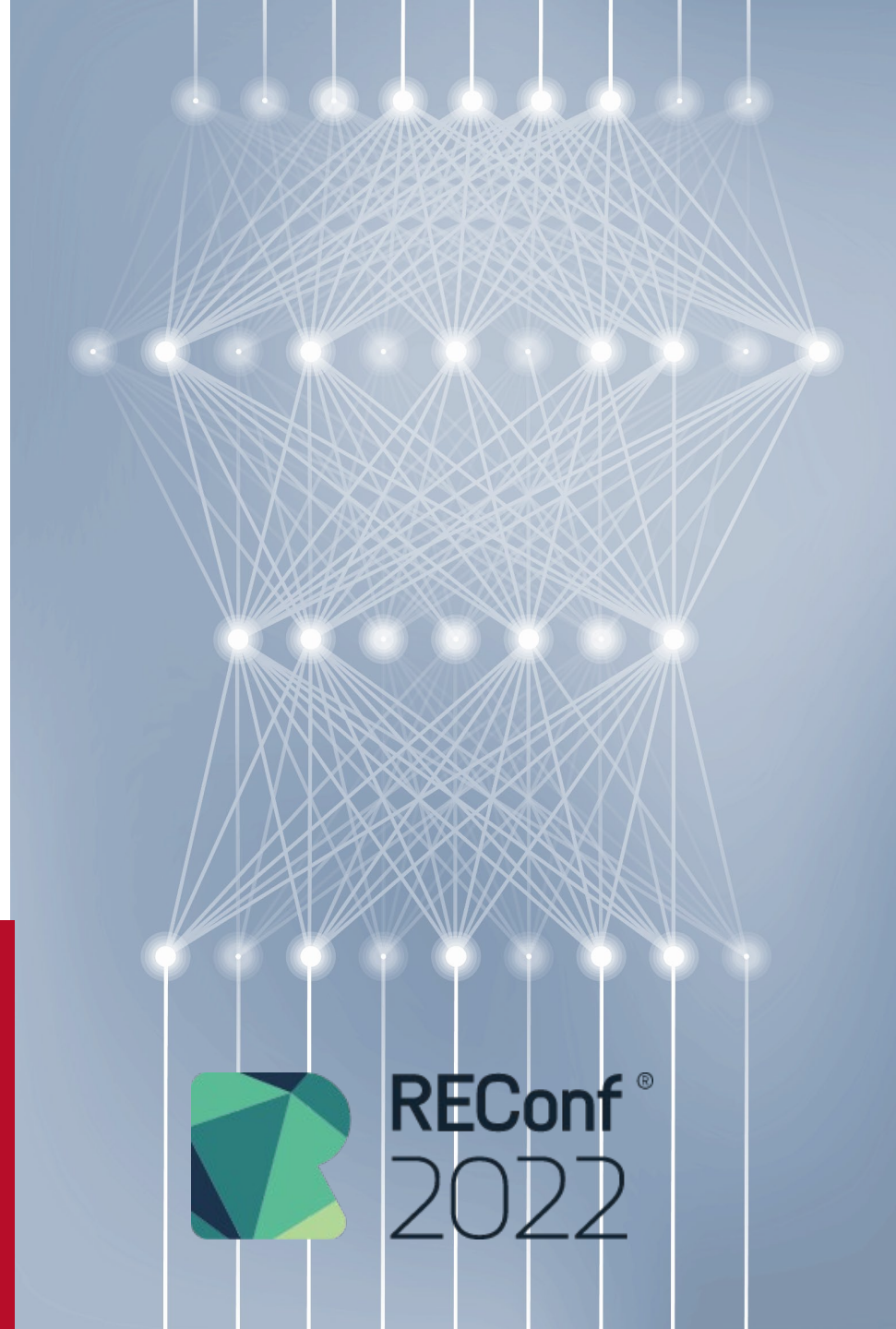
Chancen und Hürden der KI für das Requirements Engineering

Ein Projektbericht

Ursula Meseberg
microTOOL GmbH Berlin



KI – erste Begegnung

Hannover 1986

Ich kann Sie nicht verstehen.
Bitte wiederholen.

Ich kann Sie nicht verstehen.
Bitte wiederholen.

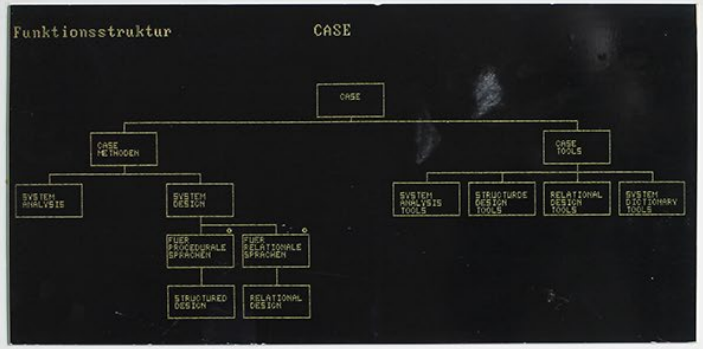
Ich kann Sie nicht verstehen.
Bitte wiederholen.

Ich kann Sie nicht verstehen.
Bitte wiederholen.

Ich kann Sie nicht verstehen.
Bitte wiederholen.

Ich kann Sie nicht verstehen.
Bitte wiederholen.

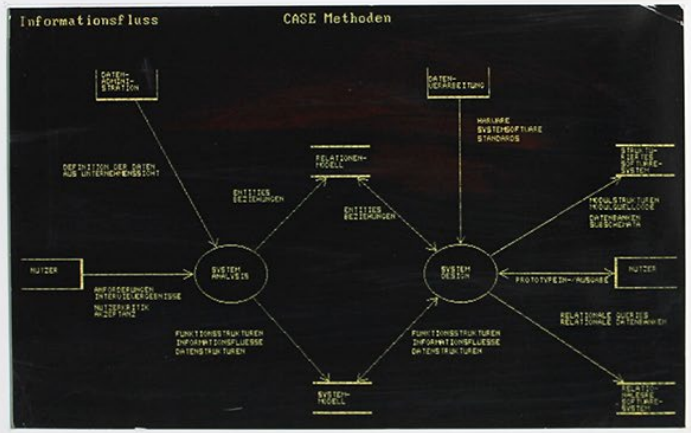
Ein PC-gestütztes, grafisch-orientiertes Verfahren zur Systementwicklung



CASE ist ein hochintegriertes Methoden- und Toolpaket für die Entwicklung von Softwaresystemen.
Die CASE METHODEN beschreiben, welche Ergebnisse pro Projektphase erarbeitet werden müssen und wie dabei vorzugehen ist.
Mit den CASE TOOLS werden die Ergebnisse gestaltet, automatisch geprüft und verwaltet.

Die CASE TOOLS

- bieten hohen Bedienungskomfort durch
  - hochauflösende Bildschirmgrafik
  - Mouse-Eingabe
  - kurze Antwortzeiten
- erfordern keine Schulung
- fügen sich ohne Installationsaufwand in die bestehende DV-Infrastruktur



Die CASE METHODEN

- arbeiten organisations-integriert bei der Systemanalyse
- werden von Fachabteilungen verstanden und akzeptiert
- sind einfach und schnell zu erlernen
- machen Projekte strukturier- und planbar

