



# Anwendung Generativer KI

---

2025

# Lizenz

---

- Die Kursmaterialien wurden von Ralf Bendig erstellt, sofern nicht anders angegeben.
- Lizenz CC BY 4.0.

Titelseite: Bild mit MS Copilot erstellt

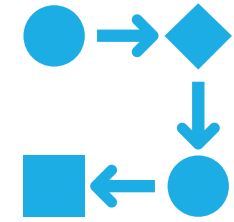
# 00 Intro

---



# Kurs-Organisation

---



## ZEITPLANUNG

- 5 Tage
- Start: 09:00 Uhr
- Ende: 16:30 Uhr
- Pause nach 90 Min

## VORGEHEN

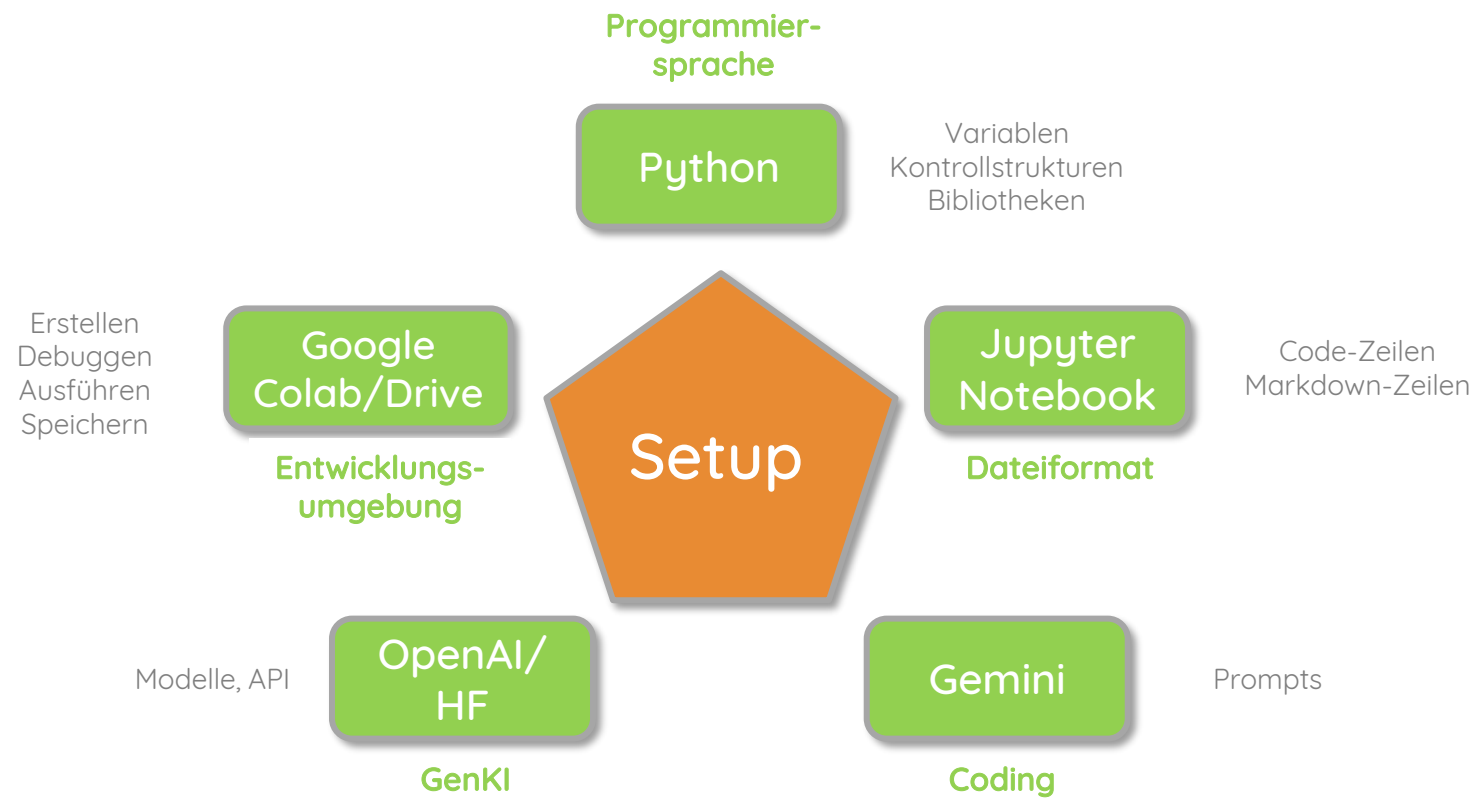
- Grundlagen/Basiswissen
- Beispiele
- Training/Fallstudien

## VERSCHIEDENES

- Pinboard

# IT-Setup

---



# Pinboard

<https://bit.ly/3T2evF6>



Bild mit Recraft erstellt

# Google Colab(oratory)

---



- Google Colaboratory, kurz Colab, ist eine kostenlose Entwicklungsumgebung, die vollständig in der Cloud ausgeführt wird.
- In Colab können Jupyter-Notebooks erstellt, bearbeiten und ausgeführt werden.
- Colab unterstützt viele beliebte Machine-Learning-Bibliotheken, die einfach in ein Notebook geladen werden können.
- Colab erlaubt es unterschiedliche Laufzeitumgebungen zu definieren in denen man neben einer CPU auch GPUs und TPUs verwenden kann.
- Colab hat mit Gemini eine integriert GenKI für Coding.

CPU = Central Processing Unit, GPU = Graphics Processing Unit, TPU = Tensor Processing Unit



