosten für Szenario B (minimale Werkstattnutzung)

Das erste Produktionsszenario A scheint bereits ganz gut zu sein, aber seiner Investorin Gabi Geld ist die geplante Produktion noch zu kostenintensiv. Gibt es jedoch noch deutlich kostengünstigere Möglichkeiten? Um wirklich sicher zu gehen, bittet er seinen alten Schulfreund und Mathematiker, Otto Optimal, um Hilfe.

Otto Optimal ist sowohl von Brunos Bierbänken als auch von seiner Fragestellung der Produktionsplanung begeistert und wird seinem alten Freund mit seinem Wissen über Optimierungsprobleme unterstützen. Auch soll er dafür eine von Brunos Garnituren bekommen. Um Brunos Produktionsproblem zu lösen, wird Otto zuerst ein mathematisches Modell der Problemstellung erstellen, welches er dann mit einem passenden Lösungsalgorithmus zur Optimalität lösen wird. So wird zweifelsfrei die optimale Lösung, bzw. das optimale Produktionsszenario, gefunden werden.

Als ersten Schritt der Modellierung überlegt sich Otto Variablen für das Modell. Gesucht sind die idealen monatlichen Produktionsmengen der Garnituren. Somit liegen 12 Zeitperioden *(nT = 12)* und ein Produkttyp (Bierbank Garnitur) vor. Die Produktionsmenge pro Monat soll die Variable $x_t \in [\ 0\ ,\ 500\ ]$ für t = 1, ... *nT*, beschreiben, da ja die Produktion mit 500 Stück pro Monat beschränkt ist. Weiters wird $i_t \geq 0$ den Lagerstand ($i_0 = 10$ für den initialen Lagerstand) und s_t die Verwendung der Werkstatt pro Monat bezeichnen (*0* ... keine Verwendung, *1* ... Verwendung). Aber wie sollen sich diese Variablen nun verhalten? Notwendige Nebenbedingungen werden in weiterer Folge das Zusammenspiel der Variablen festlegen. Als zweiten Schritt der Modellierung identifiziert Otto zwei solcher Nebenbedingungen: die *Materialflussgleichung*, welche die Lagerstände und deren Ab- und Zuflüsse pro Monat beschreibt, und die *Werkstattungleichung*, welche die Verwendung der Werkstatt be-

schreibt. Der Lagerstand i_t im Monat *t* lässt sich durch den Lagerstand des Vormonats i_{t-1}, durch die Produktion x_t und den Bedarf d_t des aktuellen Monats *t* charakterisieren:

$$i_t = i_{t-1} + x_t - d_t \quad \text{für} \quad t = 1, \ldots, nT.$$

Die Verwendung der Werkstatt ist notwendig sobald mindestens eine Garnitur produziert wird. Diese Eigenschaft wird durch die folgende Ungleichung beschrieben:

$$x_t \leq M \cdot s_t \text{ für } t = 1, \ldots, nT,$$

wobei M für eine obere Schranke für den Wertebereich von $x_t \in [\,0\,,\,500\,]$ steht (maximale Produktionsmenge pro Monat), welche in diesem Fall mit 500 gegeben ist.

Als letzten Schritt beschreibt Otto die Kosten der Produktion mittels einer Kostenfunktion (oder einer Zielfunktion). Die anfallenden Kosten unterteilen sich in Kosten für Produktion, Lager und Werkstatt und werden durch die Koeffizienten cp, ci und cs abgebildet (cp = 25, ci = 2, cs = 500). Die Kosten pro Monat t lassen sich somit wie folgt beschreiben:

$$c_p \cdot x_t + c_s \cdot s_t + c_i \frac{i_{t-1} + i_t}{2}.$$

Somit kann Otto nun das gesamte Optimierungsmodell formulieren. Es gilt, die Ziel-

Variable	Bedeutung
i_t	Lagerstand pro Woche t
x_t	Werkstattbenutzung pro Woche t
s_t	Produktionsmenge pro Woche t

funktion unter Einhaltung der Nebenbedingungen durch optimale Wahl der Variablen zu minimieren:

Minimiere die Zielfunktion

$$\sum_{t=1}^{nT} c_p \cdot x_t + c_s \cdot s_t + c_i \frac{i_{t-1} + i_t}{2}$$

unter Einhaltung der angeführten Nebenbedingungen

$$i_0 = 10\,,$$

$$i_t = i_{t-1} + x_t - d_t \text{ für } t = 1, \cdots, nT,$$

$$x_t \leq M \cdot s_t \text{ für } t = 1, \cdots, nT,$$

$$i_t \geq 0 \text{ für } t = 1, \cdots, nT,$$

$$x_t \in [0, 500] \text{ für } t = 1, \cdots, nT,$$

$$s_t \in \{0, 1\} \text{für } t = 1, \cdots, nT.$$

Wobei die vorkommenden Variablen folgende Bedeutung haben:

Bei genauer Betrachtung fällt Otto auf, dass alle vorkommenden Funktionen und Gleichungen linear in den Variablen sind, welche selber binär (diskret) (s_t) oder stetig sind (i_t , x_t). Bei einem solchen Modell spricht man von einem gemischt ▸

ganzzahligen (linearen) Modell *(mixed integer (linear) program ... MI(L)P)*. Das vorliegende Modell ist eine vereinfachte Variante des in der Literatur bekannten *Capacitated Lot-Sizing Problems (CLSP)*, welches in das Gebiet der dynamischen Losgrößenplanung fällt.

Otto löst dieses Optimierungsmodell nun mit einem ihm verfügbaren MIP-Lösungswerkzeug, welches einen branch-and-bound bzw. branch-and-cut Algorithmus realisiert. Dafür schreibt er sein mathematisches Modell in einer Modellierungssprache, damit das Lösungswerkzeug das Modell lesen und anschließend lösen kann.

```
int nT = 12;          // Anzahl der Zeitbereiche (Monate)
range T = 1 .. nT;    // Indexmenge aller Zeitbereiche
range T0 = 0 .. nT;   // Indexmenge aller Zeitbereiche inkl. anfänglichem Zustand (Index 0)

int setupCosts = 1000;       // Mietkosten der Werkstatt pro Monat
int prodCosts =    25;       // Produktionskosten per Bierbank
int invCosts =      2;       // Lagerkosten pro Bierbank pro Monat

int invo = 10;        // Anfänglicher Lagerstand
int M = 500;          // Maximale Produktionsmenge pro Monat

int demand[ T ] = [ 50, 50, 50, 100, 200, 200, 200, 100, 100, 50, 50, 50 ];

dvar int x[T] in 0 .. M;     // Variable für die produzierte Anzahl an Bierbänken
                                pro Monat im Wertebereich 0 bis M;
dvar float+ i[ T0 ];         // Variable für den Lagerstand an Bierbänken pro Monat
                                mit dem Wertebereich nichtnegativer reeller Zahlen
dvar boolean s[ T ];         // Variable für die Verwendung der Werkstatt pro Monat
                                mit dem Wertebereich 0 und 1
                             // (0 .. keine Verwendung, 1 .. Verwendung)

// Zielfunktion (zu minimierender Wert):
minimize
sum ( t in T ) ( prodCosts * x[t] + setupCosts * s[t] ) + sum( t in T ) ( 0.5 * invCosts * ( i[t-1] + i[t] ) );

// Minimierung der Zielfunktion unter der Einhaltung folgender Bedingungen:
subject to {

      i[0] == inv0; // Anfangsbestand
      forall( t in T) {
             i[t] == i[t-1] + x[t] - demand [t];   // Materialflussgleichung für alle Zeitbereiche
             x[t] <= M * s[t];                     // Werkstattungleichung für alle Zeitbereiche
      }
};
```

Erwartungsgemäß unterbietet der optimale Zielfunktionswert von 35860 € die Werte der beiden Produktionsszenarien von Bruno, sogar um ca. 15 %. Aber insbesondere wird durch das optimale Produktionsszenario die Vorgabe der Investorin Gabi Geld eindeutig erfüllt.