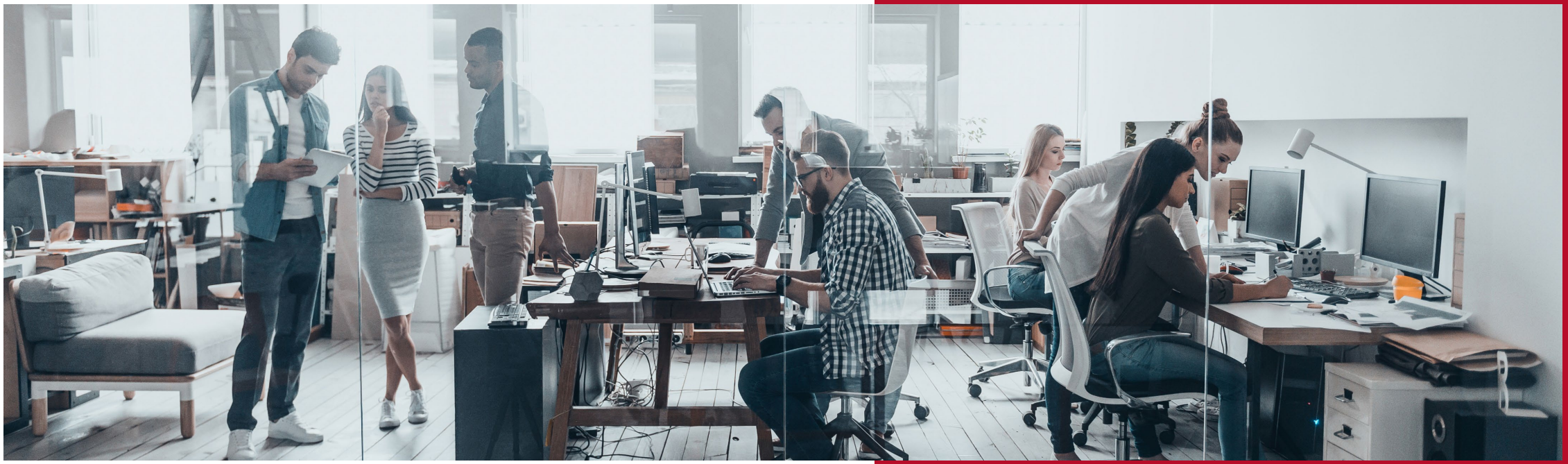
CASE
Computer Aided Software Engineering
Berliner Innovations- und Gründerzentrum

Pain Points

Viele Beteiligte

Viele Anforderungen,
User Stories, Testfälle …

Viele Arten, dasselbe zu sagen



Projekt

Start 2020

KI-Funktionen für PM/RE-Tool unter zwei Aspekten:

mehr Qualität der entwickelten Anforderungen

einfacheres Arbeiten durch neue User Experience

Drei Use Cases – der Erste zum Warm up

Sprach-
steuerung

PM/RE-Tool per Spracheingabe bedienen:

„Zeige die Liste aller Anforderungen in meinem Projekt“

„Öffne das Kanban-Board“

„Lege Anforderung … an“

Drei Use Cases – der Zweite

Verständlich-
keit

Understanding – Verständlichkeit von Anforderungen bezogen auf syntaktische Eigenschaften ermitteln:

zu lang

zu kurz

zu verschachtelt

unvollständig (fehlende Satzteile)

Drei Use Cases – der Dritte

Ähnlichkeit

Similarity – Semantische Ähnlichkeit von Anforderungen in großen Anforderungsmengen erkennen

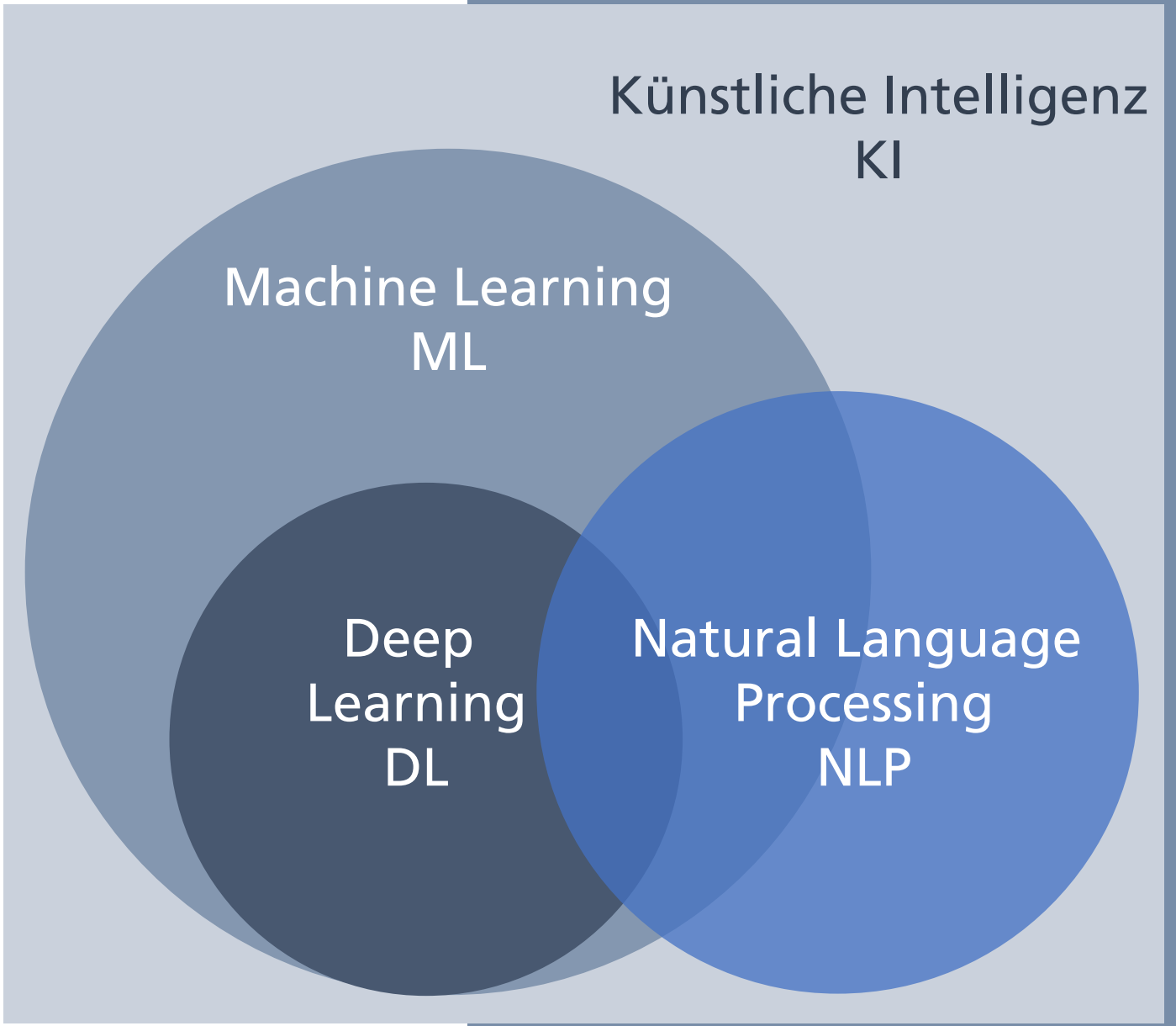
Einordnung

Maschine Learning

Algorithmen, die anhand sehr vieler Beispiele lernen, Aufgaben automatisch auszuführen, ohne dass dazu explizite Befehle nötig sind