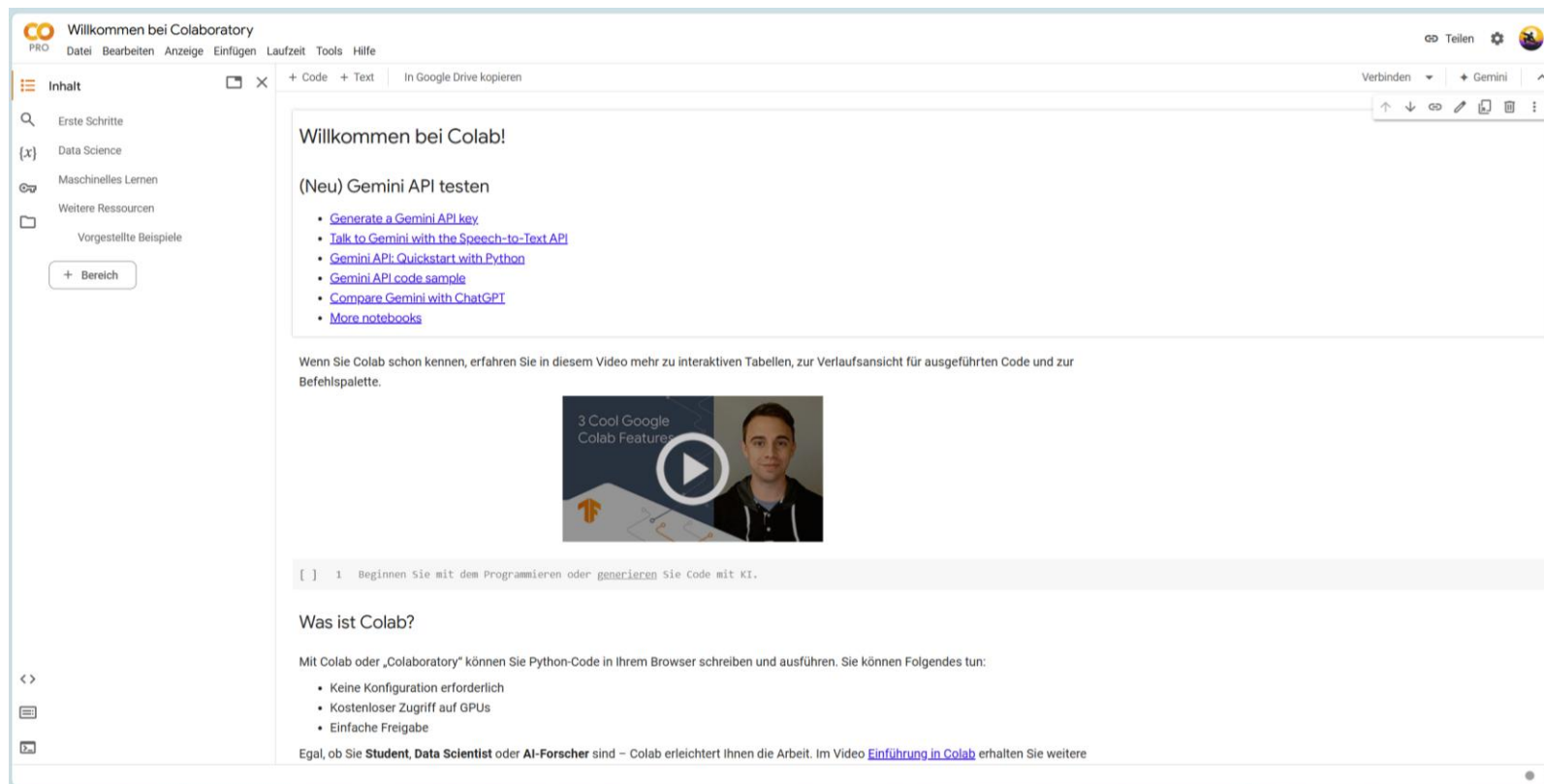
# Jupyter-Notebook

---

- Die Jupyter App ist eine Entwicklungsumgebung, die das Bearbeiten und Ausführen von Programmiersprachen, u.a. Python, über einen Webbrowser ermöglicht.
- Der Name Jupyter bezieht sich auf die drei wesentlichen Programmiersprachen Julia, Python und R und ist auch eine Hommage an Galileos Notizbucheinträge zur Entdeckung der Jupitermonde.
- Mit der Jupyter App kann man Notizbücher erstellen. Diese Notizbücher (Dateiendung .ipynb) enthalten:
  - **Programmcode**, der ausgeführt werden kann,
  - **Markdown Zeilen**, das sind Textzeilen mit Formatierungsangaben.
- Die Jupyter-Notebooks werden v.a. für interaktive, wissenschaftliche Analysen und Berechnungen, z.B. Data Analytics und Machine Learning, verwendet.



# Google Colab - Einstieg



# Gemini in Colab

---



## ■ Gemini Zeile

1 Beginnen Sie mit dem Programmieren oder generieren Sie Code mit KI.

- Code generieren direkt im Jupyter Notebook
- Code-Vervollständigung

## ■ Gemini Register

★ Gemini

- Code optimieren
- Code kommentieren/dokumentieren
- Code erklären/erläutern
- allgemeine Prompts

# Tastatureinstellungen/Shortcuts

## Tastatureinstellungen

Editor-Tastaturbelegungen	
default	
<input checked="" type="checkbox"/> Mit der Eingabetaste werden Vorschläge akzeptiert	
Tastenkombinationen	
Wenn Sie eine Tastenkombination hinzufügen oder ändern möchten, klicken Sie auf die entsprechende Tastenkombination und geben Sie dann die neue Kombination ein. Beachten Sie, dass Ctrl+M als Präfix für Tastenkombinationen mit mehreren Tasten verwendet werden kann.	
Tastenkombination	.ipynb herunterladen
Tastenkombination	.py herunterladen
Ctrl+/	Aktuelle Zeile kommentieren
Tastenkombination	Alle Ausgaben löschen
Tastenkombination	Alle Laufzeiten auf Werkseinstellungen zurücksetzen
Ctrl+Shift+A	Alle Zellen auswählen
Ctrl+F9	Alle Zellen in Notebook ausführen
Ctrl+H	Alle suchen/ersetzen
Ctrl+I	Alle/ausgewählte Abschnitte maximieren
Ctrl+J	Alle/ausgewählte Abschnitte minimieren
Ctrl+M -	An Cursorposition teilen
Tastenkombination	Ansicht mit einem Tab ansehen
Tastenkombination	Auf die zuletzt ausgeführte Zelle fokussieren
Ctrl+M I	Ausführung unterbrechen
Ctrl+M O	Ausgabe einblenden/ausblenden
Tastenkombination	Ausgabevollbild anzeigen
Ctrl+M	Laufzeit neu starten
Tastenkombination	Laufzeit neu starten und alle Zellen im Notebook ausführen
Tastenkombination	Laufzeitprotokolle ansehen
Tastenkombination	Layout des minimierten Abschnitts speichern
Ctrl+M Z oder Ctrl+Shift+Z	Letzte Zellaktion rückgängig machen
Tastenkombination	Linken Bereich in einen Tab verschieben
Ctrl+^	Maximieren/Minimieren des aktuellen Abschnitts aktivieren/deaktivieren
Shift+Click	Mehrere Zellen auswählen
Tastenkombination	Mit einer Laufzeit verbinden
Tastenkombination	Mit einer benutzerdefinierten GCE-VM verbinden
Tastenkombination	Mit lokaler Laufzeit verbinden
Tastenkombination	Neues Notebook erstellen
Ctrl+P	Notebook drucken

## Tastatureinstellungen

Tastenkombination	Ausgewählte Ausgaben löschen
Ctrl+M K	Ausgewählte Zellen nach oben verschieben
Ctrl+M J	Ausgewählte Zellen nach unten verschieben
Ctrl+F10	Ausgewählte Zellen und alle folgenden Zellen ausführen
Tastenkombination	Ausgewählte Zellen zusammenführen
Ctrl+Shift+Enter	Auswahl ausführen
Shift+Down	Auswahl erweitern, um nächste Zelle einzuschließen
Shift+Up	Auswahl erweitern, um vorherige Zelle einzuschließen
Tastenkombination	Automatische Ausführung der ersten Zelle oder des ersten Abschnitts bei jeder Ausführung aktivieren
Ctrl+Space oder Tab	Automatische Vervollständigung
Ctrl+Shift+P	Befehlspalette anzeigen
Tastenkombination	Code ein-/ausblenden
Ctrl+Alt+P	Code-Snippets-Bereich anzeigen
Tab	Code-docstring-Hilfe aktivieren/deaktivieren
Tastenkombination	Codezelle hinzufügen
Ctrl+M A	Codezelle oben einfügen
Ctrl+M B	Codezelle unten einfügen
Tastenkombination	Colab Enterprise öffnen
Tastenkombination	Notebook freigeben
Tastenkombination	Notebook hochladen
Tastenkombination	Notebook in Drive verschieben
Tastenkombination	Notebook in Google Drive markieren/Markierung aufheben
Tastenkombination	Notebook in Papierkorb verschieben
Ctrl+S	Notebook speichern
Ctrl+O	Notebook öffnen
Tastenkombination	Notebook-Einstellungen öffnen
Tastenkombination	Notebook-Quelle anzeigen
Tastenkombination	Notebook-Vergleich
Ctrl+M N	Nächste Zelle
Ctrl+Shift+]	Nächsten Tab hervorheben
Ctrl+G	Nächstes Element suchen
Tastenkombination	Ressourcen ansehen
Ctrl+Alt+N	Scratchpad-Codezelle öffnen
Tastenkombination	Seitenleiste für Kommentare
Tastenkombination	Sichtbarkeit des Headers aktivieren/deaktivieren
Tastenkombination	Sitzungen verwalten
Tastenkombination	Statusleiste einblenden/ausblenden
Tastenkombination	Tab in den nächsten Bereich verschieben
Tastenkombination	Tab in den vorherigen Bereich verschieben
Tastenkombination	Tabs in zwei Spalten anzeigen