	Jan	Feb	Mär	Apr	Mai	Jun	Jul	Aug	Sep	Okt	Nov	Dez	Summen
Bedarf	50	50	50	100	200	200	200	100	100	50	50	50	1200
Produktion	240	0	0	0	400	0	300	0	250	0	0	0	1190
Prod.-Kosten	6000 €	0	0	0	10000 €	0	7500 €	0	6250	0	0	0	29750 €
Wst. Miete	1000 €	0	0	0	1000 €	0	1000 €	0	1000 €	0	0	0	4000 €
Lagerstand	200	150	100	0	200	0	100	0	150	100	50	0	
Lagerkosten	210 €	350 €	250 €	100 €	200 €	200 €	100 €	100 €	150 €	250 €	150 €	50 €	2110 €
Monatskosten	7210 €	350 €	250 €	100 €	11200 €	200 €	8600 €	100 €	7400 €	250 €	150 €	50 €	**35860 €**

Tabelle 4: Optimales Produktionsszenario

Stolz präsentiert Otto am nächsten Tag Bruno die optimale Lösung (wie in Tabelle 4). Bruno kann nun die Bedingung von Gabi Geld erfüllen und eine hinreichend kostengünstige und sogar kostenoptimale Planung der Bierbankproduktion vorlegen.

Abschließend treffen sich Bruno, Gabi und Otto zu einem Austausch über eine weitere Zusammenarbeit. Natürlichen genießen sie dabei ein kühles Bier auf einer von Brunos Garnituren, sehr zum Wohle. ◆

Referenzen

H. Paul Williams, Model Building in Mathematical Programming, 5th edition, 2013, John Wiley & Sons Ltd.

Laurence A. Wolsey, Integer Programming, 1998, John Wiley & Sons Inc.

Yves Pochet, Laurence A. Wolsey, Production Planning by Mixed Integer Programming, 2006, Springer Science+Business Media, Inc.

George L. Nemhauser, Laurence A. Wolsey, Integer and Combinatorial Optimization, 1999, John Wiley & Sons Inc.

Autor

DI (FH) Andreas Lettner
Head of Unit Domain-specific Applications, Head of Coaches

Gute Software ist grün: Wie umweltbewusste Entwicklung zu besserer Software führt

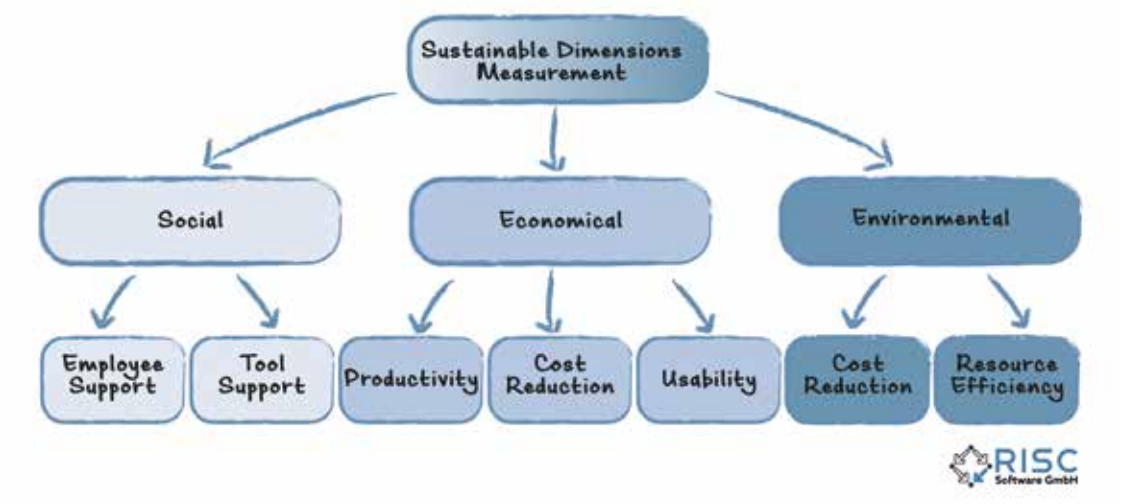
von Yvonne Marneth, BSc

Nachhaltigkeit ist ein Wert, der in allen Lebensbereichen immer wichtiger wird, auch in der Welt der Software-Entwicklung. Die "grüne" Software-Entwicklung zeigt uns Strategien und Denkweisen, unsere Applikationen nicht nur umweltschonender, sondern gleichzeitig auch günstiger und robuster zu gestalten.