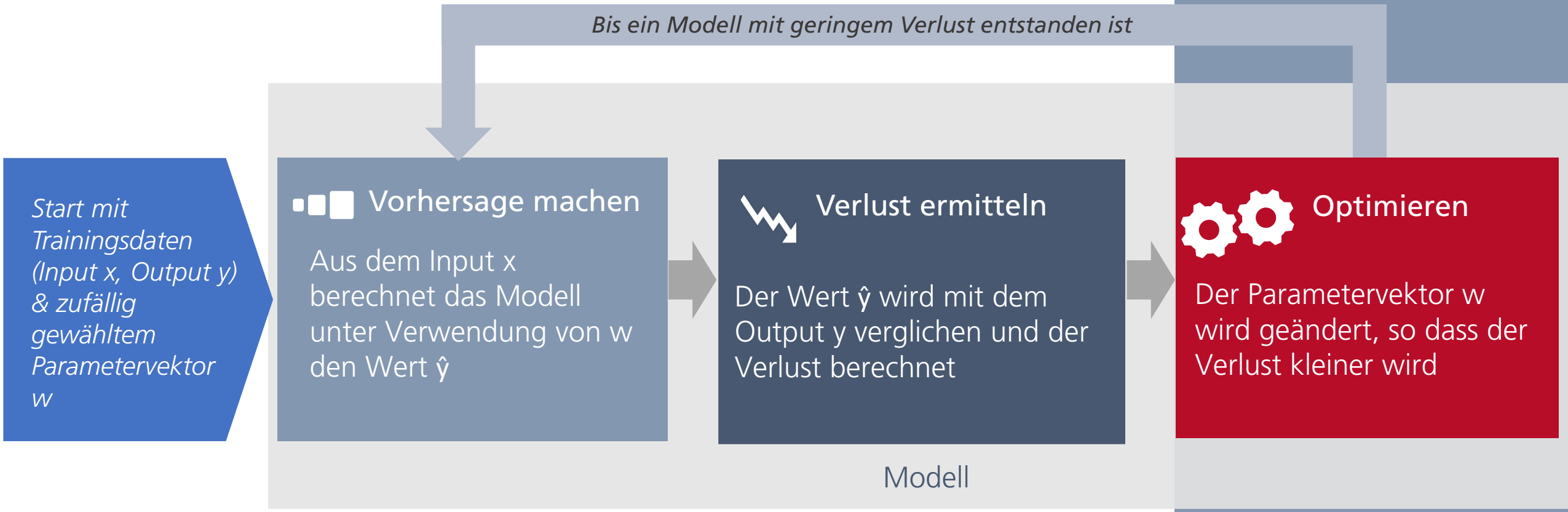
Deep Learning

Maschinelles Lernen, das auf einer Schichtenarchitektur aus künstlichen neuronalen Netzen basiert.



Überwachtes Lernen

Anhand von zahlreichen Beispielen in Form von Input-Output-Paaren wird vorgegeben, was erlernt werden soll



3 Hürden der KI-Entwicklung

- Die erforderliche große Menge an Trainingsdaten erstellen – in Deutsch
- Das passende Modell entwickeln
- Die notwendige Rechnerkapazität für das Training verfügbar machen

Ein Sprachassistent für objectiF RPM

Sprach-
steuerung

Schritt 1: Sprache in Text umwandeln

Automatic Speech Recognition oder
Speech-to-Text

Schritt 2: Formulierte Absicht des Users erkennen

Intent Recognition

Schritt 3: Ähnlichkeit der vom User genannten Elemente mit vorhandenen ermitteln

Semantic Similarity

Sprachassistent

Schritt 1: Speech-to-Text

Speech-to-Text Service

Microsoft
Azure Cognitive Services

Zeige	die Abfrage	*AlleAnforderungen*	im Package	*Anforderungsanalyse*

microTOOL GmbH Berlin. 2022

Sprachassistent

Schritt 2: Intent Recognition – die Herausforderung

Speech-to-Text Service

Zeige die Abfrage *AlleAnforderungen* im Package *Anforderungsanalyse*

- Zeige: Vorgegebener Intent Typ
- die Abfrage: Toolspezifischer Elementtyp
- *AlleAnforderungen*: Anwender-spezifischer Element-name
- im Package: Toolspezifischer Kontext-elementtyp
- *Anforderungsanalyse*: Anwender-spezifischer Kontext-elementname

Sprachassistent

Intent Recognition

Zeige	die Abfrage	AlleAnforderungen	im Package	Anforderungsanalyse

BERT Encoder von Google

BERT
von Google

Bidirectional Encoder Representations from Transformers

Open-Source-ML-Framework für die Verarbeitung natürlicher Sprache – also NLP

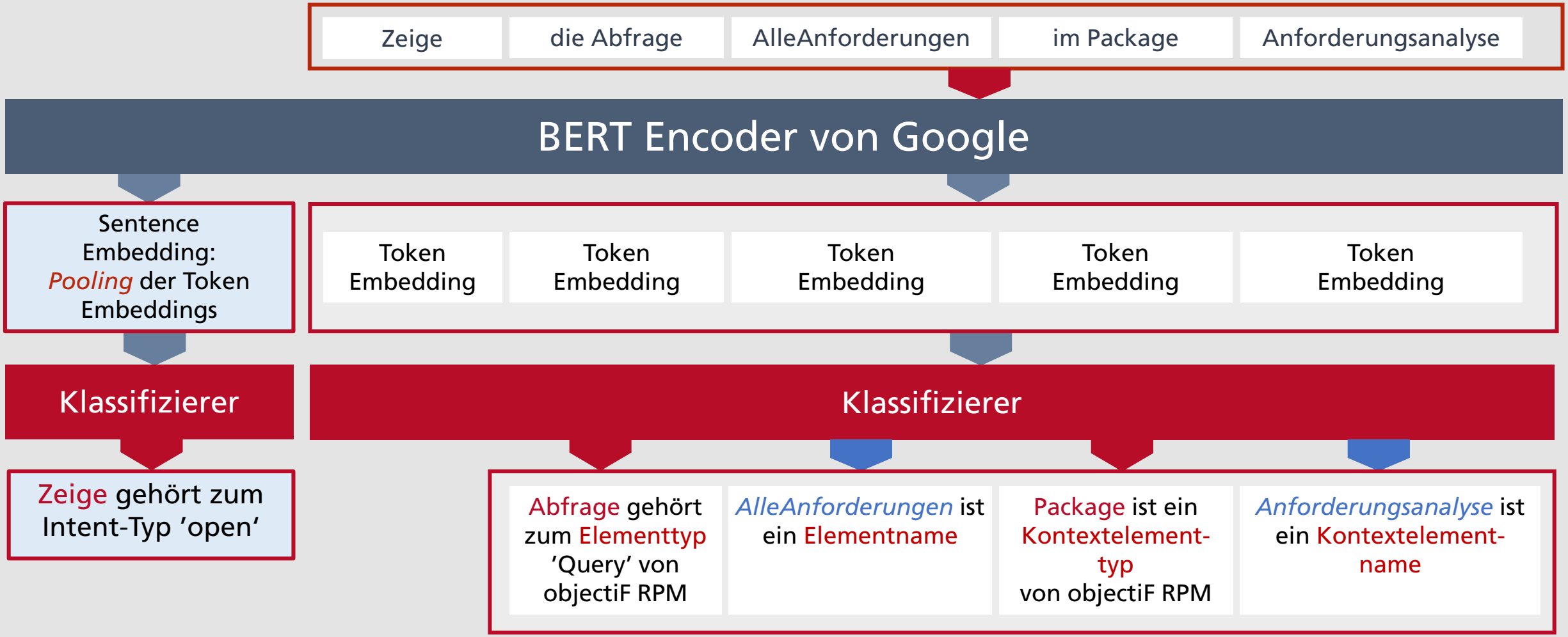
Vortrainiert mit Text aus Wikipedia

Liefert für eingegebene Wörter je eine digitale Darstellung als Vektor der Größe 768 – Token Embeddings

microTOOL GmbH Berlin. 2022

Sprachassistent

Intent Recognition



Sprachassistent

Schritt 3: Semantic Similarity

Vom User verwendeter Elementname

AlleAnforderungen

Vorhandene Elementnamen im Repository des Tools

Anforderungsliste kompakt

Anforderungen mit abgeleiteten Anforderungen

MeineAnforderungen

Semantic Similarity

Paarweiser Vergleich

Maß für die Ähnlichkeit

Ein Blick in die integrierte Lösung

Der Sprachassistent von objectiF RPM

Auf YouTube ansehen

https://www.youtube.com/watch?v=QGO1ovJEDHM

microTOOL GmbH Berlin. 2022

Was ist Verständlichkeit?

