

PROMPT ENGINEERING MEISTERN

BAND 1

# Grundlagen

Dein Einstieg in die Welt der KI-Kommunikation

---

Belkis Aslani

---

2026

## **Prompt Engineering Meistern**

Band 1: Grundlagen – Dein Einstieg in die Welt der KI-Kommunikation

© 2026 Belkis Aslani. Alle Rechte vorbehalten.

1. Auflage, März 2026

Dieses Werk ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Autors unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die in diesem Buch genannten Produkt- und Firmennamen sind Marken der jeweiligen Eigentümer.

Satz und Layout: Eigensatz des Autors

Umschlaggestaltung: Belkis Aslani

# Inhaltsverzeichnis

---

*Vorwort*

---

**1** Was ist KI eigentlich?

---

**2** LLMs verstehen – Wie Sprachmodelle funktionieren

---

**3** Dein erster Prompt

---

**4** Anatomie eines guten Prompts

---

**5** Die häufigsten Fehler beim Prompting

---

**6** Kontext ist alles

---

**7** Rollen und Personas – Die KI in eine Rolle schlüpfen lassen

---

**8** Output-Formate steuern – Damit die KI liefert, was du brauchst

---

**9** Iteratives Prompting – Warum der erste Versuch selten reicht

---

**10** Zusammenfassung und Ausblick

---

# Vorwort

---

Ich bin Belkis Aslani, und ich muss dir was gestehen: Als ich zum ersten Mal von ChatGPT gehört habe, dachte ich mir – naja, wieder so ein Hype. Wie Blockchain. Wie Clubhouse. Kommt und geht.

Ich lag komplett falsch.

Innerhalb weniger Wochen hat sich mein kompletter Arbeitsalltag verändert. Nicht weil die KI magisch alles besser konnte, sondern weil ich angefangen habe, richtig mit ihr zu kommunizieren. Und genau da liegt der Knackpunkt, den die meisten Leute übersehen: Es geht nicht darum, *ob* du KI benutzt. Es geht darum, *wie* du mit ihr redest.

Dieses Buch ist der erste Band einer zehnteiligen Reihe. Zehn Bände – das klingt nach viel, ich weiß. Aber Prompt Engineering ist ein Thema, das sich in den letzten zwei Jahren so rasant entwickelt hat, dass ein einzelnes Buch dem einfach nicht gerecht wird. Wir reden hier über eine Fähigkeit, die 2026 so grundlegend ist wie Googeln im Jahr 2005.

## Für wen ist dieses Buch?

Für dich, wenn du noch nie mit einer KI gechattet hast. Oder wenn du es schon mal probiert hast, aber die Ergebnisse eher meh waren. Oder wenn du einfach wissen willst, was dieses ganze “Prompt Engineering” eigentlich soll.

Du brauchst kein technisches Vorwissen. Keinen Informatik-Abschluss. Nicht mal besonders gute Englischkenntnisse, obwohl viele Fachbegriffe aus dem Englischen kommen – ich erkläre sie alle.

Was du brauchst: Einen Computer oder ein Smartphone mit Internetzugang. Das war's.

## Wie du dieses Buch am besten nutzt

Lies es der Reihe nach. Die Kapitel bauen aufeinander auf. Kapitel 1 erklärt, was KI überhaupt ist. Kapitel 3 lässt dich deinen ersten Prompt schreiben. Ab Kapitel 4 wird's dann richtig spannend, weil du lernst, *warum* manche Prompts funktionieren und andere nicht.

Am Ende jedes Kapitels findest du Übungen. Mach die. Ernsthaft. Prompt Engineering ist wie Kochen – du kannst hundert Rezepte lesen, aber solange du nicht selbst am Herd stehst, wirst du nicht besser.

Ich habe in jedem Kapitel echte Beispiele eingebaut. Keine theoretischen Konstrukte, sondern Prompts, die ich selbst benutze oder die mir Leute aus meinem Umfeld gezeigt haben. Manche davon sind absichtlich schlecht – damit du siehst, was passiert, wenn man es falsch macht. Und dann zeige ich dir, wie man es besser macht.

## Ein Wort zum Thema KI und Angst

Ich treffe immer wieder Leute, die Angst vor KI haben. “Die nimmt uns die Jobs weg.” “Die wird irgendwann die Weltherrschaft übernehmen.” “Das ist gefährlich.”

Diese Ängste nehme ich ernst. Aber ich glaube, die beste Antwort auf Angst ist Wissen. Wenn du verstehst, wie LLMs funktionieren – und Band 1 gibt dir genau dieses Verständnis –, dann merkst du schnell: Das sind Werkzeuge.

Mächtige Werkzeuge, ja. Aber Werkzeuge. Eine Kreissäge kann auch gefährlich sein, wenn man nicht weiß, wie man sie bedient. Das heißt nicht, dass man sie verbieten muss. Es heißt, dass man lernen sollte, damit umzugehen.

## Danksagung

Dieses Buch wäre nicht möglich gewesen ohne die unzähligen Gespräche mit Freunden, Kollegen und Fremden im Internet, die ihre Erfahrungen mit KI geteilt haben. Besonderer Dank geht an die Open-Source-Community, die Forschenden, die ihre Erkenntnisse frei zugänglich machen, und an jeden, der mutig genug war, mir seine schlimmsten Prompts zu zeigen. Wir haben alle dabei gelernt.

Und jetzt: Lass uns loslegen.

*Belkis Aslani, März 2026*

# Kapitel 1: Was ist KI eigentlich?

---

Bevor wir über Prompts reden, müssen wir über das reden, was dahintersteckt. Also: Was ist Künstliche Intelligenz? Und was ist sie *nicht*?

Ich fange mal mit dem an, was sie nicht ist. KI ist kein denkendes Wesen. Sie hat kein Bewusstsein. Sie hat keine Meinung, keine Gefühle und keinen freien Willen. Wenn ChatGPT dir schreibt “Das finde ich eine gute Idee!”, dann *findet* es gar nichts. Es produziert Text, der in diesem Kontext statistisch wahrscheinlich ist. Mehr nicht. Weniger aber auch nicht – denn das ist schon ziemlich beeindruckend.

## KI in einfachen Worten

Stell dir vor, du hast einen sehr, sehr fleißigen Praktikanten. Dieser Praktikant hat jedes Buch gelesen, das je geschrieben wurde. Jeden Wikipedia-Artikel. Jedes Forum. Jede wissenschaftliche Arbeit. Er kann dir auf fast jede Frage eine Antwort geben – aber er versteht die Antwort nicht wirklich. Er erkennt Muster. Er weiß, welche Wörter normalerweise nach welchen anderen Wörtern kommen. Er kann unglaublich überzeugend klingen, aber manchmal erzählt er dir mit völliger Überzeugung kompletten Unsinn.

Das ist KI. Zumindest die Art von KI, über die wir in diesem Buch reden.

## Kurze Geschichte – ohne Langeweile

Die Idee, dass Maschinen denken könnten, ist nicht neu. Alan Turing hat sich das schon 1950 gefragt. Er hat den berühmten “Turing-Test” vorgeschlagen: Wenn ein Mensch nicht unterscheiden kann, ob er mit einem Menschen oder einer Maschine chattet, dann ist die Maschine intelligent. Zumindest nach seiner Definition.

Danach passierte jahrzehntelang erstmal wenig. Klar, es gab Fortschritte. Schachcomputer. Spamerkennung in E-Mails. Sprachassistenten wie Siri, die meistens nicht verstanden haben, was man wollte. Aber der große Durchbruch kam erst 2017 mit einem Forschungspapier von Google: “Attention Is All You Need”.

Klingt unspektakulär? War es aber nicht. Dieses Paper hat die sogenannte Transformer-Architektur vorgestellt – das technische Fundament, auf dem heute ChatGPT, Claude, Gemini und alle anderen großen Sprachmodelle aufgebaut sind.

Und dann ging es schnell:

- **2018:** GPT-1 von OpenAI – konnte halbwegs zusammenhängende Texte schreiben
- **2020:** GPT-3 – plötzlich konnte die KI Aufsätze schreiben, Code generieren und Fragen beantworten
- **2022:** ChatGPT – der Moment, in dem die breite Öffentlichkeit aufmerksam wurde. 100 Millionen Nutzer in zwei Monaten.
- **2023:** GPT-4, Claude 2, Gemini – KI wird multimodal (kann auch Bilder verstehen)
- **2024:** Claude 3, Gemini 1.5, LLaMA 3 – die Modelle werden besser, schneller, günstiger
- **2025-2026:** Claude 4, GPT-5, DeepSeek, Agenten-KI – wir sind hier

# Schwache KI vs. Starke KI

In der Wissenschaft unterscheidet man zwei Arten von KI:

**Schwache KI (Narrow AI):** Kann eine bestimmte Aufgabe richtig gut. Schach spielen. Texte schreiben. Bilder erkennen. Aber nur das. Ein Schach-KI kann keine Gedichte schreiben. Ein Textgenerator kann kein Schach spielen. Alles, was du heute benutzen kannst, ist schwache KI.