Der Wunsch nach umweltverträglicher Software

Obwohl der Energieverbrauch im digitalen Bereich hauptsächlich durch den Einsatz von Hardware entsteht, wird dieser doch zu einem großen Teil von Software ausgelöst. Software nimmt also einen entscheidenden Einfluss auf deren Energieeffizienz. Eine umweltbewusste Einstellung bei der Entwicklung einer Applikation leistet einen bedeutenden Beitrag unsere Umwelt zu schützen. Grüne Software-Entwicklung möchte die Umweltbelastung in Form von Energieverbrauch, Treibhausgasemissionen und CO_2-Fußabdruck durch unsere Softwareprodukte minimieren, indem es nachhaltige Softwareentwicklungspraktiken, Architektur und Hardware in im Entwicklungsprozess einbringt.

Die Bereitschaft unsere Arbeit grüner zu gestalten ist oft da, doch welche konkreten Schritte können Entwickler tun, um ihre Applikationen bewusst nachhaltiger zu gestalten? Welche Kriterien machen eine Software umweltverträglich? Und sind diese überhaupt messbar?



Die 8 Prinzipien des „Green Coding“

Nachhaltigkeit in der Software-Entwicklung ist eigentlich kein neuer Gedanke. Der Fokus hat sich allerdings von den Bereichen Wirtschaftlichkeit und Änderungsfreundlichkeit nun ebenfalls auf den Bereich des Umweltschutzes ausgeweitet. Oft wird bei „Grüner Software“ zuerst an Software gedacht, deren primäre Aufgabe es ist, Prozesse im Sinne des Umweltschutzes zu verbessern. Dass auch die Architektur und der Entwicklungsprozess einer beliebigen Software nachhaltig sein kann, ist ein eher neuer Gedanke. Seit 2021 beschäftigt sich die „Green Software Foundation“, eine Non-Profit-Organisation, damit unter anderem Netzwerke, Guides, Design Patterns und Werkzeuge zu erarbeiten, um nachhaltige Softwareentwicklung für Entwickler und Entwicklerinnen zugänglich zu machen. Ein Ergebnis dieser Bemühungen sind acht fundamentale Prinzipien für die Grüne Software-Entwicklung:

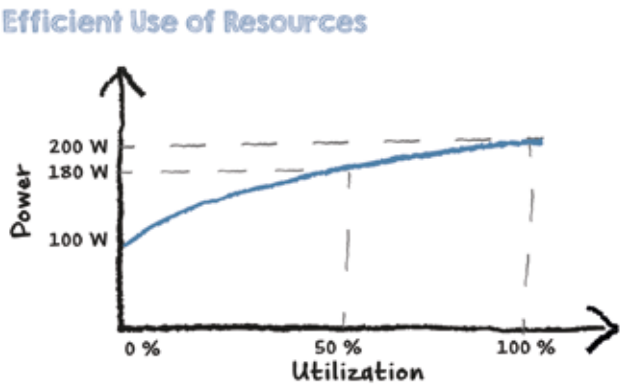
1) **Carbon: Build applications that are carbon efficient** Treibhausgase tragen dazu bei, die Temperatur auf der Erde kontinuierlich zu erhöhen. Die Reduktion von Treibhausgasen ist also notwendig, auch wenn diese durch die Entwicklung und den Betrieb von Software verursacht werden. Software an sich bewirkt nur indirekt die Freisetzung von Treibhausgasen. Sie kann aber auch dazu verwendet werden, die Produktion von Treibhausgase in anderen Bereichen zu minimieren, zum Beispiel durch den Einsatz von Software zum Monitoring und Verbesserung von anderen Prozessen.