„In diesem Sinne dürften schon sehr bald die […] nach wie vor vorhandenen pazifistischen Strömungen als diskursives Korrektiv gegen die Eindimensionalität militärischer Stärke funktionieren.“

Der Tagesspiegel vom 3. April 2022

Was wir erreichen wollen

Verständlich-
keit

Klassifizieren einer Anforderung bzw. aller Anforderungen eines Backlogs aus syntaktischer Sicht als:

- unvollständig
- einfach
- verständlich
- ausführlich
- komplex

Das heißt …

Das System wird auf diese 5 Klassen trainiert

Die 5 Klassen definieren unsere *Grundwahrheit*

Die Trainingssätze werden danach markiert,

zu welcher der 5 Klassen sie gehören

microTOOL GmbH Berlin. 2022

Definition der fünf Klassen

Beispiele

Unvollständig

Subjekt, Prädikat oder Objekt fehlt

Information speichern

Einfach

Aktivsatz

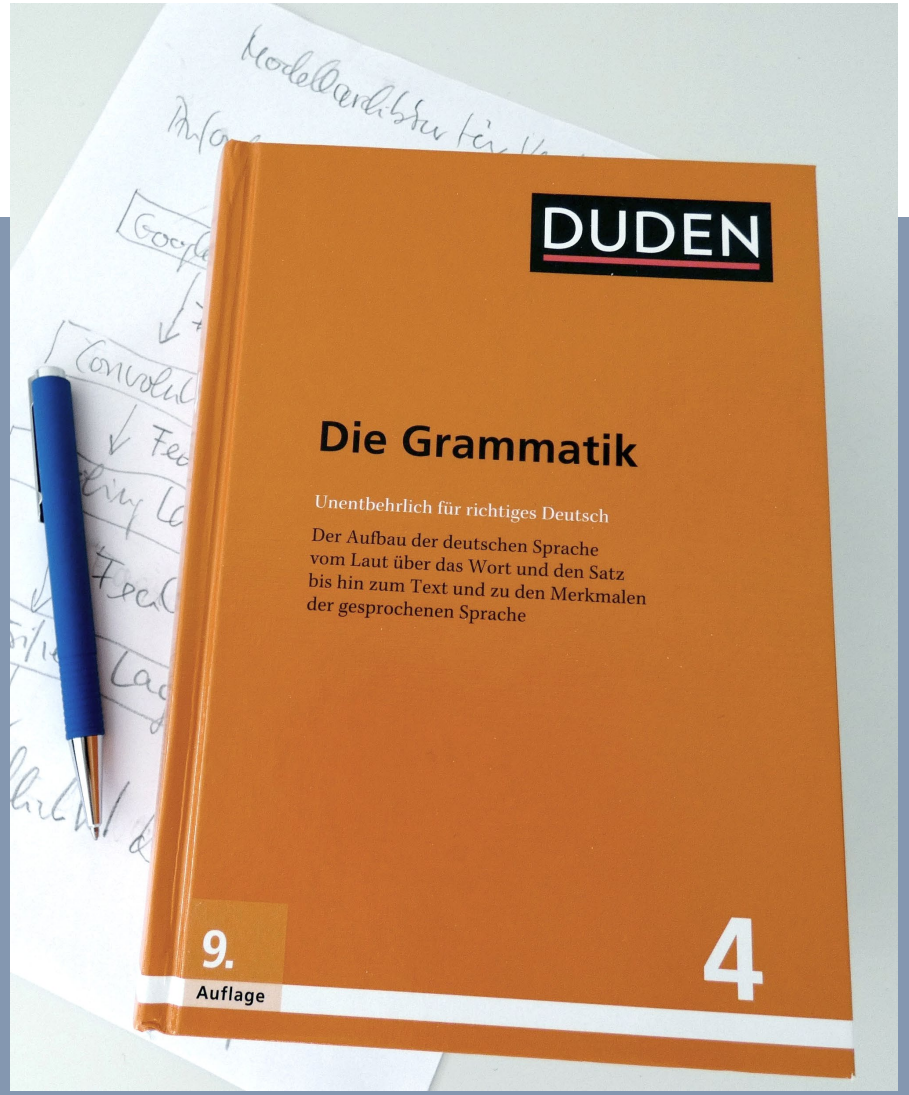
Die Anwendung speichert die Information

Aktivsatz mit Modalverb

Die Anwendung muss die Information speichern

Passivsatz für Vorgang mit Modalverb:

Die Information muss von der Anwendung gespeichert werden



microTOOL GmbH Berlin. 2022

Definition der fünf Klassen

Beispiele

Verständlich: mit einer Ergänzung

Adverbiale Bestimmung

Die Anwendung speichert die Information auf dem Server

Bedingung im Aktiv

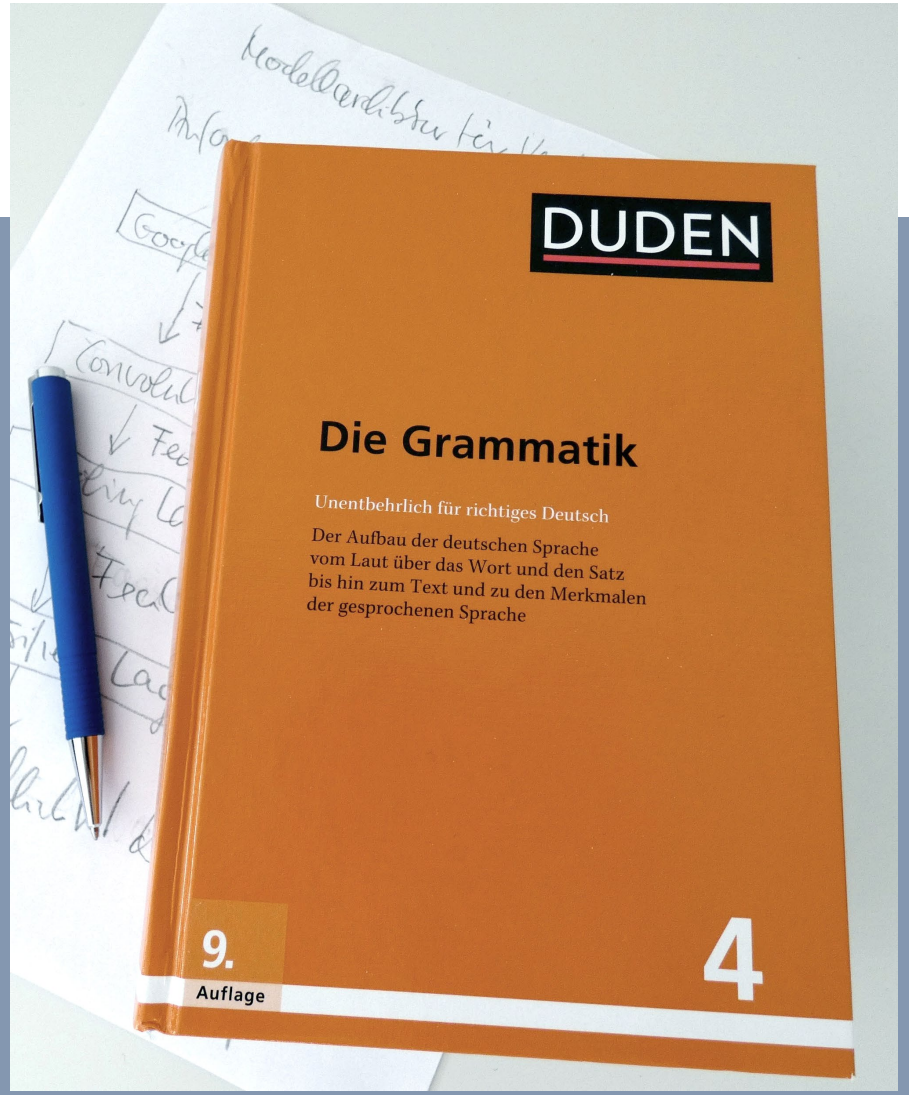
Die Anwendung speichert die Information, wenn das Ereignis A eintritt

Ausführlich: mit zwei Ergänzungen

Die Anwendung muss die Information auf dem Server speichern, wenn das Ereignis A eingetreten ist