**Starke KI (AGI – Artificial General Intelligence):** Könnte alles, was ein Mensch kann. Und mehr. Gibt es noch nicht. Ob es sie jemals geben wird – darüber streiten sich die Experten. Manche sagen, wir sind nah dran. Andere sagen, wir sind Jahrzehnte entfernt. Für dieses Buch ist das egal, denn wir arbeiten mit dem, was es gibt. Und das ist schon mehr als genug.

## Wo begegnet dir KI im Alltag?

Wahrscheinlich öfter, als du denkst:

- **Smartphone:** Autokorrektur, Gesichtserkennung, Sprachassistenten
- **Social Media:** Der Algorithmus, der entscheidet, was du siehst
- **Online-Shopping:** “Kunden, die das gekauft haben, kauften auch...”
- **Navigation:** Google Maps schätzt deine Ankunftszeit basierend auf Verkehrsdaten
- **E-Mail:** Spam-Filter sortieren den Müll aus
- **Streaming:** Netflix und Spotify schlagen dir vor, was du als nächstes schauen oder hören könntest
- **Bankwesen:** Betrugserkennung bei Kreditkartenzahlungen

Das alles ist KI. Keine Magie. Keine Science-Fiction. Einfach Mathematik, Statistik und unglaublich viele Daten.

## Was hat das mit Prompts zu tun?

Gute Frage. Die KI-Systeme, die ich gerade aufgezählt habe, arbeiten meistens im Hintergrund. Du gibst ihnen keine Anweisungen – sie tun einfach ihr Ding. Spam erkennen. Wege berechnen. Empfehlungen aussprechen.

Large Language Models sind anders. Die reagieren auf das, was du ihnen sagst. Und das, was du ihnen sagst – dein Prompt – bestimmt, was du zurückbekommst. Es ist ein Gespräch, kein Knopfdruck.

Deswegen ist Prompt Engineering so wichtig. Weil die Qualität deiner Ergebnisse direkt davon abhängt, wie gut du mit der KI kommunizierst. Und genau das lernst du in den nächsten Kapiteln.

## Das Wichtigste aus diesem Kapitel

- KI ist ein Werkzeug, kein denkendes Wesen
- Aktuelle KI-Systeme (LLMs) basieren auf der Transformer-Architektur von 2017
- Alles, was wir heute nutzen, ist “schwache KI” – gut in einer Sache, aber nicht universell intelligent
- KI begegnet dir im Alltag ständig, aber LLMs sind besonders, weil du direkt mit ihnen kommunizierst
- Die Qualität deiner KI-Ergebnisse hängt von der Qualität deiner Prompts ab

---

## Übung

**Finde 5 KI-Anwendungen in deinem Alltag.**

Nimm dir 10 Minuten und geh durch deinen normalen Tag. Vom Aufstehen bis zum Schlafengehen. Wo begegnet dir KI? Schreib mindestens fünf Beispiele auf. Bonus: Überlege, welche davon im Hintergrund arbeiten (wie der Spam-Filter) und welche du aktiv benutzt (wie einen Sprachassistenten).

Wenn du weniger als fünf findest, bist du wahrscheinlich nicht genau genug hingeschaut. Die meisten Leute kommen locker auf zehn oder mehr, wenn sie ehrlich sind.

# Kapitel 2: LLMs verstehen – Wie Sprachmodelle funktionieren

---

OK, jetzt wird's ein bisschen technischer. Aber keine Sorge – ich erkläre das so, dass du es auch ohne Informatik-Studium verstehst. Versprochen.

## Was sind Large Language Models?

LLM steht für “Large Language Model” – großes Sprachmodell. Der Name sagt eigentlich schon alles:

- **Large:** Diese Modelle sind riesig. Sie haben Milliarden von Parametern. GPT-4 hat geschätzte 1,8 Billionen Parameter. Claude und Gemini spielen in ähnlichen Größenordnungen. Das sind Zahlen, die sich kein Mensch mehr vorstellen kann.
- **Language:** Sie arbeiten mit Sprache. Text rein, Text raus. Manche können inzwischen auch Bilder und Audio verarbeiten, aber im Kern sind es Sprachmodelle.
- **Model:** Es sind mathematische Modelle. Keine Programme im klassischen Sinn, die Regeln abarbeiten. Sondern statistische Modelle, die Muster gelernt haben.

## Wie “denkt” ein LLM?

Kurzantwort: Gar nicht. Jedenfalls nicht so, wie du und ich denken.

Wenn du ChatGPT fragst “Was ist die Hauptstadt von Frankreich?”, dann passiert Folgendes: Das Modell schaut sich deine Frage an und berechnet, welches Wort am wahrscheinlichsten als nächstes kommt. Und dann das nächste. Und das nächste. Wort für Wort.

Es denkt nicht: “Hmm, Frankreich... Paris ist die Hauptstadt.” Es “denkt”: “Nach ‘Die Hauptstadt von Frankreich ist’ kommt mit 98,7% Wahrscheinlichkeit das Wort ‘Paris’.”

Ich weiß, das klingt ernüchternd. Wie kann etwas, das nur Wörter aneinanderreihet, so kluge Antworten geben? Der Trick liegt im Training. Diese Modelle haben so unglaublich viel Text gesehen, dass ihre Vorhersagen erstaunlich gut sind. Manchmal täuschend gut. Manchmal so gut, dass man vergisst, dass da keine Intelligenz dahintersteckt.

Aber genau deshalb passieren auch Fehler. Sogenannte **Halluzinationen** – wenn das Modell dir mit fester Überzeugung etwas Falsches erzählt. Dazu kommen wir in Kapitel 5 noch genauer.

## Training: Woher kommt das Wissen?

LLMs werden mit gigantischen Textmengen trainiert. Bücher, Webseiten, Wikipedia, Code-Repositories, wissenschaftliche Paper, Foren, Nachrichtenartikel – alles, was in Textform existiert und zugänglich ist.

Das Training läuft grob in zwei Phasen:

### **Phase 1: Pre-Training**

Das Modell liest (vereinfacht gesagt) das halbe Internet und lernt, wie Sprache funktioniert. Welche Wörter zusammengehören. Wie Sätze aufgebaut sind. Welche Antwort auf welche Frage typischerweise folgt. Das ist der teure Teil – Pre-Training kostet Hunderte Millionen Euro an Rechenleistung.

## Phase 2: Fine-Tuning und RLHF

Das rohe Modell nach dem Pre-Training ist wie ein Lexikon mit Persönlichkeitsstörung. Es weiß viel, aber es antwortet wirr und manchmal gefährlich. Deshalb wird es nachtrainiert:

- **Fine-Tuning:** Menschliche Trainer zeigen dem Modell, wie gute Antworten aussehen
- **RLHF (Reinforcement Learning from Human Feedback):** Menschen bewerten Antworten als gut oder schlecht, und das Modell lernt, bessere Antworten zu geben

Das Ergebnis ist das, was du als ChatGPT, Claude oder Gemini kennst – ein Modell, das höflich, hilfreich und halbwegs sicher antwortet.

## Die wichtigsten Begriffe – einfach erklärt

Wenn du über LLMs liest, stolperst du ständig über ein paar Fachbegriffe. Hier sind die wichtigsten:

### Token

LLMs denken nicht in Wörtern, sondern in Token. Ein Token ist ein Stückchen Text – manchmal ein ganzes Wort, manchmal nur ein Teil davon.

Beispiel: Das Wort “Prompt” ist meistens ein Token. Das Wort “Prompt Engineering” sind zwei Token. Das Wort “Unwahrscheinlichkeit” wird in mehrere Token zerlegt.

Warum ist das wichtig? Weil LLMs ein Limit haben, wie viele Token sie auf einmal verarbeiten können. Das bringt uns zum nächsten Begriff.

## Context Window

Das Context Window (Kontextfenster) ist die maximale Menge an Text, die ein LLM gleichzeitig “sehen” kann. Das umfasst sowohl deinen Prompt als auch die Antwort des Modells.

- GPT-4: bis zu 128.000 Token (~100 Buchseiten)
- Claude 3.5/4: bis zu 200.000 Token (~150 Buchseiten)
- Gemini 1.5/2: bis zu 1.000.000 Token (~750 Buchseiten)

Das klingt nach viel, und meistens reicht es auch. Aber bei sehr langen Dokumenten oder komplexen Gesprächen kann es eng werden. Dann “vergisst” das Modell ältere Teile des Gesprächs.

## Parameter

Parameter sind die “Stellschrauben” des Modells. Je mehr Parameter, desto mehr Muster kann das Modell lernen. Mehr Parameter bedeutet tendenziell bessere Ergebnisse – aber auch höhere Kosten und langsamere Antworten.

Du musst die genaue Zahl nicht kennen. Merk dir einfach: Größere Modelle sind meistens besser, kosten aber mehr.

## Temperatur

Die Temperatur steuert, wie “kreativ” das Modell antwortet.