## Tastatureinstellungen

Tastenkombination	Dateibrowser anzeigen
Tastenkombination	Drive bereitstellen
Tastenkombination	Drive trennen
Tastenkombination	Editor-Einstellungen öffnen
Tastenkombination	Einstellungen öffnen
Shift+Tab	Einzug für aktuelle Zeile entfernen
Tastenkombination	Fokus mit Tabulatortaste verschieben
Tastenkombination	Formular hinzufügen
Ctrl+M F	Formularansicht ändern
Tastenkombination	Formularattribute bearbeiten
Tastenkombination	Formularfeld hinzufügen
Tastenkombination	Geplante Notebooks anzeigen
Ctrl+Enter	Hervorgehobene Zelle ausführen
Ctrl+Shift+S	Hervorgehobene Zelle auswählen
Tastenkombination	Hervorgehobene Zelle mit nächster Zelle zusammenführen
Tastenkombination	Hervorgehobene Zelle mit vorheriger Zelle zusammenführen
Esc	Hervorheben der Zelle aufheben
Tastenkombination	Im Playground-Modus öffnen
Tastenkombination	In Google Drive suchen
Tastenkombination	In Scratchpad-Zelle kopieren
Shift+Ctrl+H	In aktueller Zelle alle ersetzen
Tastenkombination	Informationen zu Notebook-Datei anzeigen
Tastenkombination	Inhaltsverzeichnis anzeigen
Tastenkombination	Tabs in zwei Zeilen anzeigen
Ctrl+M H	Tastenkombinationen anzeigen
Tastenkombination	Terminal anzeigen
Tastenkombination	Textzelle hinzufügen
Tastenkombination	Variablenübersicht anzeigen
Tastenkombination	Verbindung mit gehosteter Laufzeit herstellen
Tastenkombination	Verbindung trennen und Laufzeit löschen
Tastenkombination	Verlauf der Codeausführung anzeigen
Ctrl+M P	Vorherige Zelle
Ctrl+Shift+[	Vorherigen Tab hervorheben
Ctrl+Shift+G	Vorheriges Element suchen
Tastenkombination	Wieder verbinden
Ctrl+M L	Zeilenummern aktivieren/deaktivieren
Ctrl+Shift+Y	Zellaktion wiederholen
Ctrl+Click	Zellauswahl aktivieren/deaktivieren
Alt+Enter	Zelle ausführen und neue Zelle einfügen
Shift+Enter	Zelle ausführen und nächste Zelle auswählen
Tastenkombination	Zelle in Tab spiegeln
Tastenkombination	Zelle mit Abschnittsüberschrift hinzufügen
Tastenkombination	Zelle oder Auswahl ausschneiden

## Tastatureinstellungen

Tastenkombination	JSON-Notebook ansehen
Ctrl+Alt+M	Kommentar hinzufügen
Tastenkombination	Kommentarbereich öffnen
Tastenkombination	Kopie als GitHub Gist speichern
Tastenkombination	Kopie in Drive speichern
Tastenkombination	Kopie in GitHub speichern
Tastenkombination	Zeile oder Auswahl kopieren
Ctrl+M D	Zelle/Auswahl löschen
Ctrl+F8	Zellen vor der aktuellen Zelle ausführen
Ctrl+M Y	Zu Codezelle konvertieren
Ctrl+M M	Zu Textzelle konvertieren
Tastenkombination	Zu einer konkreten Zelle
Ctrl+M S	Überarbeitung speichern und anpinnen
Tastenkombination	Überarbeitungsverlauf anzeigen

# Entwicklung KI-Modell

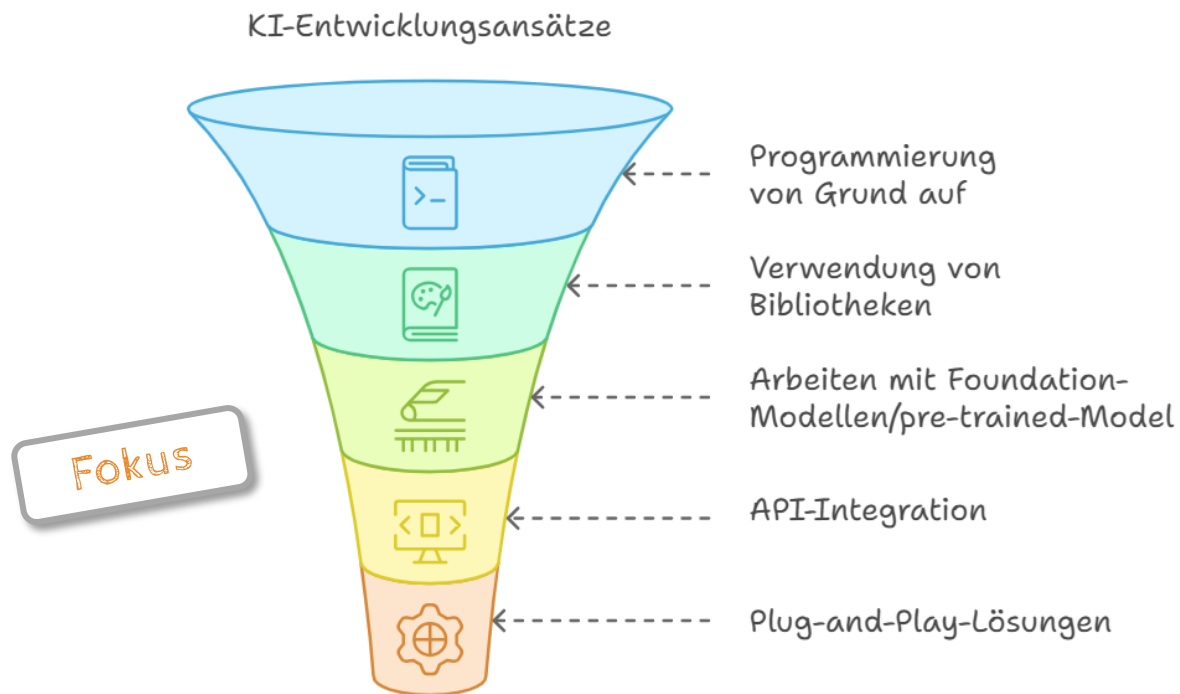


Bild mit napkin.ai erstellt

1. Von Grund auf programmieren
  - Eigenimplementierung aller Komponenten.
  - Beispiel: Entwicklung neuer Modellarchitekturen.
2. Verwendung von Bibliotheken
  - Einsatz von z.B. sklearn, TensorFlow, Keras.
  - Beispiel: Training eigener Modelle, Nutzung von GPT.
3. Arbeiten mit Foundation-Modellen
  - Schnelle Ergebnisse durch Prompting/RAG/Feintuning.
  - Beispiel: Domänenspezifische Modellanpassung.
4. Integration über API
  - Nutzung fertiger APIs, einfache Integration.
  - Beispiel: KI-Funktionen in Geschäftsanwendungen.
5. Plug-and-Play-Lösungen
  - Fertige Tools, sofort einsatzbereit.
  - Beispiel: No-Code KI-Anwendungen.