Komplex

Die Anwendung soll die Information auf dem Server speichern, wenn die Berechtigung vorliegt und das Ereignis A eintritt



microTOOL GmbH Berlin. 2022

Modellarchitektur

Verständlichkeit

Die	Anwendung	muss	die	Information	speichern

Multi-lingualer T5 Encoder von Google

T5

von Google

Text-To-Text Transfer Transformer

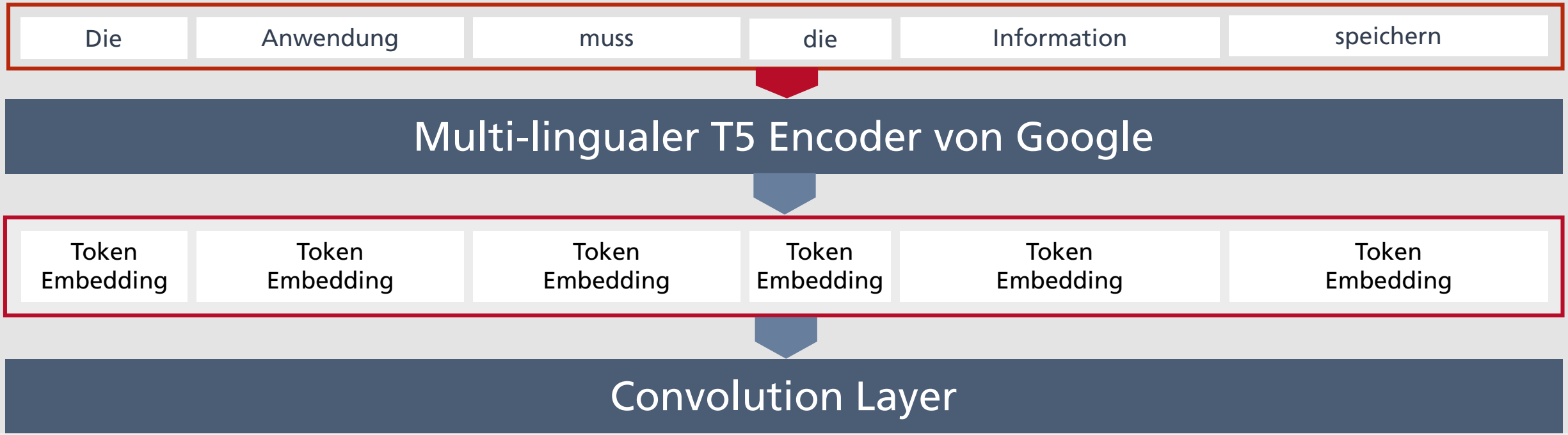
Open-Source-ML-Framework für die Verarbeitung natürlicher Sprache – also NLP

Vortrainiert mit dem Colossal Clean Crawled Corpus (C4)

Liefert für eingegebene Wörter jeweils eine digitale Darstellung als Vektor der Größe 1024