- **Temperatur 0:** Sehr vorhersagbar. Das Modell wählt immer das wahrscheinlichste Wort. Gut für Fakten, Zusammenfassungen, Code.
- **Temperatur 1:** Kreativer, aber auch unvorhersagbarer. Gut für Brainstorming, Geschichten, kreatives Schreiben.
- **Temperatur > 1:** Zunehmend chaotisch. Kann lustig sein, produziert aber oft Unsinn.

In den meisten Chat-Interfaces (ChatGPT, Claude) kannst du die Temperatur nicht direkt einstellen – das machen nur Entwickler über die API. Aber es hilft zu verstehen, warum du manchmal verschiedene Antworten auf die gleiche Frage bekommst.

## Die wichtigsten LLMs – eine Übersicht

Stand März 2026 gibt es eine Handvoll Modelle, die du kennen solltest:

### ChatGPT / GPT-4o / GPT-5 (OpenAI)

Der Platzhirsch. Das bekannteste LLM. Gut für allgemeine Aufgaben, Konversation, Code. Kostenlose Version verfügbar, Pro-Version mit erweiterten Funktionen.

### Claude (Anthropic)

Mein persönlicher Favorit für längere Texte und komplexe Aufgaben. Claude hat ein besonders großes Kontextfenster und ist gut darin, Nuancen zu verstehen. Aktuell in Version 4.

### Gemini (Google)

Googles Antwort auf ChatGPT. Stark bei Recherche und Fakten, weil es Zugriff auf Google-Suche hat. Riesiges Kontextfenster von bis zu einer Million Token.

### LLaMA / Meta AI (Meta)

Open-Source-Modell von Meta. Kann kostenlos heruntergeladen und auf eigenem Rechner betrieben werden. Besonders interessant für Entwickler und Datenschutz-Bewusste.

## DeepSeek (DeepSeek)

Ein chinesisches Open-Source-Modell, das 2025 für Aufsehen gesorgt hat, weil es trotz geringerer Trainingskosten mit den Top-Modellen mithalten konnte. Zeigt, dass mehr Geld nicht automatisch bessere Modelle bedeutet.

## Mistral (Mistral AI)

Europäisches Open-Source-Modell aus Frankreich. Kompakt, schnell und überraschend leistungsfähig. Zeigt, dass auch kleinere Modelle ihren Platz haben.

## Welches Modell ist das beste?

Keins. Oder alle. Es kommt drauf an.

Ich weiß, das ist eine unbefriedigende Antwort. Aber es ist die ehrliche. Jedes Modell hat Stärken und Schwächen. ChatGPT ist gut für schnelle Konversation. Claude ist stark bei langen, komplexen Texten. Gemini kann live im Internet suchen.

Mein Tipp: Leg dir kostenlose Accounts bei mindestens zwei verschiedenen Anbietern an. Teste denselben Prompt in verschiedenen Modellen. Du wirst schnell merken, was dir liegt.

## Das Wichtigste aus diesem Kapitel

- LLMs sind statistische Modelle, die Wort für Wort vorhersagen
- Sie “denken” nicht – sie berechnen Wahrscheinlichkeiten
- Training besteht aus Pre-Training (viel Text lesen) und Fine-Tuning (gute Antworten lernen)
- Token sind die Grundeinheit – Modelle haben ein begrenztes Kontextfenster

- Es gibt viele gute LLMs – kein einzelnes ist “das Beste” für alles
- 

## Übung

### **Teste denselben Prompt in zwei verschiedenen LLMs.**

1. Geh auf [chat.openai.com](https://chat.openai.com) und erstelle einen kostenlosen Account (falls du noch keinen hast)
2. Geh auf [claude.ai](https://claude.ai) und erstelle einen kostenlosen Account
3. Gib beiden genau denselben Prompt: “Erkläre mir in 3 Sätzen, warum der Himmel blau ist.”
4. Vergleiche die Antworten:
5. Welche ist verständlicher?
6. Welche ist genauer?
7. Welche gefällt dir besser – und warum?

Schreib deine Beobachtungen auf. Das ist der Anfang deines Prompt-Tagebuchs, das dir in den nächsten Kapiteln noch sehr nützlich sein wird.

# Kapitel 3: Dein erster Prompt

---

Genug Theorie. Jetzt wird's praktisch.

In diesem Kapitel schreibst du deinen ersten Prompt. Und deinen zweiten. Und deinen zehnten. Am Ende wirst du ein Gefühl dafür haben, was passiert, wenn du mit einem LLM sprichst – und was nicht.

## Was ist ein Prompt überhaupt?

Ein Prompt ist die Eingabe, die du einem LLM gibst. Das kann eine Frage sein, eine Anweisung, eine Bitte – oder alles zusammen. Im Grunde ist es das, was du in das Textfeld tippst.

Das Wort kommt aus dem Englischen und bedeutet so viel wie “Anstoß” oder “Aufforderung”. Du stößt die KI an, etwas zu tun.

Ein paar Beispiele für Prompts:

- “Was ist die Hauptstadt von Kanada?”
- “Schreib mir eine Einkaufsliste für ein italienisches Abendessen.”
- “Fasse diesen Text in drei Sätzen zusammen: [Text]”
- “Du bist ein erfahrener Fitness-Trainer. Erstelle mir einen Trainingsplan für Anfänger.”

Alles davon sind Prompts. Der erste ist simpel. Der letzte ist schon deutlich raffinierter. In den nächsten Kapiteln lernst du, warum – und wie du selbst solche Prompts baust.

## Account erstellen – so geht's

Falls du noch keinen Account hast, hier die schnellste Route:

### Option A: ChatGPT (OpenAI)

1. Geh auf [chat.openai.com](https://chat.openai.com)
2. Klick auf “Registrieren”
3. E-Mail, Passwort, fertig
4. Die kostenlose Version (GPT-4o mini) reicht für den Anfang

### Option B: Claude (Anthropic)

1. Geh auf [claude.ai](https://claude.ai)
2. Registriere dich mit E-Mail oder Google-Account
3. Die kostenlose Version gibt dir Zugang zu Claude 3.5 Sonnet

### Option C: Gemini (Google)

1. Geh auf [gemini.google.com](https://gemini.google.com)
2. Melde dich mit deinem Google-Account an
3. Kostenlos nutzbar

Ich empfehle dir, mindestens zwei davon auszuprobieren. Für dieses Buch ist es egal, welches Modell du benutzt – die Prinzipien funktionieren bei allen.

## Dein allererster Prompt

Öffne dein bevorzugtes LLM und tippe Folgendes ein:

```
Erkläre mir, was Prompt Engineering ist.
```

Das war's. Drück Enter. Lies die Antwort.

War das jetzt ein guter Prompt? Naja, er funktioniert. Du bekommst eine Antwort. Aber er ist auch ziemlich vage. Das Modell weiß nicht, wer du bist, was du schon weißt oder wie detailliert die Antwort sein soll.

Jetzt probier diesen hier:

```
Ich bin Anfänger und habe noch nie mit KI gearbeitet. Erkläre mir in einfachen Worten, was Prompt Engineering ist. Verwende ein Alltagsbeispiel, um es anschaulich zu machen. Maximal 200 Wörter.
```

Merkst du den Unterschied in der Antwort? Der zweite Prompt gibt dem Modell viel mehr Informationen:

- **Wer du bist:** Anfänger ohne Vorwissen
- **Was du willst:** Erklärung von Prompt Engineering
- **Wie du es willst:** Einfache Worte, Alltagsbeispiel
- **Wie lang:** Maximal 200 Wörter

Das ist im Kern schon Prompt Engineering: Dem Modell genug Informationen geben, damit es dir genau das liefert, was du brauchst.

## 10 Prompts zum Ausprobieren

Hier sind zehn Prompts, die du jetzt direkt testen kannst. Ich habe sie nach Schwierigkeit sortiert – von simpel bis schon ziemlich clever:

### 1. Die einfache Frage

```
Was ist der Unterschied zwischen Wetter und Klima?
```

### 2. Die Zusammenfassung

Fasse die Handlung von "Der Herr der Ringe" in 5 Sätzen zusammen.

### **3. Die Liste**

Gib mir 7 Tipps, wie ich morgens schneller wach werde.

### **4. Die Übersetzung**

Übersetze folgenden Satz ins Englische, Französische und Spanische:

"Ich hätte gerne einen Tisch für zwei Personen."

### **5. Die Analyse**

Was sind die Vor- und Nachteile von Homeoffice? Erstelle eine Tabelle mit je 5 Punkten.

### **6. Die kreative Aufgabe**

Schreib mir ein kurzes Gedicht über Montagmorgen. Es soll humorvoll sein.

### **7. Die Rollenaufgabe**

Du bist ein Ernährungsberater. Was sollte ich frühstücken, wenn ich mich gesund ernähren will, aber morgens wenig Zeit habe?