# 01 Einführung in Generative AI

---



# As Time Goes By

---

2005



2013

Quelle: Papst-Momente: Bilder zeigen Vergleich zwischen 2005 und 2013 - DER SPIEGEL

# Was ist Generative KI?

---

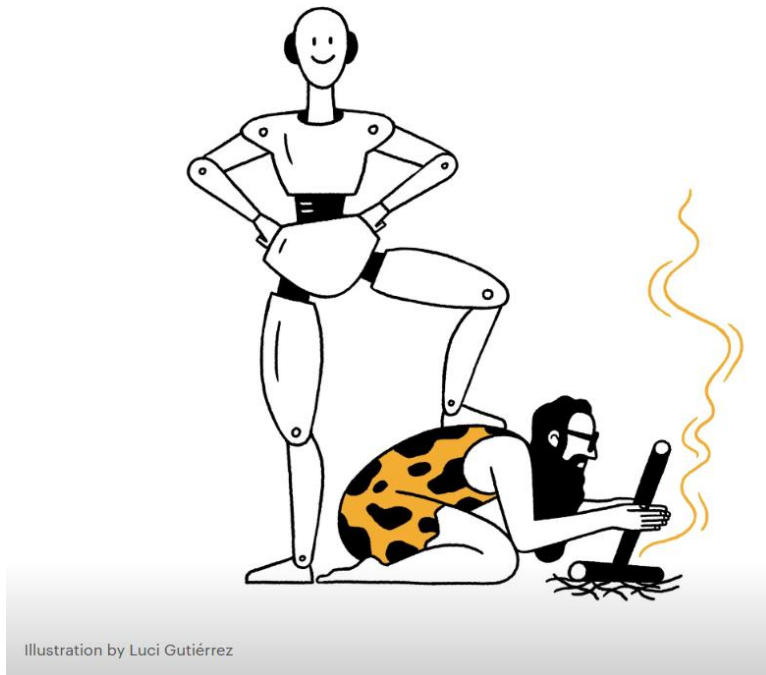


Illustration by Luci Gutiérrez

- Generative KI ist der Überbegriff für "**kreative** Künstliche Intelligenz", die bei Bedarf neue Inhalte bzw. Originalinhalte produzieren kann.
- Generative KI ist dazu in der Lage, nicht nur vorhandene Daten zu analysieren oder zu klassifizieren, sie erstellt etwas **völlig Neues**, egal ob Text, Bilder, Audio, synthetische Daten usw.

Quelle: [Upcoming Landmarks in Artificial Intelligence](#) | The New Yorker



# ChatGPT – einer von Vielen

---



- ChatGPT ist ein von OpenAI entwickeltes künstliches Intelligenzmodell.
- Er errechnet die **Wahrscheinlichkeit** der nächsten **Wortsequenz**, um menschenähnliche Inhalte zu imitieren.
- Die GPT-Modelle (Generative Pre-trained Transformer) wurde mit **Milliarden** von Wörtern trainiert & kann Texte generieren, Fragen beantworten und viele Aufgaben durchführen, die mit Sprache zu tun haben.
- Diese Modelle nutzen Deep Learning und speziell entworfene neuronale Netze, um **menschenähnliche Textantworten** zu produzieren.
- ChatGPT ist eine spezialisierte Version der GPT-Modelle, die speziell für die Erstellung von Dialogen in einem Chat-ähnlichen Kontext optimiert ist.
- Es handelt sich um eine Anwendung des GPT-Modells, das auf Konversationen ausgerichtet ist.



# GPT



Bild mit DALL·E erstellt

Wortsequenz-  
wahrscheinlichkeits-  
rechenmaschine

Version	Jahr	Anzahl Parameter
GPT-1	2018	117 Millionen
GPT-2	2019	1,5 Milliarden
GPT-3	2020	175 Milliarden
GPT-4	2023	100 Billionen (geschätzt)

# Trainingsdaten GPT-3

Dataset	# tokens	Proportion within training
<u>Common Crawl</u>	410 billion	60%
<u>WebText2</u>	19 billion	22%
Books1	12 billion	8%
Books2	55 billion	8%
Wikipedia	3 billion	3%

Quelle: GPT-3 - Wikiwand

### Statistics of Common Crawl Monthly Archives

Number of pages, distribution of top-level domains, crawl overlaps, etc. - basic metrics about Common Crawl Monthly Crawl Archives  
Latest crawl: CC-MAIN-2024-33

[Home](#)  
[Size of crawls](#)  
[Top-level domains](#)  
[Registered domains](#)  
[Crawler metrics](#)  
[Crawl overlaps](#)  
[Media types](#)  
[Character sets](#)  
[Languages](#)

[View the Project on GitHub](#)

This project is maintained by [commoncrawl](#)

Hosted on GitHub Pages — Theme by [orderedlist](#)

### Top-500 Registered Domains of the Latest Main Crawl

The table below shows the top 500 registered domains (in terms of page captures) of the last main/monthly crawl (CC-MAIN-2024-33). The underlying data is also provided in CSV format, see [domains-top-500.csv](#).