2) **Electricity: Build applications that are energy efficient** Es gibt bereits vielversprechende mathematische Modelle, welche die Energieeffizienz verschiedener Funktionen in einer Applikation rechnerisch annähern und bereits recht gute Kennzahlen liefern. Auch der Einsatz von KI-basierten Systemen haben bereits gute Ergebnisse geliefert, um die Energieeffizienz von Software zu bewerten – wobei hier allerdings auch bemerkt werden muss, dass neuronale Netze selbst meist einen höheren Stromverbrauch verursachen als klassische Rechenmodelle. Es gibt bereits verschiedene Werkzeuge für Entwickler*innen, um die Energieeffizienz ihres Codes zu messen. Die Ergebnisse vieler Verfahren variieren jedoch stark, basierend darauf, wie, in welchen Zeiträumen und in welchem Kontext der Energieverbrauch betrachtet wird. Das macht es schwer, digitale Produkte untereinander zu vergleichen, sie können aber trotzdem gute Anhaltspunkte liefern, an welchen Stellen eine Applikation Verbesserungspotenziale hat.

3) **Carbon Intensity: Consume electricity with the lowest carbon intensity** Strom in einem lokalen Netz setzt sich aus verschiedenen Quellen zusammen. Große Teile werden noch immer aus fossilen Brennstoffen gewonnen. Der Anteil, der durch erneuerbare Energiequellen erzeugt wird, hängt stark von der geographischen Lage und der dort vorherrschenden energiepolitischen Einstellung ab. Selten haben Einrichtungen großen Einfluss darauf, wie der Strom, den sie verbrauchen, erzeugt wird. Meist es nicht einmal klar nachvollziehbar. Diesem Problem kann man allerdings bereits gegensteuern, indem beim Betrieb einer Applikation Serverstandorte vermieden werden, bei denen bereits bekannt ist, dass große Teile des lokalen Netzes durch nicht-umweltfreundliche Quellen gespeist werden.

4) **Embodied Carbon: Build applications that are hardware efficient** Ein ständiger Konflikt der Software-Entwicklung besteht in dem Drang, auf dem neuesten Stand der Technik zu bleiben. Die Umweltbelastung, die durch einen physischen Server entsteht, beschränkt sich jedoch nicht nur auf den Stromverbrauch während der Laufzeit, sondern kommt bereits bei der Herstellung eines Rechners, sowie später bei seiner Entsorgung zu tragen. Die Umweltbelastung durch seine Herstellung und Entsorgung kann dabei sogar höher ausfallen als durch die Energie, die während des Betriebs benötigt wird. Sinnvoll wäre es also, Server so lange wie möglich in Betrieb zu halten. Grundsätzlich nutzen sich Computer nicht ab, da sie keine beweglichen Teile haben. Trotzdem gibt es ein Verfallsdatum: Zum Beispiel, wenn sie modernen Workloads nicht mehr gewachsen sind, müssen sie ersetzt werden. Software, die mit älteren Geräten kompatibel ist, hilft dabei, das Leben von Hardware zu verlängern und dadurch ihre Umweltbelastung zu verringern.

5) **Energy Proportionality: Maximise the energy efficiency of hardware** Gleichzeitig ist das Verhältnis zwischen der Auslastung und dem Energieverbrauch nicht proportional, auch weil ein laufender Computer immer einen bestimmten Grundverbrauch hat. Je besser ein Computer ausgelastet ist, desto energieeffizienter funktioniert das Gerät. Server können ihren Energieverbrauch bei geringer Last nur bedingt reduzieren, sodass sie sofort auf Anfragen reagieren können, wenn diese gemacht werden. Durch laufende Server-Skalierung kann in Zeiten niedriger Auslastung Energie gespart werden, indem die Software auf weniger physischen Maschinen betrieben wird. Auch die “Serverless”-Architektur hilft dabei, den Stromverbrauch in Ruhezeiten minimal zu halten.



6) **Networking: Reduce the amount of data and distance it must travel across the network** Eine eher unsichtbare Umweltbelastung besteht im Internet selbst. Das Internet besteht aus einem weltweiten Netzwerk von Geräten, welche zusätzliche Energie verbrauchen. Jede Anfrage über das Internet verursacht Energieverbrauch. Wie groß dieser ist, hängt vor allem von der Distanz zwischen Client und Server ab, wie viele Netzwerkgeräte die Anfrage weiterleiten müssen, wie diese betrieben werden etc. – aber auch von dem verwendeten Protokoll. Eine ressourceneffiziente Software sollte also unnötigen Datentransfer vermeiden, sowie die passenden Protokolle wählen, um Daten möglichst effizient zu übertragen. ▸