### **8. Die Vergleichsaufgabe**

Vergleiche Python und JavaScript für Anfänger. Was ist leichter zu lernen und warum? Antworte in maximal 150 Wörtern.

## 9. Die Schritt-für-Schritt-Anleitung

Erkläre mir Schritt für Schritt, wie ich eine Pressemitteilung schreibe. Gib mir für jeden Schritt ein konkretes Beispiel.

## 10. Die komplexe Aufgabe

Ich möchte eine Geburtstagsparty für 20 Personen planen.  
Budget: 500 Euro.  
Die Gäste sind zwischen 25 und 35 Jahre alt. Es soll ein Motto geben.  
Erstelle mir einen vollständigen Plan mit Motto-Vorschlag, Einkaufsliste, Zeitplan und Deko-Ideen.

Probier alle zehn aus. Achte darauf, wie sich die Antworten verändern, je detaillierter dein Prompt wird. Bei Prompt 1 bis 3 bekommst du kurze, allgemeine Antworten. Bei Prompt 9 und 10 bekommst du ausführliche, strukturierte Ergebnisse.

## Was ein Prompt ist – und was er nicht ist

Ein Prompt ist **kein Google-Suchbegriff**. Das ist einer der häufigsten Fehler, den Anfänger machen. Bei Google tippst du “beste Restaurants Berlin” ein und bekommst eine Liste von Webseiten. Bei einem LLM kannst du viel mehr tun:

```
Ich suche ein Restaurant in Berlin-Mitte für ein Geschäftsessen.  
Es sollte gehobene Küche sein, aber nicht zu steif. Mein Gast ist Vegetarier. Budget: 80-120 Euro pro Person. Schlage mir 3 Optionen vor und erkläre, warum sie passen.
```

Das hier würde bei Google nicht funktionieren. Bei einem LLM schon. Weil du dem Modell Kontext, Einschränkungen und ein klares Ziel gibst.

Ein Prompt ist auch **kein Befehl an einen dummen Computer**. Du musst nicht in einer bestimmten Syntax schreiben. Du musst keine speziellen Kommandos kennen. Schreib einfach, was du willst – als würdest du mit einem sehr geduldigen, sehr belesebenen Assistenten reden.

Und ein Prompt ist **kein einmaliger Versuch**. Wenn die Antwort nicht passt, schreib eine Folgenachricht. “Das ist mir zu technisch, erkläre es einfacher.” Oder “Gib mir mehr Details zu Punkt 3.” Das Gespräch geht weiter. Genau das macht LLMs so mächtig – du kannst nachbohren, korrigieren und verfeinern.

## Das Wichtigste aus diesem Kapitel

- Ein Prompt ist deine Eingabe an ein LLM – Frage, Anweisung oder beides
  - Je mehr Kontext und Details du gibst, desto besser die Antwort
  - Prompts sind kein Google-Suchbegriff – du kannst viel komplexere Anfragen stellen
  - Der erste Prompt muss nicht perfekt sein – du kannst immer nachbessern
-

# Übung

## **Schreibe 5 eigene Prompts und beobachte die Ergebnisse.**

Denk an Dinge, die dich in deinem Alltag beschäftigen. Dein Job, dein Hobby, dein nächster Urlaub, ein Problem, das du gerade hast. Schreib zu jedem Thema einen Prompt und teste ihn.

Achte besonders auf:

- Wie ausführlich ist die Antwort?
- Entspricht sie dem, was du erwartet hast?
- Was hättest du anders formulieren können?

Schreib dir die Ergebnisse auf. In den nächsten Kapiteln wirst du lernen, wie du diese Prompts systematisch verbessern kannst.

# Kapitel 4: Anatomie eines guten Prompts

---

Du hast jetzt deine ersten Prompts geschrieben. Manche haben gut funktioniert, andere weniger. Aber woran liegt das eigentlich? Was macht einen Prompt gut – und was macht ihn schlecht?

In diesem Kapitel zerlege ich gute Prompts in ihre Einzelteile. Danach wirst du verstehen, warum manche Prompts goldene Ergebnisse liefern und andere nur Standardbrei.

## Die 5 Bausteine eines guten Prompts

Ich hab mir angewöhnt, jeden Prompt gedanklich in fünf Bausteine aufzuteilen. Nicht jeder Prompt braucht alle fünf – aber je mehr du davon verwendest, desto präziser wird das Ergebnis.

### 1. Die Aufgabe (Was soll die KI tun?)

Das ist der offensichtlichste Teil, aber viele Leute machen es sich hier zu einfach. “Schreib mir was über Marketing” ist eine Aufgabe. Aber eine miese.

**Vage:** “Schreib mir was über Marketing.”

**Besser:** “Schreib mir eine Einleitung für einen Blogartikel über E-Mail-Marketing für kleine Unternehmen.”

**Noch besser:** “Schreib mir eine Einleitung (maximal 100 Wörter) für einen Blogartikel über E-Mail-Marketing für kleine Unternehmen mit weniger als 10 Mitarbeitern.”

Siehst du, wie jede Version spezifischer wird? Spezifisch schlägt vage. Immer.

## 2. Der Kontext (Was muss die KI wissen?)

Kontext ist das, was die meisten Anfänger weglassen – und was den größten Unterschied macht. Das Modell weiß nicht, wer du bist, was du schon probiert hast, oder warum du die Frage stellst.

### Ohne Kontext:

```
Wie schreibe ich eine Bewerbung?
```

### Mit Kontext:

```
Ich bin 28, habe 3 Jahre Erfahrung als Grafikdesigner in  
einer Agentur  
und bewerbe mich jetzt bei einem Tech-Startup als UI/UX De-  
signer.  
Wie sollte ich mein Anschreiben aufbauen?
```

Beim ersten Prompt bekommst du eine generische Antwort, die für jeden und niemanden passt. Beim zweiten bekommst du eine maßgeschneiderte Anleitung.

## 3. Das Format (Wie soll die Antwort aussehen?)

Du kannst dem Modell sagen, in welchem Format es antworten soll. Liste? Tabelle? Fließtext? Aufzählung mit Nummern? Bullet Points? Kurz oder lang?

Gib mir 5 Ideen für ein Firmenevent.  
Format: Nummerierte Liste. Zu jeder Idee ein Satz Beschreibung und eine grobe Kostenschätzung.

Ohne Formatangabe entscheidet das Modell selbst – und das Ergebnis passt dann oft nicht zu dem, was du brauchst.

## 4. Der Ton (Wie soll es klingen?)

Formell oder locker? Akademisch oder umgangssprachlich? Ernst oder humorvoll? Das alles kannst du steuern.

Erkläre Photosynthese. Schreib so, als würdest du es einem 10-Jährigen erklären.

vs.

Erkläre Photosynthese auf dem Niveau einer Biologie-Vorlesung im ersten Semester.

Gleiche Frage, komplett andere Antwort. Der Ton macht's.

## 5. Einschränkungen (Was soll die KI NICHT tun?)

Manchmal ist es genauso wichtig zu sagen, was du nicht willst.

Erkläre mir, wie eine Blockchain funktioniert.  
Verwende keine technischen Fachbegriffe.  
Keine Analogien mit Banken oder Geld.  
Maximal 200 Wörter.

Einschränkungen sind wie Leitplanken – sie halten die Antwort auf der Spur.

## Vorher/Nachher: 5 Prompts im Vergleich

Jetzt wird's konkret. Ich zeige dir fünf Prompts in der "vorher"-Version (wie sie die meisten Leute schreiben) und in der "nachher"-Version (mit den 5 Bausteinen).

### Beispiel 1: Rezept

#### Vorher:

Gib mir ein Rezept.

*Ergebnis: Irgendein zufälliges Rezept. Vielleicht Spaghetti Bolognese, vielleicht Sushi.*

#### Nachher:

Gib mir ein vegetarisches Rezept für ein Abendessen unter der Woche.  
Es sollte maximal 30 Minuten dauern und mit Zutaten funktionieren, die man in jedem Supermarkt bekommt. Für 2 Personen.  
Format: Zutatenliste + Schritt-für-Schritt-Anleitung.

*Ergebnis: Ein passendes Rezept, das du tatsächlich nachkochen kannst.*

### Beispiel 2: E-Mail

#### Vorher:

Schreib mir eine E-Mail an meinen Chef.

#### Nachher:

Schreib mir eine höfliche, aber bestimmte E-Mail an meinen Vorgesetzten.  
Ich möchte um ein Gespräch über eine Gehaltserhöhung bitten.  
Ich bin seit 2 Jahren in der Firma und habe zuletzt ein großes Projekt erfolgreich abgeschlossen. Ton: professionell, aber nicht unterwürfig.  
Länge: maximal 150 Wörter.

### Beispiel 3: Lernen

#### Vorher:

Erkläre mir Python.

#### Nachher:

Ich möchte Python lernen und habe keine Programmiererfahrung.  
Erstelle mir einen Lernplan für die ersten 4 Wochen.  
Pro Woche: 1 Thema, 1 kurze Erklärung, 1 praktische Übung.  
Halte es einfach – keine fortgeschrittenen Konzepte.