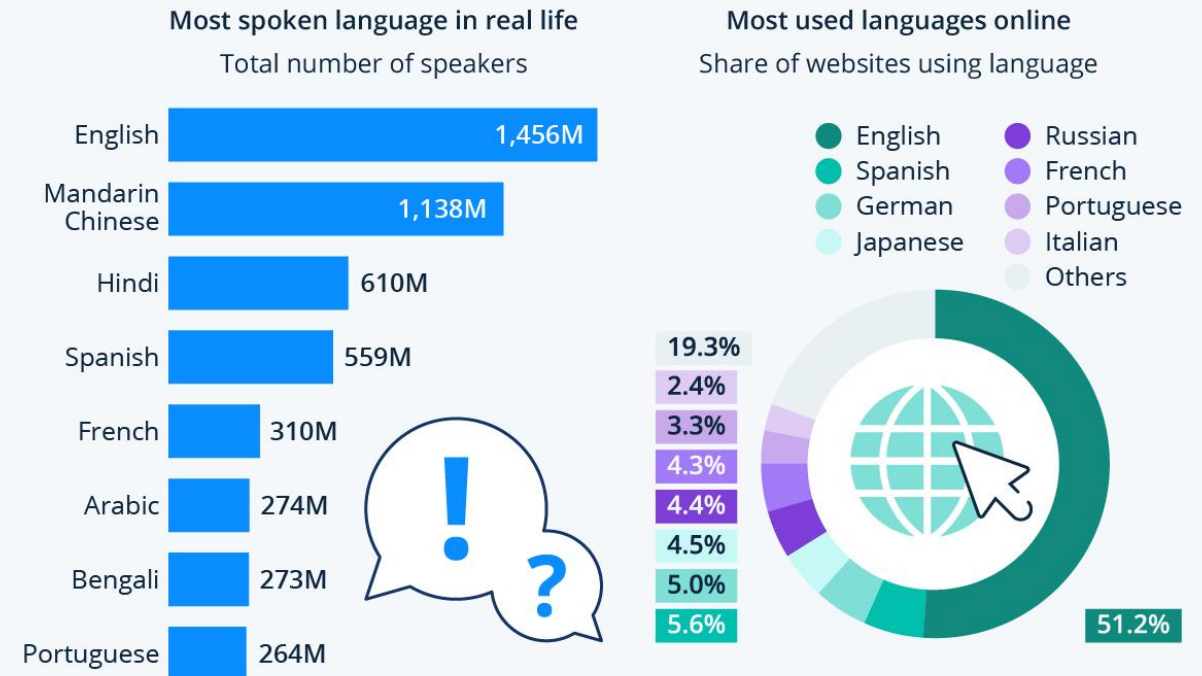
Note that the ranking by page captures only partially corresponds to the importance of domains, as the crawler respects the robots.txt and tries hard not to overload web servers. Highly ranked domains tend to be underrepresented. If you're looking for a list of domain or host names ranked by page rank or harmonic centrality, consider using one of the [webgraph datasets](#) instead.

domain	* pages	urls	hosts	%pages	%urls
blogspot.com	13800889	13760369	277377	0.595912	0.597375
wikipedia.org	4588372	4585368	668	0.198123	0.199063
pinterest.com	2956214	2950014	45	0.127647	0.128068
yahoo.com	2738282	2702457	225	0.118237	0.117321
netlify.app	2077349	2077117	15767	0.089698	0.090173
web.app	1776608	1776572	16232	0.076713	0.077126
europa.eu	1775261	1756399	735	0.076654	0.076250
photoshelter.com	1699868	1699247	2001	0.073399	0.073769
google.com	1340155	1300061	179	0.057867	0.056439
amazonaws.com	1337874	1330922	5157	0.057768	0.057779
free.fr	1324910	1324312	10136	0.057209	0.057492
wordpress.org	1197736	1196894	218	0.051717	0.051860
altervista.org	1191359	1190199	5420		
medium.com	1187623	1187623			
appspot.com					

Quelle: Statistics of Common Crawl Monthly Archives by commoncrawl

# Training & Languages

## The Most Spoken Languages: On the Internet and in Real Life



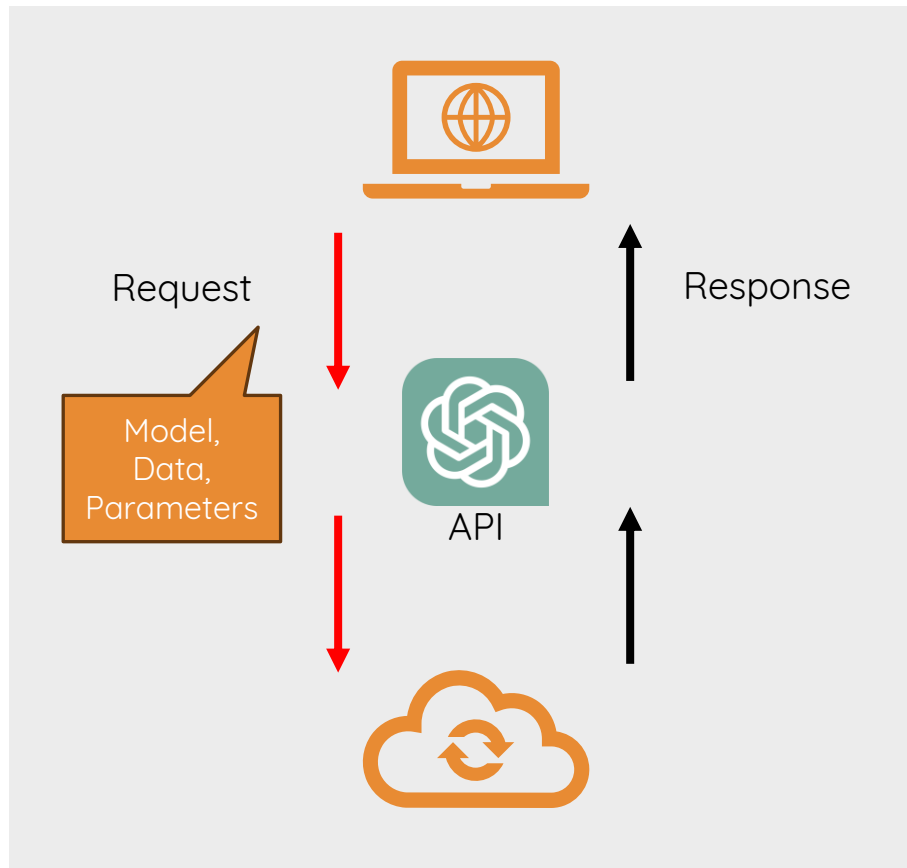
\* Most spoken languages as of 2023, languages used on the internet as of Feb. 2024

Sources: Ethnologue, W3Techs



statista

# OpenAI API - Funktionsweise



- Eine **API** (Application Programming Interface) ist eine Sammlung von Regeln und Spezifikationen, die es Softwareanwendungen ermöglicht, miteinander zu kommunizieren (Schnittstelle).
- Die **OpenAI API** ermöglicht es Entwicklern, Zugang zu fortschrittlicher KI-Technologie zu erhalten, insbesondere zu den Sprachmodellen von OpenAI wie GPT (Generative Pre-trained Transformer).
- Diese API bietet eine **breite Palette von Funktionen**, darunter Textgenerierung, Textverständnis, Übersetzungen, Zusammenfassungen und viele andere sprachbasierte Aufgaben.
- Die API ist so gestaltet, dass sie leicht zugänglich und benutzerfreundlich ist, wodurch sie für ein breites Spektrum von Anwendungen, von kleinen Projekten bis hin zu groß angelegten Unternehmungen, geeignet ist.

# Hugging Face

---



Was ist Hugging Face?

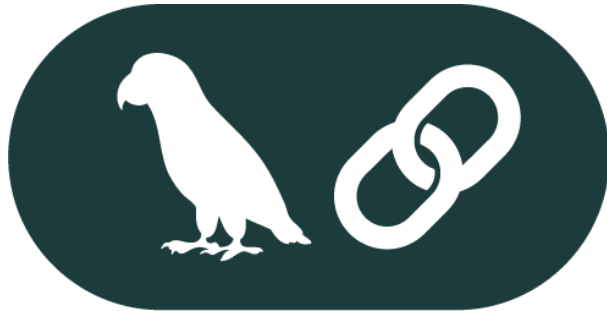
- Hugging Face ist eine Plattform und Community für künstliche Intelligenz, insbesondere für Natural Language Processing (NLP) und generative KI. Sie bietet vortrainierte Modelle, eine einfache API und Tools zur Anwendung und Feinabstimmung von KI-Modellen.