Modellarchitektur

Verständlichkeit



Convolution Layer

Stammt aus der Bildverarbeitung und ist trainierbar

Open-Source aus der ML-Bibliothek TensorFlow/Keras

Dient hier zum Filtern nach Satzglieder/Wortarten

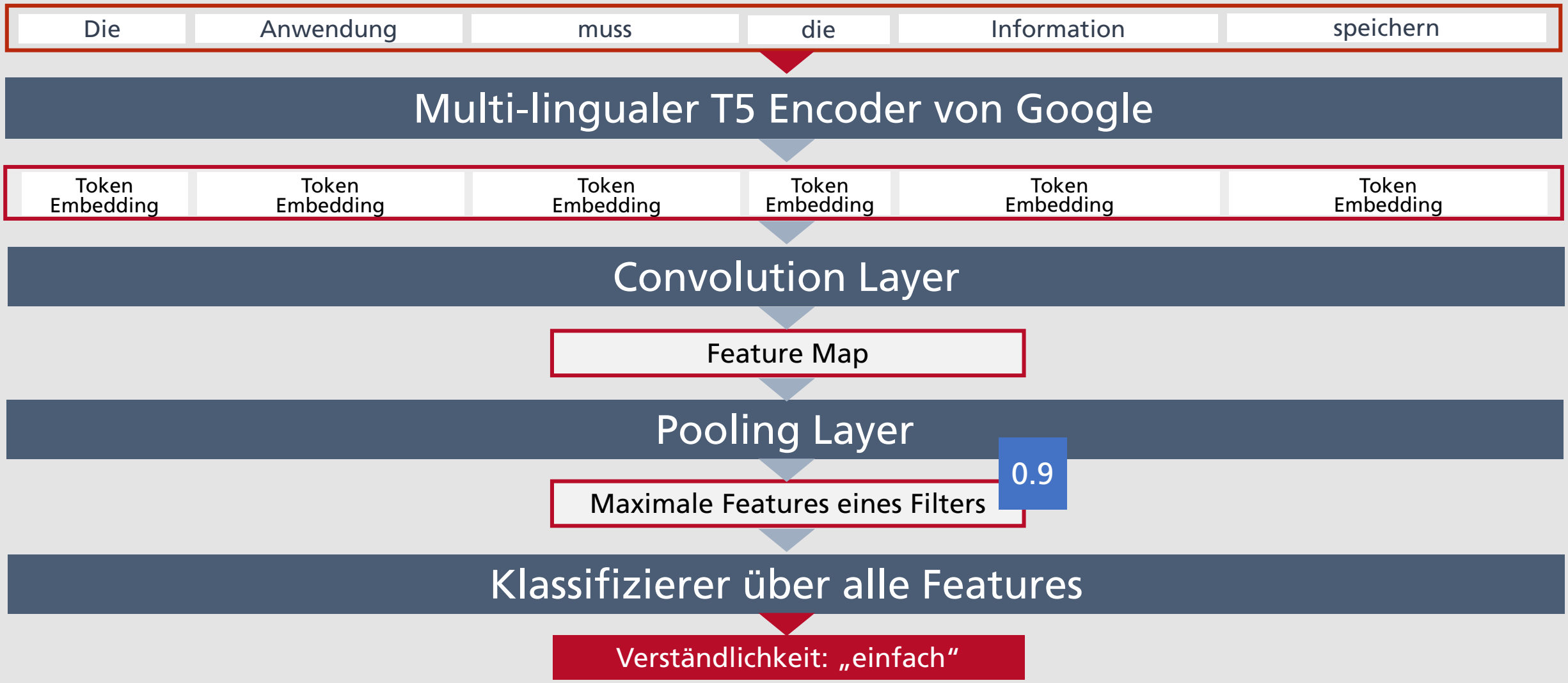
Modellarchitektur

Verständlichkeit

Features – liefern ein dem Ziel des Filters entsprechendes Maß

Modellarchitektur

Verständlichkeit



Was wir erreichen wollen

Ähnlichkeit

Vorhersage der semantischen Ähnlichkeit von Sätzen auf einer Skala

sehr ähnlich — überhaupt nicht ähnlich

1 0 -1

Anwendbar auf Anforderungen, Test Cases und Bugs

Beispiele für Ähnlichkeit

Wenn für Satzglieder Synonyme verwendet werden …

sehr ähnlich

Die Anwendung muss die Information speichern

Die Software muss die Information speichern

etwas ähnlich

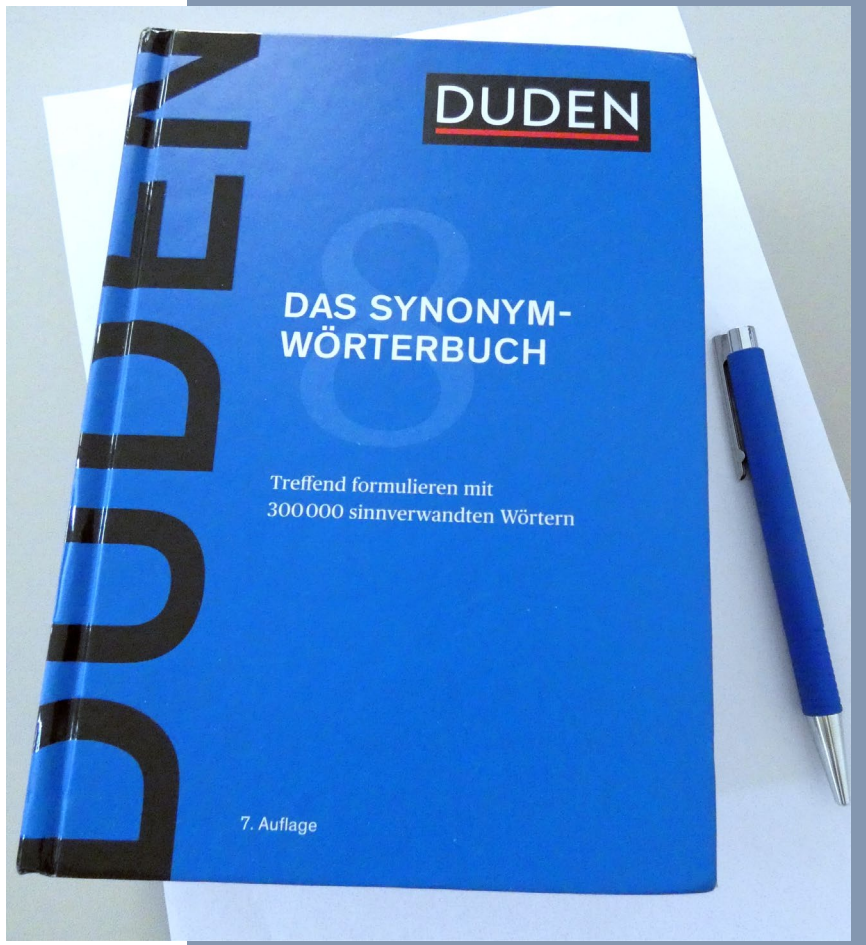
Die Anwendung muss die Information speichern

Die Anwendung muss die Dokumente ablegen

kaum ähnlich

Die Anwendung muss die Information speichern

Das Team muss das Release bereitstellen



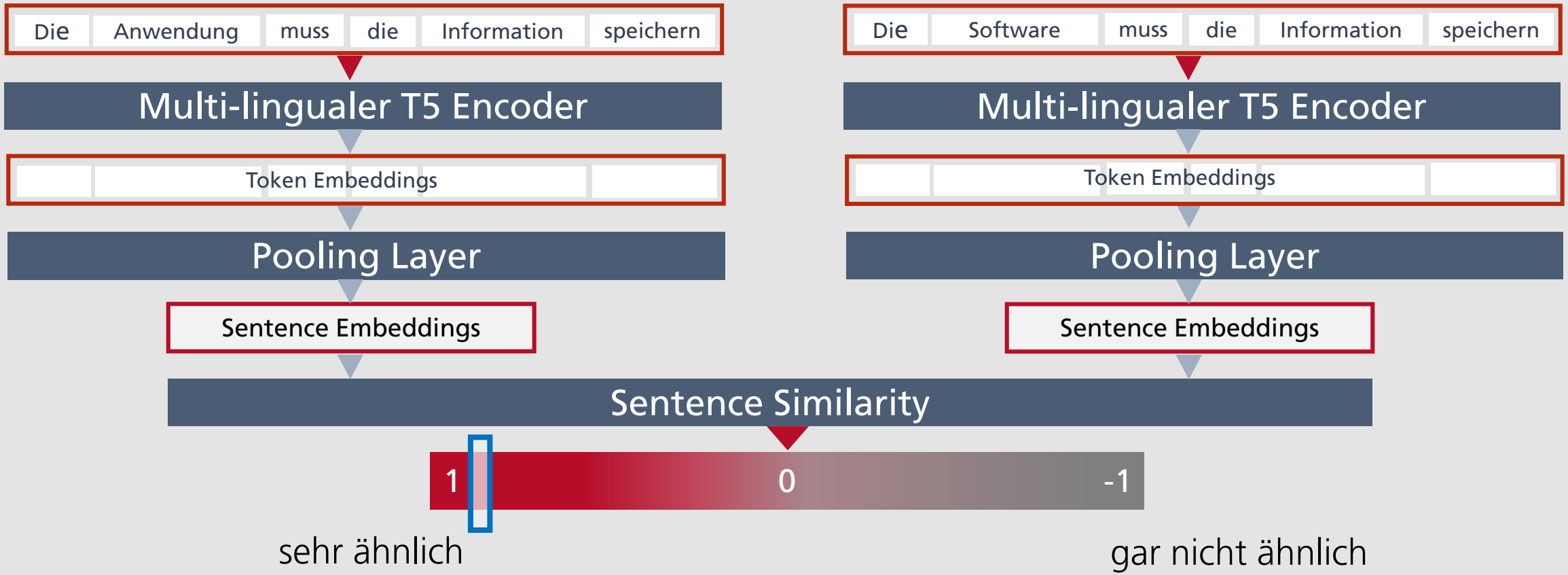
Das heißt …

Das System wird auf das Erkennen von semantischer Ähnlichkeit trainiert

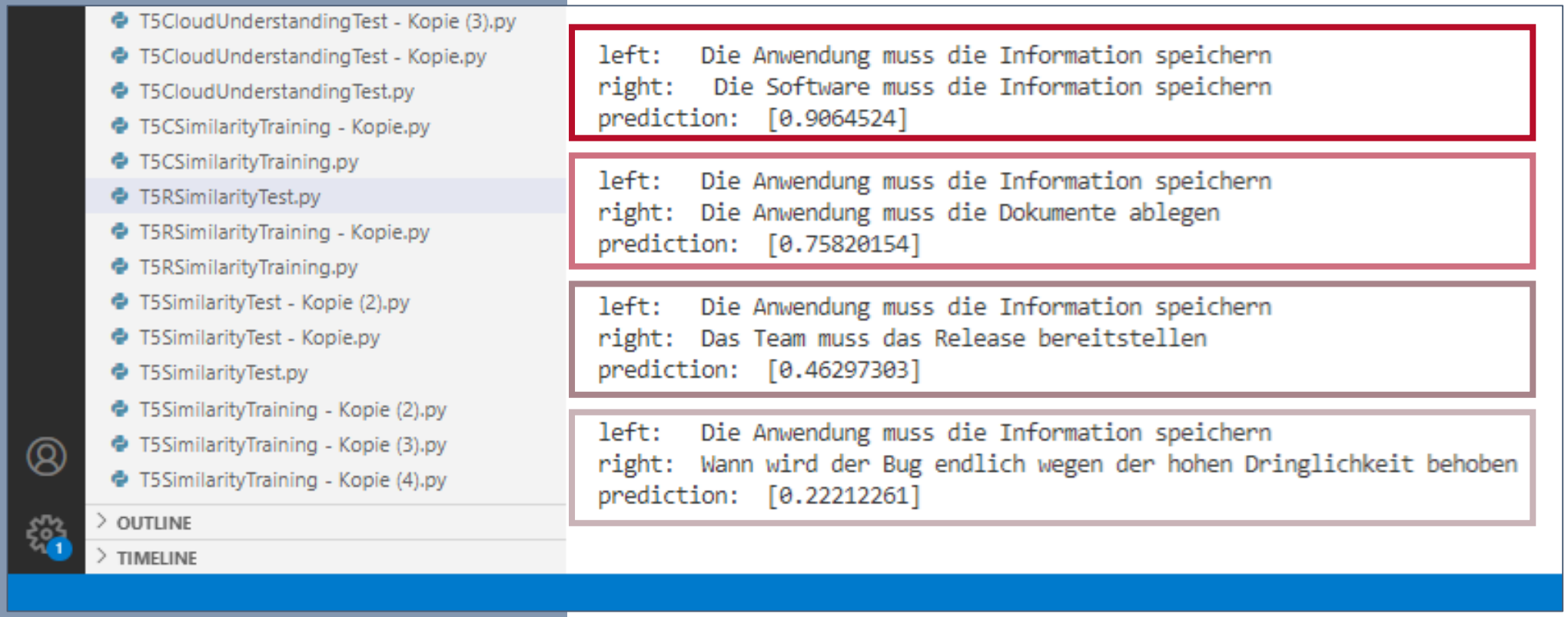
mit den verfügbaren Trainingssätzen des Semantic Textual Simularity Benchmark (STSB) Dataset