### Beispiel 4: Brainstorming

#### Vorher:

Gib mir Geschäftsideen.

#### Nachher:

Ich lebe in einer deutschen Kleinstadt (30.000 Einwohner), habe ein Budget von 10.000 Euro und kann 20 Stunden pro Woche investieren. Gib mir 5 realistische Geschäftsideen, die ich nebenberuflich starten könnte. Zu jeder Idee: Was ist es, warum könnte es funktionieren, was ist das Risiko?

## Beispiel 5: Texterstellung

### Vorher:

Schreib mir einen Social-Media-Post.

### Nachher:

Schreib mir einen LinkedIn-Post über die Wichtigkeit von Weiterbildung im Bereich KI. Zielgruppe: Berufstätige zwischen 30-50. Ton: motivierend, aber nicht aufdringlich. Kein Corporate-Sprech. Länge: 100-150 Wörter. Inklusive einem konkreten Call-to-Action am Ende.

## Spezifisch vs. Vage – Warum der Unterschied so groß ist

Ich bekomme oft die Frage: “Muss ich wirklich so viele Details angeben? Das Modell ist doch intelligent genug, um zu verstehen, was ich meine.”

Die Antwort ist: Nein, ist es nicht. Nicht weil es dumm wäre, sondern weil es dein Gedankenleser nicht ist.

Wenn du in ein Taxi steigst und sagst “Fahren Sie los”, dann fährt der Fahrer irgendwohin. Wenn du sagst “Bitte zum Hauptbahnhof, und nehmen Sie nicht die Autobahn, da ist Stau”, dann kommst du an, wo du hinwillst.

LLMs funktionieren genauso. Sie brauchen klare Anweisungen. Nicht weil sie bockig sind, sondern weil sie *alles* können – und deswegen nicht wissen, was du gerade willst, wenn du es nicht sagst.

Heißt das, dass jeder Prompt ein Roman sein muss? Nein. Für eine einfache Frage reicht eine einfache Frage. “Was ist die Hauptstadt von Australien?” braucht keinen Kontext. Aber für alles, was über eine Faktenfrage hinausgeht, gilt: Mehr Details = bessere Ergebnisse.

## Eine Faustregel, die mir immer hilft

Bevor ich einen Prompt schreibe, stelle ich mir diese Frage:

*“Wenn ich das einem sehr kompetenten Praktikanten am ersten Tag sagen würde – hätte er genug Informationen, um genau das zu liefern, was ich will?”*

Wenn die Antwort nein ist, fehlt was. Dann füge ich Kontext, Format oder Einschränkungen hinzu. Wenn die Antwort ja ist, drücke ich Enter.

## Das Wichtigste aus diesem Kapitel

- Gute Prompts haben bis zu 5 Bausteine: Aufgabe, Kontext, Format, Ton, Einschränkungen
- Nicht jeder Prompt braucht alle 5 – aber mehr ist meistens besser
- Spezifische Prompts schlagen vage Prompts in jeder Kategorie
- Stell dir die Praktikanten-Frage: Hat das Modell genug Info, um zu liefern?

# Übung

## **Verbessere diese 5 schlechten Prompts.**

Nimm jeden der folgenden Prompts und schreibe eine verbesserte Version – mit so vielen der 5 Bausteine wie möglich.

1. “Schreib mir einen Text.”
2. “Was soll ich essen?”
3. “Hilf mir bei meiner Präsentation.”
4. “Erkläre mir Wirtschaft.”
5. “Schreib mir eine Geschichte.”

Teste sowohl die Original-Version als auch deine verbesserte Version in einem LLM deiner Wahl. Vergleiche die Ergebnisse. Du wirst den Unterschied sofort sehen.

# Kapitel 5: Die häufigsten Fehler beim Prompting

---

Jeder macht Fehler. Ich auch. Als ich angefangen habe, mit LLMs zu arbeiten, habe ich Prompts geschrieben, für die ich mich heute schämen würde. Aber genau aus diesen Fehlern habe ich am meisten gelernt.

In diesem Kapitel zeige ich dir die Fehler, die fast jeder am Anfang macht. Wenn du die vermeidest, bist du schon weiter als 80% aller LLM-Nutzer.

## Fehler 1: Zu vage sein

Das ist der Klassiker. Der häufigste Fehler überhaupt.

### Schlecht:

Hilf mir mit meinem Projekt.

Was für ein Projekt? Schulprojekt? Softwareprojekt? Heimwerkerprojekt? In welcher Phase? Was genau brauchst du? Das Modell kann hellsehen? Kann es nicht.

### Besser:

Ich arbeite an einer PowerPoint-Präsentation für einen Kunden. Das Thema ist "Digitale Transformation im Mittelstand". Die Präsentation soll 15 Minuten dauern. Ich brauche eine Gliederung mit 8-10 Folien. Zu jeder Folie: Titel und 3 Stichpunkte.

## Fehler 2: Zu viel auf einmal wollen

Das Gegenteil von zu vage. Manche Leute packen alles in einen einzigen Prompt und erwarten ein perfektes Ergebnis.

### Schlecht:

Schreib mir einen kompletten Businessplan für ein Café. Mit Finanzplanung, Marktanalyse, Wettbewerbsanalyse, Personalplanung, Marketingstrategie, Standortanalyse und Risikoanalyse. Alles auf Deutsch. Mindestens 5000 Wörter. Mit Diagrammen.

Das ist nicht ein Prompt – das sind zehn. Und das Ergebnis wird in keinem Bereich wirklich gut sein, weil das Modell versucht, alles oberflächlich abzuhandeln.

**Besser:** Teile es auf. Mach erst die Gliederung. Dann die Marktanalyse. Dann die Finanzplanung. Schritt für Schritt.

Erstelle eine Gliederung für einen Businessplan für ein Café in einer deutschen Universitätsstadt. Nenne die wichtigsten Abschnitte und was jeweils enthalten sein sollte.

Und dann im nächsten Prompt:

```
Jetzt schreib mir den Abschnitt "Marktanalyse" aus. [...]
```

## Fehler 3: Keinen Kontext geben

Das hatten wir schon in Kapitel 4, aber es ist so wichtig, dass ich es hier nochmal betone. Ohne Kontext rät das Modell. Und meistens rät es falsch.

### Schlecht:

```
Wie führe ich ein schwieriges Gespräch?
```

### Besser:

```
Ich muss einem Mitarbeiter sagen, dass seine Leistung in den letzten 3 Monaten nachgelassen hat. Ich bin sein direkter Vorgesetzter. Die Firma ist ein kleines Tech-Startup mit 15 Leuten, der Ton ist normalerweise locker. Wie führe ich dieses Gespräch, ohne dass es eskaliert?
```

Der Kontext macht aus einer generischen Antwort eine, die du tatsächlich verwenden kannst.

## Fehler 4: Ergebnisse nicht hinterfragen

Das ist kein Prompt-Fehler im engeren Sinn, aber es gehört hierher. Viele Leute nehmen die erste Antwort eines LLMs für bare Münze. Das ist gefährlich.

LLMs halluzinieren. Das ist kein Bug, das ist ein Feature – wenn man so will. Das Modell ist darauf trainiert, plausibel klingende Antworten zu generieren. Plausibel klingend und korrekt sind aber zwei verschiedene Dinge.

Hier ein paar Warnsignale, die ich über die Zeit gelernt habe:

- **Zahlen und Statistiken:** LLMs erfinden Zahlen. “Laut einer Studie von Harvard...” – diese Studie existiert möglicherweise nicht. Prüfe Zahlen immer.
- **Zitate:** Das Modell kann dir ein wunderbares Zitat von Einstein liefern, das Einstein nie gesagt hat. Verifizieren.
- **Aktuelle Informationen:** LLMs haben einen Wissens-Stichtag. Sie wissen nicht, was gestern passiert ist (es sei denn, sie haben Internetzugang wie Gemini).
- **Quellenangaben:** Wenn das Modell Links oder Quellenangaben nennt – prüfe, ob sie existieren. Häufig sind sie komplett erfunden.

**Faustregel:** Nutze LLMs als Startpunkt, nicht als Endpunkt. Sie liefern dir einen Entwurf, eine Idee, eine Richtung. Die Überprüfung liegt bei dir.

## Fehler 5: Nicht iterieren

“Ich hab’s einmal probiert und es war schlecht. KI bringt nichts.”

Diesen Satz höre ich ständig. Und er zeigt ein fundamentales Missverständnis. Prompt Engineering ist ein Prozess, kein Knopfdruck. Der erste Versuch ist selten perfekt – genau wie der erste Entwurf eines Textes selten druckreif ist.

Wenn die Antwort nicht passt, sag dem Modell, was nicht stimmt:

Das war zu allgemein. Geh mehr ins Detail bei Punkt 3.

Oder:

Der Ton ist zu formell. Schreib das lockerer, als würdest du es einem Freund erklären.