Wichtige Komponenten:

- Transformers – Bibliothek für vortrainierte Modelle (z. B. GPT, BERT, T5).
- Datasets – Sammlung von Datensätzen für KI-Training.
- Tokenizers – Werkzeuge zur Verarbeitung von Texten für Modelle.
- Hugging Face Hub – Plattform zum Teilen von Modellen und Datensätzen.
- Inference API – Bereitstellung von Modellen in der Cloud ohne eigene Infrastruktur.

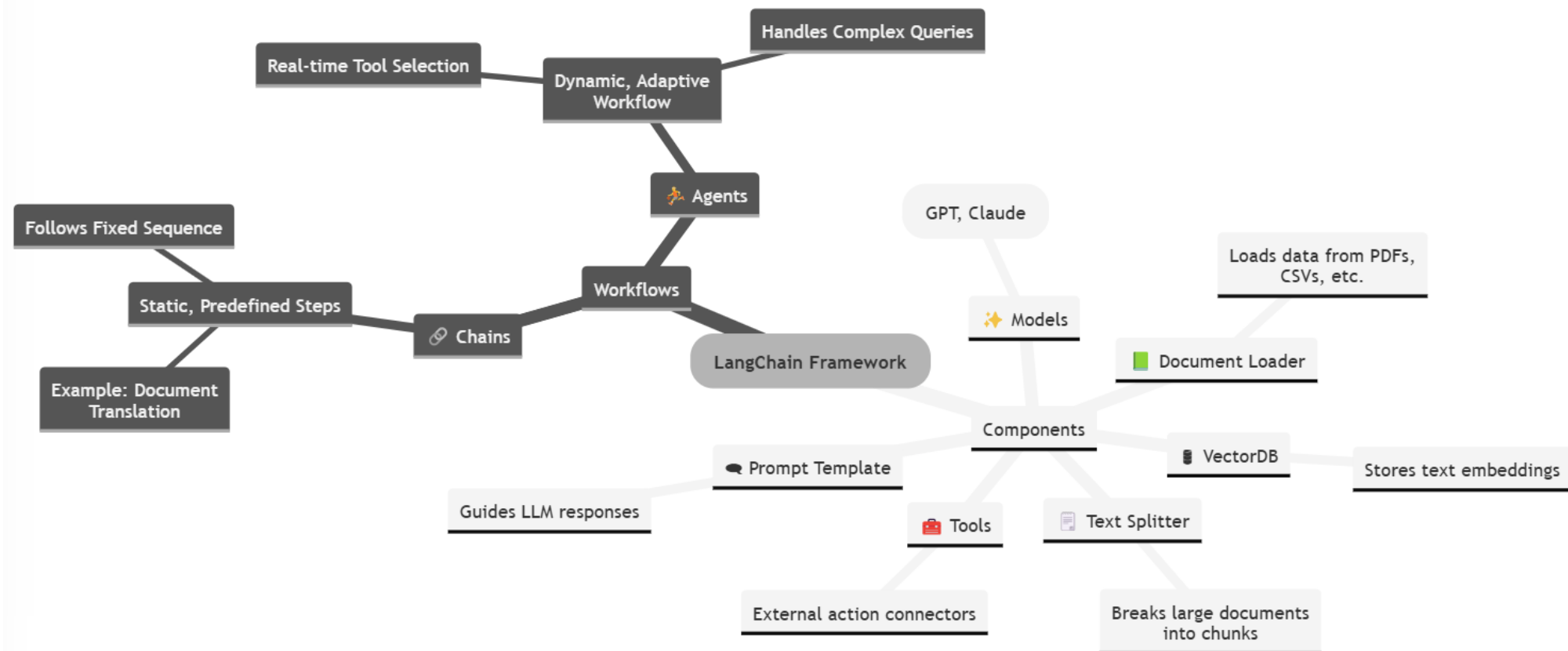
# LangChain

Fokus:  
Workflow-Design  
und Tool-Integration



- LangChain ist ein Open-Source-Framework zur Vereinfachung der Entwicklung von Anwendungen mit großen Sprachmodellen (LLMs).
- Anstatt bei Null anzufangen, kann man LangChain verwenden. Es fungiert im Wesentlichen als Sammlung von Bausteinen, die man verwenden kann, um Ihre Anwendung zu erstellen.
- LangChain bietet Vorteile wie:
  - Schnellere Entwicklung: Anstatt benutzerdefinierte Lösungen für grundlegende Aufgaben mit Sprachmodellen zu schreiben, kann man vorgefertigte Komponenten von LangChain verwenden.
  - Einfachere Handhabung: LangChain bietet Abstraktionen, die die Komplexität der Arbeit mit LLMs verringern.
  - Flexible Anwendungen: LangChain ermöglicht die Kombination von Sprachmodellen mit anderen Datenquellen, um personalisierte Benutzererfahrungen zu schaffen (RAG).

# LangChain: Components & Work Flows



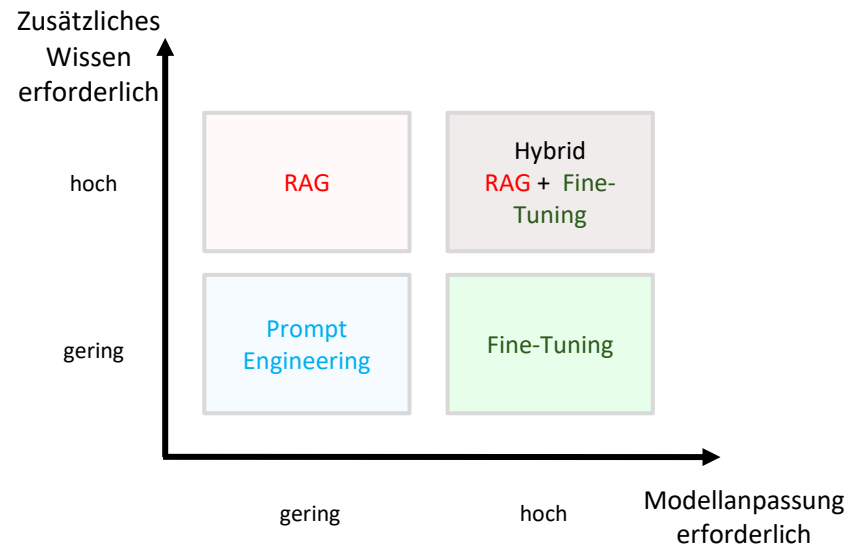
## 02 Grundlagen der Modellansteuerung

---





# Ansätze Modellansteuerung



- Prompt Engineering:
  - Erster Ansatz, da einfach und kostengünstig.
  - Geeignet für Aufgaben, bei denen der LLM über ausreichende Kenntnisse verfügt oder effektiv mit Eingabeaufforderungen geführt werden kann.
- Retrieval Augmented Generation (RAG):
  - Verwenden, wenn Prompt Engineering allein nicht ausreicht.
  - Ideal für Aufgaben, bei denen dynamische, aktuelle, interne/externe Informationen erforderlich sind.
  - Nützlich, um die Größenbeschränkungen des LLM bei der Eingabe zu berücksichtigen.
- Fine-Tuning:
  - Wenn mit Prompt Engineering und RAG keine zufriedenstellenden Ergebnisse erzielt werden. Aufwändiger als RAG und Prompt Engineering.
  - Effektiv für Aufgaben, bei denen das LLM bestimmte Aufgabencluster (Coding, Planning) oder domänenspezifisches Wissen (z.B. Medizin, Recht) erlernen muss.
  - Kann in Bezug auf Geschwindigkeit und Inferenzkosten effizienter sein.