Modellarchitektur

Siamesisches Modell für Ähnlichkeit



Die „echte“ Vorhersage für die Beispiele



3 Hürden genommen

Die erforderliche große Menge an Trainingsdaten erstellen – in Deutsch

- Vorhandene Trainings-Datasets für Fine-tuning nutzen

Das passende Modell entwickeln

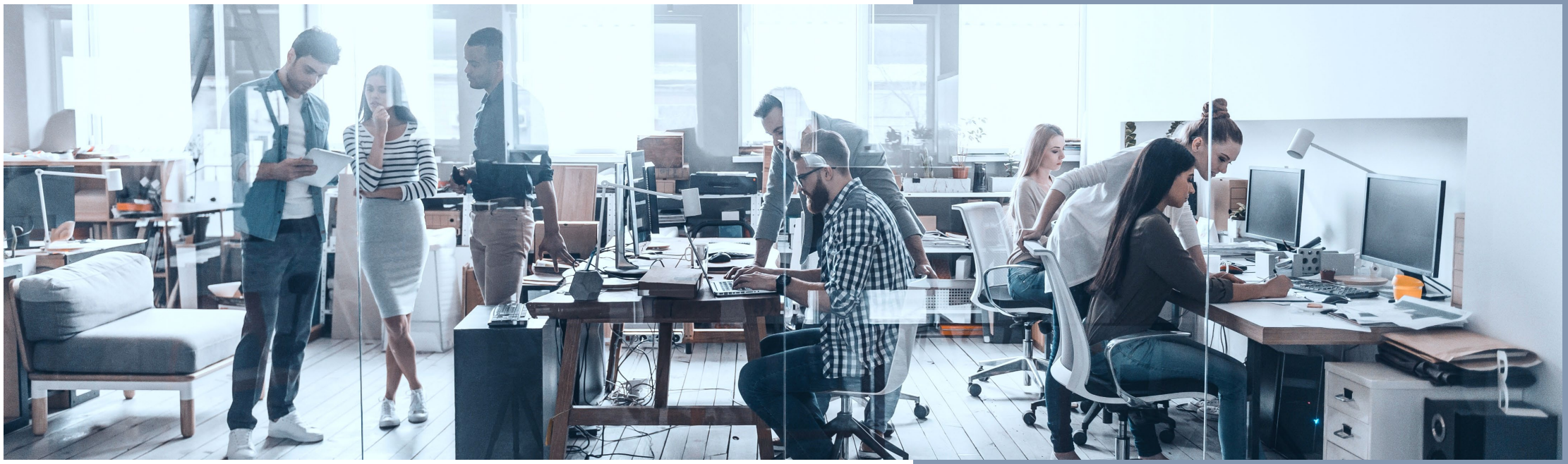
- Vortrainierte Modelle einsetzen

Die notwendige Rechnerkapazität für das Training verfügbar machen

- Training in die Cloud verlagern

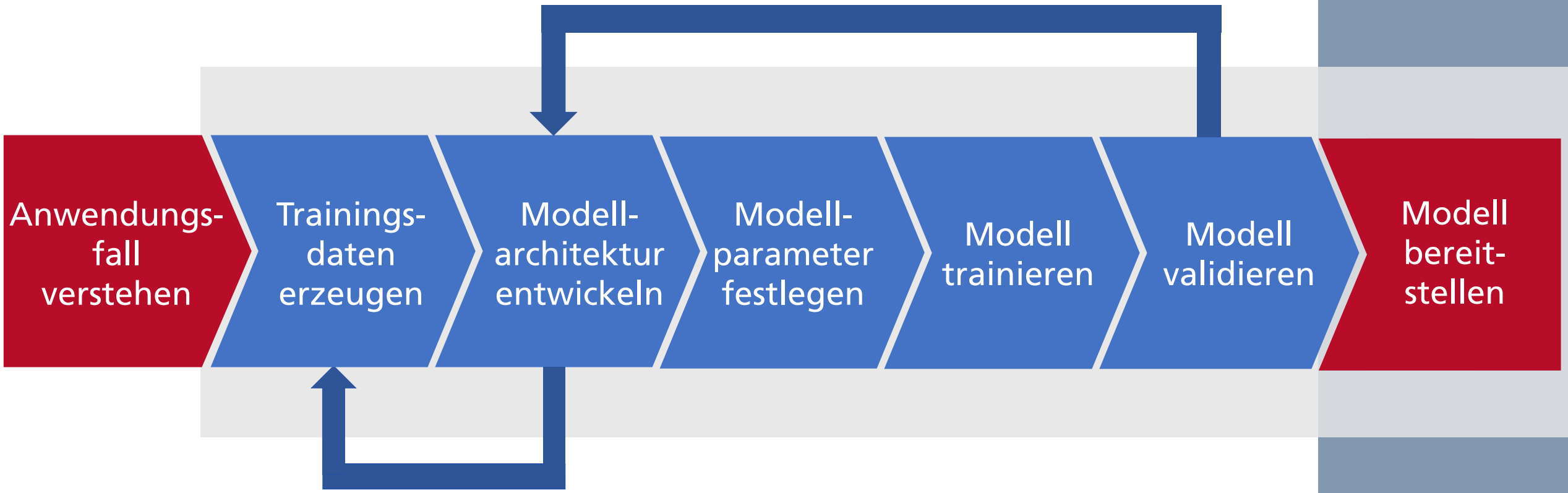
Die eigentliche Hürde

Die Köpfe von Softwareentwicklern/ Softwareentwicklerinnen in Richtung KI bewegen



Der KI-Entwicklungsprozess

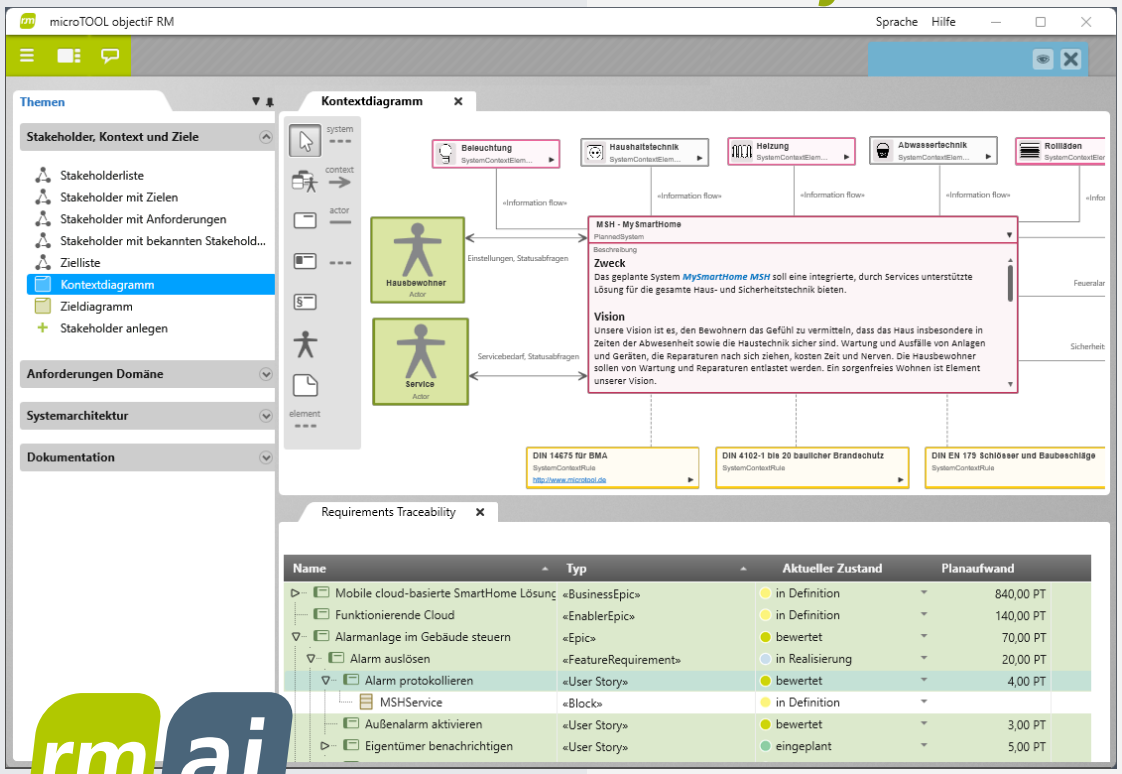
Ähnlich – und doch ganz anders als der SW-Entwicklungsprozess