Oder:

Die Antwort ist gut, aber zu lang. Kürze sie auf die Hälfte.

Das ist kein Zeichen, dass du schlecht promptest. Das ist normaler Workflow.

## Fehler 6: Die falschen Erwartungen haben

Ein LLM ist kein Mensch. Es hat kein Weltwissen im menschlichen Sinn. Es hat Muster gelernt. Deswegen gibt es Dinge, die es gut kann, und Dinge, die es schlecht kann.

### **Gut darin:**

- Texte schreiben und umformulieren
- Zusammenfassungen erstellen
- Brainstorming und Ideengenerierung
- Strukturierung und Gliederung
- Code schreiben und erklären
- Sprachen übersetzen
- Konzepte erklären

### **Schlecht darin:**

- Mathematik (ja, wirklich – einfache Rechnungen gehen schief)
- Verlässliche Fakten liefern (insbesondere aktuelle)
- Emotionale Intelligenz (es simuliert Empathie, hat aber keine)

- Kreative Originalität (es kombiniert Bekanntes neu, erfindet aber nichts wirklich Neues)
- Logische Rätsel und Brainteaser (oft überraschend schlecht)

Wenn du weißt, wo die Stärken und Schwächen liegen, kannst du die KI viel gezielter einsetzen.

## **Fehler 7: “Bitte” und “Danke” – bringt das was?**

Interessante Frage, die mir oft gestellt wird. Muss man höflich zu einer KI sein?

Kurze Antwort: Es schadet nicht, bringt aber technisch gesehen wenig. Das Modell verarbeitet “Bitte erkläre mir X” und “Erkläre mir X” praktisch gleich. Höflichkeit verbraucht Token, ändert aber selten die Qualität der Antwort.

Trotzdem sage ich oft “Danke”. Nicht weil die KI es braucht, sondern weil es mir hilft, im Gesprächsmodus zu bleiben. Wenn ich die KI wie eine Person behandle, schreibe ich automatisch klarere, besser formulierte Prompts. Außerdem – wer weiß, vielleicht liest die KI eines Tages dieses Buch und erinnert sich daran, dass ich nett war.

Das war ein Witz. LLMs erinnern sich an nichts. Aber höflich sein schadet nie.

## **Das Wichtigste aus diesem Kapitel**

- Zu vage und zu viel auf einmal sind die häufigsten Fehler
- Kontext macht den Unterschied zwischen generischer und nützlicher Antwort
- LLMs halluzinieren – hinterfrage Fakten, Zahlen und Quellen immer
- Der erste Prompt muss nicht perfekt sein – iteriere

- Kenne die Stärken und Schwächen von LLMs
  - Höflichkeit schadet nicht, ist aber technisch irrelevant
- 

## Übung

### Identifiziere die Fehler in diesen 10 Prompts.

Lies jeden Prompt und schreibe auf, welcher Fehler gemacht wird. Dann schreibe eine verbesserte Version.

1. "Hilf mir."
2. "Schreib mir alles über den Zweiten Weltkrieg in 500 Wörtern mit Quellenangaben."
3. "Was ist besser?"
4. "Du bist ein Experte. Erkläre mir alles."
5. "Schreib mir eine Bewerbung." (Ohne weitere Infos)
6. "Was ist der aktuelle Bitcoin-Kurs?" (An ein Modell ohne Internetzugang)
7. "Gib mir 100 Ideen für Social-Media-Posts, mit Grafik-Beschreibung, Hashtags, optimaler Posting-Zeit und Zielgruppenanalyse für jeden Post."
8. "Erzähl mir was Lustiges."
9. "Berechne  $17,3 \times 24,8 + \sin(45^\circ) / \log(128)$ " (An ein Sprachmodell)
10. "Schreib mir einen Text. Aber nicht zu lang. Und nicht zu kurz. Und lustig, aber seriös."

Bei manchen ist es offensichtlich. Bei anderen musst du ein bisschen nachdenken. Genau das ist der Punkt – nach diesem Kapitel wirst du solche Fehler instinktiv erkennen.

# Kapitel 6: Kontext ist alles

---

Wenn es ein einziges Wort gibt, das den Unterschied zwischen einem mittelmäßigen und einem großartigen Prompt ausmacht, dann ist es: Kontext.

Ich hab das in den vorherigen Kapiteln schon mehrfach erwähnt. Aber jetzt gehen wir richtig in die Tiefe. Denn Kontext geben ist eine Kunst, die man lernen kann – und die sich sofort auszahlt.

## Warum Kontext so wichtig ist

Ein LLM hat kein Gedächtnis über Sitzungen hinweg. Es weiß nicht, wer du bist. Es weiß nicht, was du gestern gefragt hast (es sei denn, es steht noch im selben Chat). Es weiß nicht, ob du 16 oder 60 bist, ob du Arzt oder Bäcker bist, ob du Deutsch als Muttersprache sprichst oder nicht.

Wenn du fragst “Wie investiere ich mein Geld?”, dann weiß das Modell nicht:

- Wie viel Geld du hast
- Wie alt du bist
- Wie risikobereit du bist
- Ob du schon Erfahrung mit Investments hast
- In welchem Land du lebst (Steuerrecht!)
- Was dein Ziel ist (Altersvorsorge? Hausbau? Urlaub?)

Ohne diese Informationen bekommst du eine Standardantwort, die für niemanden wirklich passt. Mit diesen Informationen bekommst du eine Antwort, die tatsächlich hilfreich ist.

# Die 5 Kontext-Kategorien

Ich teile Kontext in fünf Kategorien ein. Du musst nicht immer alle verwenden – aber es hilft, sie im Kopf zu haben.

## 1. Wer bist du?

Deine Rolle, dein Hintergrund, deine Erfahrung.

Ich bin Grundschullehrerin mit 10 Jahren Berufserfahrung...

Ich bin Student im 3. Semester Informatik...

Ich bin Geschäftsführer eines mittelständischen Unternehmens mit 50 Mitarbeitern...

## 2. Was ist die Situation?

Der Hintergrund, die Umstände, das Problem.

...und bereite eine Elternabend-Präsentation vor, bei der es um den Einsatz von Tablets im Unterricht geht. Manche Eltern sind skeptisch.

...und muss morgen eine Klausur in Algorithmen und Datenstrukturen schreiben. Ich verstehe Rekursion noch nicht richtig.

### 3. Wer ist die Zielgruppe?

Für wen ist das Ergebnis bestimmt?

```
Die Zielgruppe sind Eltern ohne technisches Vorwissen.
```

```
Der Text ist für mein LinkedIn-Netzwerk - hauptsächlich HR-  
Manager  
und Recruiter.
```

```
Das soll ein internes Memo für mein Team sein.  
Die kennen den Kontext schon, ich muss nicht bei null an-  
fangen.
```

### 4. Was hast du schon versucht?

Das ist ein Kontext, den fast niemand gibt – aber der unglaublich hilfreich ist.

```
Ich habe schon versucht, das Problem mit einer For-Schleife  
zu lösen,  
aber ich bekomme einen IndexError. Hier ist mein Code:  
[...]
```

```
Ich habe schon 3 verschiedene Entwürfe für die Einleitung  
geschrieben,  
aber alle klingen zu steif. Hier ist der letzte Versuch:  
[...]
```

Wenn das Modell weiß, was nicht funktioniert hat, kann es dir bessere Alternativen vorschlagen.

## 5. Was ist das Ziel?

Was willst du mit dem Ergebnis machen? Wohin soll die Reise gehen?

Ich brauche den Text für eine Bewerbung bei einem DAX-Konzern.

Die Analyse soll als Entscheidungsgrundlage dienen, ob wir in einen neuen Markt eintreten.

Ich will den Prompt später als Template wiederverwenden - er sollte also allgemein genug sein, dass ich nur die Details austauschen muss.

## Kontext in Aktion: Vorher und Nachher

Lass mich dir drei Beispiele zeigen, die den Unterschied deutlich machen.

### Beispiel 1: Fitness

#### Ohne Kontext:

Erstelle mir einen Trainingsplan.

*Ergebnis: Ein generischer Plan mit Kniebeugen, Liegestützen, Joggen – für irgendjemanden.*

#### Mit Kontext:

Erstelle mir einen Trainingsplan. Ich bin 34, männlich, 185cm, 92kg.  
Ich war die letzten 2 Jahre inaktiv, hatte aber vorher Erfahrung mit Krafttraining. Ich habe 3 Tage pro Woche jeweils 60 Minuten Zeit.  
Zugang zu einem Fitnessstudio. Ziel: Wieder fit werden und 5kg abnehmen in 3 Monaten. Ich habe leichte Knieprobleme – keine Sprungübungen bitte.

*Ergebnis: Ein maßgeschneiderter Plan, der deine Einschränkungen berücksichtigt.*

## Beispiel 2: Texterstellung

### Ohne Kontext:

Schreib mir eine Produktbeschreibung.