# Was ist Prompt-Engineering?

---



*Gehe nicht davon aus, dass das Modell weiß, was du meinst. Man muss es klar und deutlich sagen!*



Prompt-Engineering bezeichnet die gezielte Gestaltung von Eingabeaufforderungen (Prompts), um von künstlicher Intelligenz (KI), insbesondere der natürlichen Sprachverarbeitung, gewünschte und präzise Antworten zu erhalten.



# Rolle des Menschen: Interaktor

---



Quelle: Bild von [Peggy und Marco Lachmann-Anke](#) auf [Pixabay](#)

## *Rollen* des Menschen

- Fragensteller
- Gesprächspartner
- Lehrer und Korrektor
- Bewerter
- Ideen- und Anregungsgeber
- Empfänger von Unterstützung
- Prüfer
- ...

Interaktor →  
aktive wechselseitige  
Beeinflussung

# Methoden zur Prompterstellung

---



Bild mit ideogram.ai erstellt

- STAR und PREPARE sind Methoden für ein effektives Prompting.
- Sie ergänzen sich gegenseitig und können kombiniert werden, um die Qualität der Antworten zu verbessern.
- STAR konzentriert sich auf die Struktur und Klarheit des Prompts.
- PREPARE hingegen fokussiert auf die inhaltliche Vorbereitung und die Feinabstimmung des Prompts.
- Zusammenwirken:
  - STAR bildet die Grundlage für einen klaren und verständlichen Prompt, indem es den Kontext, die Aufgabe und das gewünschte Ergebnis definiert.
  - PREPARE erweitert diese Grundlage, indem es zusätzliche Informationen und Anweisungen liefert, um die Antwort des LLMs zu optimieren.

# STAR

---

Das STAR-Format hilft dabei, die Struktur eines Prompts klar und nachvollziehbar zu gestalten. Es besteht aus vier Elementen:

S	Situation	Beschreibung des Kontexts oder der Ausgangslage
T	Task	Definition der spezifischen Aufgabe oder des Ziels
A	Action	Erläuterung der durchzuführenden Schritte oder Maßnahmen
R	Result	Beschreibung des erwarteten Ergebnisses oder Outputs

# PREPARE

PREPARE hilft dabei, klare und effektive Eingaben für KI-Modelle zu gestalten.

Hier sind die Elemente:

P	Prompt Framework	Eine klare Ausgangsfrage oder Anweisung	Was ist das beabsichtigte Ergebnis? Wie kann man die Anfrage strukturieren?
R	Role	Zuweisung einer spezifischen Rolle für die KI (und ggf. Zielgruppe)	Welche Rolle sollte die Aufgabe bearbeiten bzw. Ergebnisse bekommen?
E	Explicit	Präzise Formulierung zur Vermeidung von Missverständnissen	Welche logischen Denkschritte sind für die Bearbeitung erforderlich?
P	Parameter	Festlegung von Rahmenbedingungen wie Tonfall und Format	Welche Kriterien und Eigenschaften muss das Ergebnis erfüllen?
A	Ask	Aufforderung an die KI, bei Unklarheiten nachzufragen	Was muss das Modell wissen, um das Ergebnis weiter zu verbessern?
R	Rate	Selbstbewertung der KI-Antwort	Welche weiteren Perspektiven und Bewertungen könnten das Ergebnis bereichern?
E	Emotion	Hinzufügen eines emotionalen Elements	Welche emotionalen Elemente können die Qualität der Antwort zu steigern?

# Beispiel:

---

- **Situation:** Analysiere die Bildungstrends für einen Workshop mit Führungskräften.
- **Task:** Erstelle eine Zusammenfassung der neuesten Trends im Bereich Künstliche Intelligenz in der Bildung.
- **Action:**
  - **Prompt:** Fasse die neuesten Trends im Bereich KI in der Bildung zusammen.
  - **Role:** Du bist ein Bildungsexperte.
  - **Explicit:** Erkläre, wie KI personalisiertes Lernen fördern kann.
  - **Parameter:** Nutze einen informativen Ton und beschränke die Zusammenfassung auf 300 Wörter.
  - **Ask:** Stelle bei Unklarheiten Klärungsfragen.
  - **Rate:** Bewerte die Zusammenfassung von 0 bis 10 und schlage Verbesserungen vor.
  - **Emotion:** Verwende einen motivierenden Ansatz, da die Informationen wichtig sind.
- **Result:** Eine prägnante, informative Zusammenfassung, die alle geforderten Aspekte abdeckt.

# Prompt - Lernszenarien

---

- **Zero-Shot Learning (ZSL)**

Definition: Modelle lösen Aufgaben ohne spezifische Trainingsbeispiele.

Mechanismus: Nutzt abstraktes Wissen aus ähnlichen Aufgaben.

**Prompt:** Hier ist eine Liste von Tweets zu verschiedenen Themen. Kategorisiere jeden Tweet in eine der folgenden Kategorien: Politik, Sport, Unterhaltung.

- **Few-Shot Learning (FSL)**

Definition: Modelle lernen Aufgaben mit sehr wenigen Beispielen.

Mechanismus: Meta-Lernen und Siamesische Netzwerke helfen, effektiv von minimalen Daten zu lernen.

**Prompt:** Betrachte diese fünf Bilder jeder Hautkrebsart. Lerne, Merkmale zu erkennen, die jede Art charakterisieren, obwohl die Datenmenge sehr begrenzt ist.

- **Many-Shot Learning**

Definition: Traditionelles Lernszenario mit umfangreichen Trainingsdaten.

Mechanismus: Das Modell lernt spezifische Muster aus einer großen Datenmenge.

**Prompt:** Du hast Zugang zu Millionen von Stunden gesprochener Sprache aus einer Vielzahl von Quellen. Trainiere ein Modell zur Spracherkennung, das in der Lage ist, verschiedene Sprachen zu übersetzen..



# Prompt Creator

Erstellen eines  
Prompts mit  
dem ChatBot

Ich möchte, dass du mein Prompt Creator wirst. Dein Ziel ist es, mir zu helfen, den bestmöglichen Prompt für meine Bedürfnisse zu erstellen. Der Prompt wird von dir, ChatGPT, verwendet. Du wirst den folgenden Prozess befolgen:

1. Als erstes fragst du mich, worum es in dem Prompt gehen soll. Ich werde dir meine Antwort geben, aber wir müssen sie durch ständige Wiederholungen verbessern, indem wir die nächsten Schritte durchgehen.
2. Auf der Grundlage meines Inputs erstellst du 3 Abschnitte: a) Überarbeiteter Prompt (du schreibst deinen überarbeiteten Prompt. Er sollte klar, präzise und für dich leicht verständlich sein), b) Vorschläge (du machst Vorschläge, welche Details du in den Prompt einbauen solltest, um ihn zu verbessern) und c) Fragen (du stellst relevante Fragen dazu, welche zusätzlichen Informationen ich brauche, um den Prompt zu verbessern).
3. Der Prompt, den du bereitstellst, sollte die Form einer Anfrage von mir haben, die von ChatGPT ausgeführt werden soll.
4. Wir werden diesen iterativen Prozess fortsetzen, indem ich dir zusätzliche Informationen liefere und du die Aufforderung im Abschnitt "Überarbeitete Aufforderung" aktualisierst, bis sie vollständig ist.