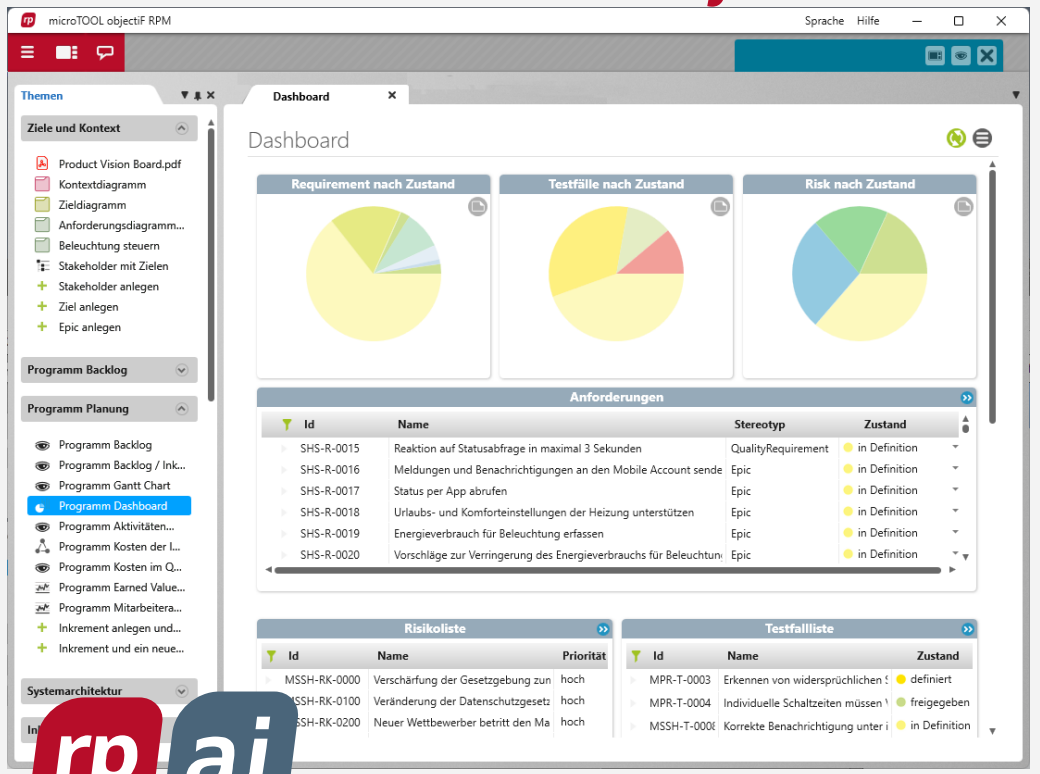
objectiFAI

Ergänzende Produktkomponente

objectiF®RM



objectiFRPM®



Blick in die Zukunft

Entwicklung generativer Tasks mit T5

- Question Answering im Kontext des Tool Repository
- Abstractive Summarization von RE-Dokumenten

„ML Systeme in der Praxis sind oft komplizierte Pipelines aus mehreren Algorithmen und Diensten.

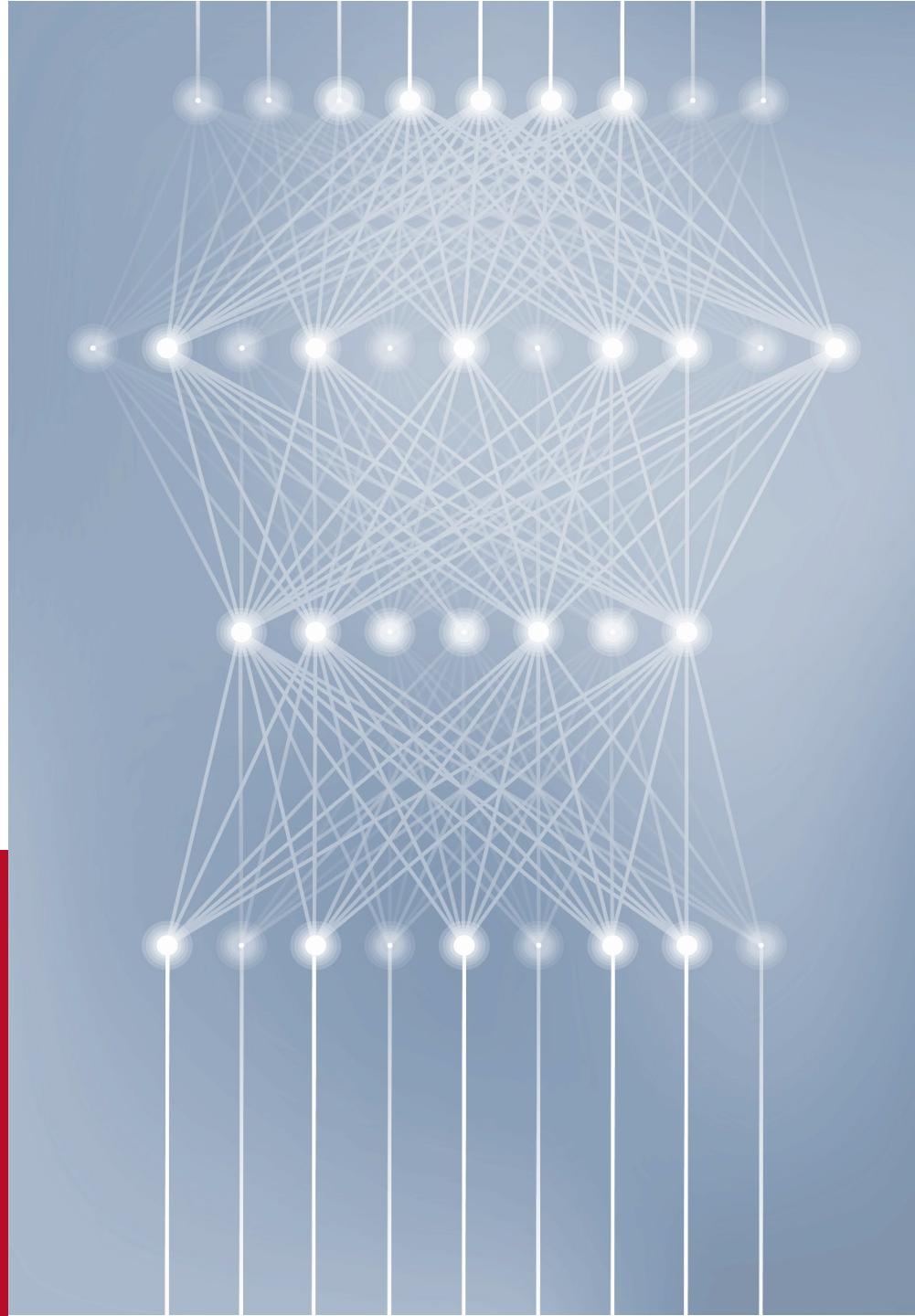
Noch schwieriger wird die Sache dadurch, dass die ML-Entwicklung oft mehr eine Kunst als eine Wissenschaft ist.

Sie erfordert viel Erfahrung, gezieltes Raten und schlicht und einfach Glück.“

Paolo Perrotta

Aus: Machine Learning für Software Entwickler
Von der Python-Codezeile zur Deep-Learning Anwendung

Danke

www.microtool.de

microTOOL GmbH Berlin. 2022