### Mit Kontext:

Schreib mir eine Produktbeschreibung für unseren neuen kabellosen Bluetooth-Kopfhörer "SoundFlex Pro". Zielgruppe: Pendler und Home-Office-Arbeiter, 25-45 Jahre. USP: 40 Stunden Akkulaufzeit und aktive Geräuschunterdrückung. Preis: 89 Euro. Ton: modern und einladend, aber nicht reißerisch. Für unsere Website, maximal 150 Wörter. Wir konkurrieren mit Sony und Bose – also nicht billig wirken.

## Beispiel 3: Problemlösung

### Ohne Kontext:

Mein Team funktioniert nicht. Was soll ich tun?

### **Mit Kontext:**

Ich leite ein 6-köpfiges Marketing-Team. Seit ein neuer Mitarbeiter vor 3 Monaten dazugekommen ist, gibt es Spannungen. Zwei ältere Teammitglieder fühlen sich übergangen, weil der Neue viele Aufgaben direkt von der Geschäftsführung bekommt. Die Stimmung ist merklich schlechter geworden, Deadlines werden knapp eingehalten. Ich möchte die Situation lösen, ohne jemanden bloßzustellen. Was sind meine Optionen?

Du siehst den Unterschied. Beim ersten Prompt bekommst du “Versuchen Sie, offene Kommunikation zu fördern” und ähnlichen Standardkram. Beim zweiten Prompt bekommst du eine Analyse der spezifischen Situation mit konkreten Handlungsempfehlungen.

## **Wie viel Kontext ist zu viel?**

Gute Frage. Kann man es übertreiben? Theoretisch ja. Wenn dein Prompt länger ist als die Antwort, die du erwartest, machst du wahrscheinlich etwas falsch.

Aber in der Praxis erlebe ich das Gegenteil: Die meisten Leute geben zu wenig Kontext, nicht zu viel. Mein Rat: Lieber eine Information zu viel als eine zu wenig. Das Modell ignoriert, was es nicht braucht. Aber es kann nicht erraten, was du nicht sagst.

Eine Ausnahme: Sensible Daten. Gib keine Passwörter, Kreditkartennummern, vertrauliche Firmendaten oder persönliche Informationen ein, die du nicht in der Cloud haben willst. Die Daten werden von den Anbietern verarbeitet – wie genau, steht in deren Datenschutzerklärung. Im Zweifel: anonymisiere.

## Der “Goldene Satz” – ein Trick, den ich oft benutze

Wenn ich nicht weiß, wie viel Kontext ich geben soll, starte ich mit einem “Goldenen Satz”. Das ist ein einzelner Satz, der die wichtigste Kontextinformation enthält.

Ich bin Lehrer an einer Berufsschule und unterrichte Jugendliche zwischen 16 und 18 Jahren.

Dieser eine Satz verändert jede Antwort. Das Modell passt automatisch Sprache, Komplexität und Beispiele an.

Probier es aus: Stell eine beliebige Frage einmal ohne und einmal mit dem Goldenen Satz davor. Der Unterschied ist oft verblüffend.

## Das Wichtigste aus diesem Kapitel

- Kontext ist der wichtigste Baustein eines guten Prompts
- 5 Kontext-Kategorien: Wer bist du, Situation, Zielgruppe, was hast du versucht, was ist das Ziel
- Lieber zu viel Kontext als zu wenig
- Sensible Daten anonymisieren oder weglassen
- Der “Goldene Satz” – ein Satz Kontext kann alles verändern

# Übung

## **Nimm 3 einfache Prompts und füge Kontext hinzu.**

Hier sind drei bewusst kontextlose Prompts:

1. “Wie lerne ich besser?”
2. “Schreib mir einen Brief.”
3. “Was soll ich beruflich machen?”

Für jeden Prompt:

- Überlege dir eine konkrete Person mit einer konkreten Situation
- Füge mindestens 3 der 5 Kontext-Kategorien hinzu
- Teste beide Versionen (ohne und mit Kontext) in einem LLM
- Vergleiche die Ergebnisse

Bonus: Versuche den “Goldenen Satz” – finde für jedes Szenario den einen Satz, der den größten Unterschied macht.

# Kapitel 7: Rollen und Personas

## – Die KI in eine Rolle schlüpfen lassen

---

Stell dir vor, du fragst jemanden nach einem Trainingsplan. Wenn du einen zufälligen Passanten fragst, bekommst du eine andere Antwort als von einem Personal Trainer. Gleiche Frage, völlig anderes Ergebnis.

Bei LLMs funktioniert das genauso. Und das Beste: Du kannst der KI sagen, wer sie sein soll.

### “Du bist ein...” – So weist du Rollen zu

Der einfachste Weg, eine Rolle zuzuweisen, ist ein Satz am Anfang deines Prompts:

```
Du bist ein erfahrener Steuerberater mit 20 Jahren Berufserfahrung  
in Deutschland. Ein Kunde fragt dich: [...]
```

```
Du bist eine freundliche Grundschullehrerin. Erkläre einem  
8-Jährigen,  
warum es Jahreszeiten gibt.
```

Du bist ein kritischer Lektor, der kein Blatt vor den Mund nimmt.  
Lies folgenden Text und sag mir ehrlich, was daran schlecht ist: [...]

Was passiert hier? Du gibst dem Modell einen Rahmen. Es weiß jetzt, aus welcher Perspektive es antworten soll, welchen Wortschatz es verwenden soll und welches Niveau angemessen ist.

## Warum Rollen funktionieren

Erinnere dich an Kapitel 2: LLMs haben während des Trainings Millionen von Texten gelesen, die von verschiedenen “Typen” von Menschen geschrieben wurden. Ärzte schreiben anders als Comedians. Professoren formulieren anders als Blogger.

Wenn du dem Modell sagst “Du bist ein erfahrener Koch”, dann greift es auf die Muster zurück, die es von Köchen, Kochbüchern und Food-Bloggern gelernt hat. Es verwendet Fachbegriffe wie “blanchieren” statt “kurz in kochendes Wasser tun”. Es strukturiert die Antwort wie ein Rezept. Es denkt an Dinge wie Garzeiten und Würzung.

Das ist kein Trick. Es ist die Art, wie diese Modelle funktionieren – und du nutzt es zu deinem Vorteil.

## 7 Rollen, die besonders gut funktionieren

Hier sind Rollen, die ich regelmäßig benutze und die konstant gute Ergebnisse liefern:

## 1. Der Experte

Du bist ein Experte für [Thema] mit [X] Jahren Erfahrung.

Funktioniert für fast alles. Steuerrecht, Gartenarbeit, Autoversicherungen. Die KI antwortet fundierter, wenn sie denkt, sie sei ein Experte.

## 2. Der Lehrer

Du bist ein geduldiger Lehrer, der Anfängern komplexe Themen in einfachen Worten erklärt.

Perfekt, wenn du etwas lernen willst. Die Antworten sind didaktisch aufgebaut, mit Beispielen und ohne Fachjargon.

## 3. Der kritische Reviewer

Du bist ein strenger aber fairer Kritiker. Analysiere [Text/Idee/Plan] und zeige Schwächen auf, die ich übersehen habe.

Einer meiner Favoriten. Die meisten Leute benutzen KI nur zum Bestätigen. Aber sie ist am nützlichsten, wenn sie dir sagt, was nicht funktioniert.

## 4. Der Berater

Du bist ein erfahrener Business-Berater. Dein Kunde kommt zu dir mit folgendem Problem: [...]

Die KI gibt dir dann keine Schulbuch-Antwort, sondern eine praxisorientierte Empfehlung.

## 5. Der Übersetzer/Vereinfacher

Du bist ein Übersetzer, der komplizierte Fachtexte in einfache Alltagssprache übersetzt. Kein Wort mit mehr als 3 Silben.

Klingt lustig, funktioniert aber hervorragend, wenn du einen komplizierten Text für Laien aufbereiten musst.

## 6. Der kreative Partner

Du bist ein kreativer Brainstorming-Partner. Wirf mir ungewöhnliche Ideen zu, auch wenn sie erstmal verrückt klingen. Wir filtern später.

Das gibt der KI die Erlaubnis, wild zu denken, statt die sicherste Antwort zu geben.

## 7. Der Devil's Advocate

Du bist ein Devil's Advocate. Finde Gegenargumente zu meiner Position, auch wenn du denkst, dass ich Recht habe. Ich brauche die andere Perspektive.

Unglaublich nützlich für Entscheidungsfindung und Argumentationsschärfung.

## Personas: Rollen mit Persönlichkeit

Eine Persona geht noch einen Schritt weiter als eine Rolle. Statt nur "Experte" zu sagen, gibst du der KI eine komplette Persönlichkeit.

### Rolle:

Du bist ein Marketing-Experte.

### Persona:

Du bist Sarah, 42, Marketingleiterin bei einem mittelständischen Unternehmen in Hamburg. Du arbeitest seit 15 Jahren im B2B-Marketing, hast schon drei Firmen beraten und bist bekannt für deine direkte, no-bullshit Kommunikation. Du magst keine Buzzwords und sagst lieber "das funktioniert nicht" als "da gibt es Optimierungspotenzial".