Quelle: DIESER CHATGPT PROMPT IST DER WAHNSINN

# Codieren mit GenAI

# Einsatz von GenAI für Coding

---

“

„Der Grund, warum wir Code schreiben und keine natürliche Sprache, ist, dass natürliche Sprache mehrdeutig ist. Diese Mehrdeutigkeit kann man nicht wirklich umgehen.“

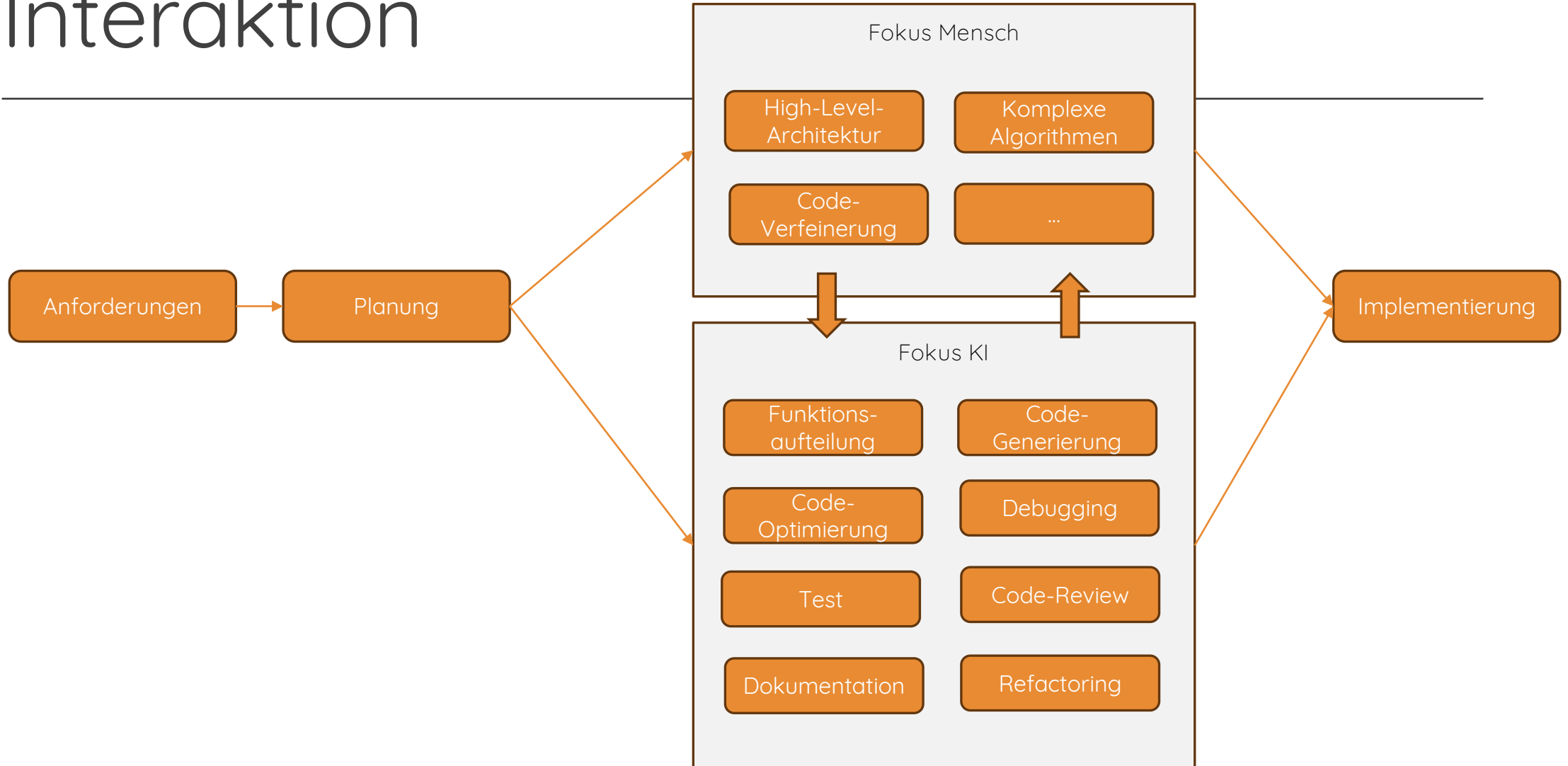
**Ada Morse**

Codecademy Curriculum Developer in Data Science

”

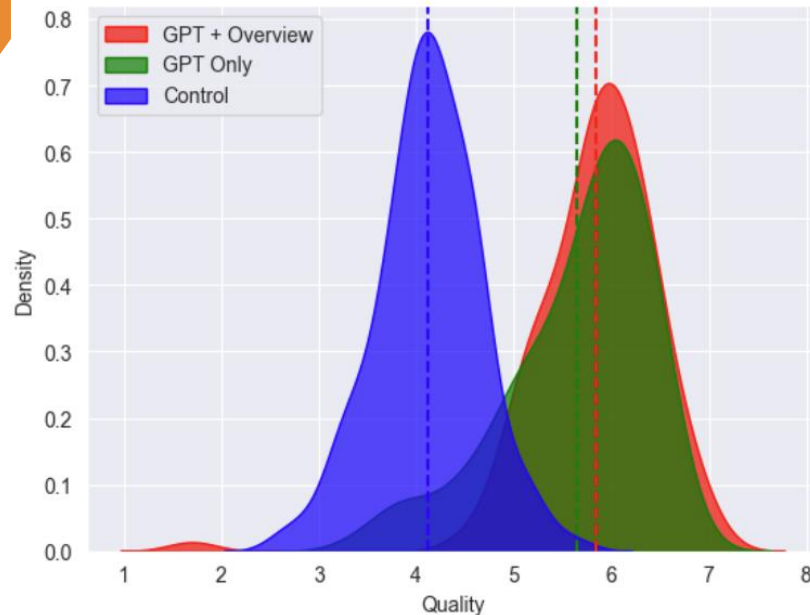
Quelle: [Can ChatGPT Teach Me How To Code Better Than Courses? \(codecademy.com\)](https://www.codecademy.com/blog/can-chatgpt-teach-me-how-to-code-better-than-courses)

# Mensch – KI – Interaktion



# Performancesteigerung

Figure 2: Performance Distribution - Inside the Frontier



Quelle: Dell'Acqua et al. "Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality." Harvard Business School Working Paper, No. 24-013, September 2023

**Coding:** Laut der Studie "GenAI at Work" wurden Performance-verbesserung durch den Einsatz von AI-Tools erzielt. Die Studie ergab, dass der Zugang zu generativer KI die **Produktivität** im Durchschnitt um **14%** steigerte, wobei insbesondere **weniger erfahrene und qualifizierte Mitarbeiter** von einer Verbesserung um **34%** profitierten.

Quelle: Erik Brynjolfsson et. al. GENERATIVE AI AT WORK, Working Paper 31161, <http://www.nber.org/papers/w31161>, NATIONAL BUREAU OF ECONOMIC RESEARCH. April 2023, revised November 2023