7) **Demand Shaping: Build carbon-aware applications** Demand Shaping beschäftigt sich mit der Praxis den Service, bzw. die Qualität des Service umgekehrt an die Nachfrage anzupassen. Video Conferencing Software beispielsweise reduziert die Videoqualität und priorisiert Audioübertragung zu Zeiten hoher Auslastung. Diese Methodik kann verwendet werden, um Applikationen umweltfreundlicher zu gestalten. So kann die User-Experience angepasst werden, wenn die Kosten des Betriebs zu hoch werden. Da es hier bisweilen zu Kosten in der Verwendbarkeit kommt, kann ein „Eco-Mode" auch als bewusste Entscheidung während der Verwendung angeboten werden.

8) **Measurement & Optimisation: Focus on step-by-step optimisations that increase the overall carbon efficiency** Es ist nicht notwendig, alle Eventualitäten aus dem Vorhinein abzuwägen. Einzelne bewusste Entscheidungen können bereits einen großen Unterschied machen. Wichtig ist es vor allem, Entwicklungen zu beobachten und kontinuierlich Schritt für Schritt Verbesserungen einzubringen. Diese Vorgehensweise ist nicht nur im Sinne der Nachhaltigkeit, sondern sorgt gleichzeitig auch für eine anpassungsfähige Software mit einer proaktiven Wartung.

Die Kosten grünerer Software

Wer eine umweltfreundliche Software möchte, muss bei ihrer Entwicklung viele Parameter im Blick behalten und sich Gedanken über Möglichkeiten und Trade-offs machen. Ein großer Faktor bei diesen Entscheidungen spielt auch die Frage nach den Kosten dieser Veränderungen. Tatsächlich decken sich viele der Merkmale einer umweltschonenden Software aber mit denen einer qualitativ hochwertigen und im Betrieb sehr günstigen Software.

Dadurch, dass bewusst Ressourcen eingespart und achtsam eingesetzt werden, können die Betriebskosten gesenkt werden. Da potenziell weniger physische Server, Arbeitsspeicher und Rechenleistung benötigt werden. Die Benutzererfahrung muss unter solchen Einsparungen meist nicht leiden – im Gegenteil. Eine effiziente Software ist für gewöhnlich schneller, zuverlässiger und belastet die Geräte der Nutzenden weniger. Die Priorisierung von Ressourceneffizienz lenkt den Fokus zurück auf die Kernfunktionalität eines Produkts und so zu Verbesserungen im UX Design führen, indem überflüssige Elemente identifiziert und eliminiert werden.

Der bewusste Umgang mit Ressourcen und deren Monitoring führt zu einer zuverlässigen, sicheren, langlebigen und anpassungsfähigen Software. Die leicht höheren Kosten, die bei der initialen Entwicklung der Applikation möglicherweise anfallen, können also als Investition in eine zukunftsstarke Software betrachtet werden.

Die RISC Software GmbH unterstützt Sie gerne bei der Entwicklung umweltfreundlicher Software und bringt ihre Expertise ein. ◆

Autorin

Yvonne Marneth, BSc
Software Developer

RISC Software GmbH
www.risc-software.at

linkedin.com/company/risc-software-gmbh

twitter.com/RISC_Software

facebook.com/RISC.Software

xing.com/pages/riscsoftwaregmbh

Impressum
Herausgeber und
Medieninhaber:
RISC Software GmbH,
Softwarepark 32a, 4232 Hagenberg,
+43 7236 93028, office@risc-software.at
Für den Inhalt verantwortlich: DI Wolfgang Freiseisen
Chefredaktion: Mag. Cornelia Staub
Design und Grafische Gestaltung: Ladan Ghezel Ayagh

Ausgabe 2023 Version: 1.0 | 06.10.2025
Bildnachweis: RISC Software GmbH, iStock.com,
AdobeStock, generated with midjourney

Aus Gründen der besseren Leserlichkeit verzichten wir teilweise auf geschlechtsspezifische Formulierungen. Sämtliche personenbezogenen Bezeichnungen beziehen sich auf alle Geschlechter in gleicher Weise.