Je detaillierter die Persona, desto konsistenter und charakturvoller die Antworten. Das ist besonders nützlich, wenn du:

- Fiktive Interviews für Marktforschung simulierst
- Einen konsistenten Charakter für eine Geschichte brauchst
- Verschiedene Perspektiven auf ein Problem haben willst

## Mehrere Perspektiven einholen

Ein richtig starker Move: Lass die KI aus verschiedenen Rollen auf dieselbe Frage antworten.

Ich überlege, mich selbstständig zu machen. Gib mir Rat aus drei

verschiedenen Perspektiven:

1. Ein erfolgreicher Unternehmer, der seit 10 Jahren selbstständig ist
2. Ein Steuerberater
3. Ein Psychologe, der sich auf Stress und Work-Life-Balance spezialisiert hat

Jede Perspektive: 3-4 Sätze, klar getrennt.

Du bekommst drei verschiedene Blickwinkel in einer Antwort. Der Unternehmer sagt dir, worauf es ankommt. Der Steuerberater warnt dich vor finanziellen Fallstricken. Der Psychologe fragt, ob du emotional bereit bist.

Das ist Beratung auf einem Niveau, für das du sonst drei verschiedene Termine brauchst.

## Rollen kombinieren mit anderen Bausteinen

Rollen funktionieren am besten, wenn du sie mit den anderen Bausteinen aus Kapitel 4 kombinierst:

Du bist ein erfahrener UX-Designer (ROLLE).

Ich baue eine App für Senioren, die ihre Medikamente verwalten sollen (KONTEXT).

Erstelle eine Liste von 5 Design-Prinzipien, die ich beachten sollte (AUFGABE).

Jedes Prinzip: Überschrift + 2 Sätze Erklärung + ein konkretes Beispiel (FORMAT).

Vermeide Fachjargon – mein Chef ist kein Designer (EINSCHRÄNKUNG).

Siehst du, wie alles zusammenspielt? Rolle, Kontext, Aufgabe, Format, Einschränkung – die fünf Bausteine aus Kapitel 4, hier in Aktion.

## Typische Fehler bei Rollen

### Fehler 1: Zu generisch

Du bist schlau. Hilf mir.

Das ist keine Rolle, das ist ein Wunsch. Sei spezifisch.

### Fehler 2: Widersprüchliche Rollen

Du bist ein konservativer Finanzberater, der riskante Investments empfiehlt.

Das Modell wird verwirrt sein und inkonsistent antworten.

### Fehler 3: Die Rolle vergessen

Wenn du in einem längeren Chat die Rolle nur am Anfang zuweist, kann das Modell sie nach einigen Nachrichten "vergessen". Erinnere es gelegentlich: "Bleib bitte in deiner Rolle als Steuerberater."

## Das Wichtigste aus diesem Kapitel

- Rollen geben dem Modell einen Rahmen für Perspektive und Fachsprache
- "Du bist ein..." ist der einfachste Einstieg
- Personas (mit Persönlichkeit) liefern konsistentere Ergebnisse als generische Rollen
- Mehrere Perspektiven in einem Prompt liefern vielseitigere Antworten
- Rollen mit den anderen 4 Bausteinen kombinieren für beste Ergebnisse

## Übung

### **Erstelle 5 verschiedene Rollen-Prompts zum selben Thema.**

Wähle ein Thema, das dich interessiert. Zum Beispiel: “Soll ich auf Social Media aktiver werden?”

Jetzt schreibe 5 Prompts – immer mit einer anderen Rolle:

1. Ein Social-Media-Manager
2. Ein Psychologe
3. Ein Datenschutz-Experte
4. Ein erfolgreicher Influencer
5. Ein Großvater, der skeptisch gegenüber Technologie ist

Teste alle 5 im selben LLM und vergleiche die Antworten. Achte darauf:

- Wie unterscheidet sich die Sprache?
- Welche Argumente kommen nur in bestimmten Rollen vor?
- Welche Perspektive war am überraschendsten?

Das ist Prompt Engineering in Aktion: Durch die Rolle steuerst du nicht nur *was* die KI sagt, sondern *wie* und *aus welcher Perspektive* sie es sagt.

# Kapitel 8: Output-Formate steuern – Damit die KI liefert, was du brauchst

---

Du weißt jetzt, wie du gute Prompts schreibst, Kontext gibst und Rollen zuweist. Aber was nützt die beste Antwort, wenn sie im falschen Format kommt?

Stell dir vor, du brauchst eine Tabelle für eine Präsentation, aber die KI liefert dir einen Fließtext. Oder du willst eine kurze Zusammenfassung und bekommst einen Roman. Das passiert ständig – und es ist komplett vermeidbar.

In diesem Kapitel lernst du, wie du das Output-Format kontrollierst. Das spart dir Zeit und Nerven.

## Die wichtigsten Formate

### Listen

Listen sind das am häufigsten gewünschte Format. Es gibt zwei Varianten:

**Nummerierte Liste (wenn Reihenfolge wichtig ist):**

Gib mir die 5 wichtigsten Schritte, um ein LinkedIn-Profil zu optimieren. Nummerierte Liste.

### **Bullet Points (wenn Reihenfolge egal ist):**

Nenne mir die Vorteile von Remote-Arbeit. Aufzählung mit Stichpunkten.

## **Tabellen**

Tabellen sind Gold wert für Vergleiche.

Vergleiche ChatGPT, Claude und Gemini in einer Tabelle.  
Spalten: Name, Anbieter, Stärken, Schwächen, Kosten (kostenlose Version).

Das Modell erstellt dir eine saubere Markdown-Tabelle, die du direkt kopieren kannst.

## **Strukturierter Text mit Überschriften**

Schreib mir einen Überblick über Projektmanagement-Methoden.  
Verwende Überschriften (## für Hauptmethoden) und darunter jeweils  
3-4 Sätze Beschreibung plus ein Beispiel, wann die Methode sinnvoll ist.

## JSON (für Entwickler und Datenverarbeitung)

Erstelle eine Liste von 5 fiktiven Produkten mit Name, Preis, Kategorie und Kurzbeschreibung. Format: JSON-Array.

Ergebnis:

```
[
  {
    "name": "SmartLamp X1",
    "preis": 49.99,
    "kategorie": "Smart Home",
    "beschreibung": "LED-Lampe mit App-Steuerung und 16 Millionen Farben"
  },
  ...
]
```

## Markdown

Wenn du mit Markdown arbeitest (z.B. für GitHub, Notion oder Obsidian):

Erstelle eine Projektdokumentation in Markdown-Format. Verwende Überschriften, Code-Blöcke, Listen und eine Tabelle.

## Länge steuern

Die Länge der Antwort ist einer der einfachsten Stellschrauben – und einer der wichtigsten.

**Wortanzahl:**

Erkläre Quantencomputing. Maximal 100 Wörter.

### **Satzanzahl:**

Fasse diesen Artikel in genau 3 Sätzen zusammen.

### **Absatzanzahl:**

Schreib eine Produktbeschreibung. Genau 2 Absätze.  
Erster Absatz: Was ist das Produkt?  
Zweiter Absatz: Warum sollte man es kaufen?

### **Seitenangabe:**

Erstelle einen Bericht über KI im Bildungswesen.  
Umfang: ungefähr 2 DIN-A4-Seiten.

Ein Hinweis zur Genauigkeit: LLMs sind nicht super präzise bei Wortanzahlen. Wenn du “maximal 100 Wörter” sagst, bekommst du vielleicht 95 oder 115. Für die meisten Zwecke ist das nah genug. Wenn du es wirklich exakt brauchst, sag “Zähle die Wörter und schreibe die Anzahl am Ende.”

## **Sprache und Tonalität steuern**

Das Format umfasst nicht nur die Struktur, sondern auch den Stil.

### **Formell:**

Schreib eine E-Mail an den Vorstand. Ton: formell und professionell.  
Keine Umgangssprache, keine Emojis.

### **Locker:**

Schreib einen Instagram-Caption. Ton: locker, freundlich, leicht humorvoll. Wie ein Gespräch unter Freunden.

### **Akademisch:**

Verfasse einen Absatz im Stil einer wissenschaftlichen Arbeit.  
Verwende Passivkonstruktionen und Fachterminologie.  
Zitiere im APA-Format.

### **Kindgerecht:**

Erkläre, wie eine Batterie funktioniert.  
Schreib so, dass ein 7-Jähriger es versteht.  
Verwende einfache Wörter und ein Alltagsbeispiel.

## **Strukturierte Ausgaben erzwingen**

Manchmal brauchst du eine ganz bestimmte Struktur. Dann gib dem Modell ein Template:

Analysiere die folgende Geschäftsidee. Verwende exakt diese Struktur:

```
**Idee:** [Kurzbeschreibung in einem Satz]
**Zielgruppe:** [Wer würde das kaufen?]
**Stärken:** [3 Bullet Points]
**Schwächen:** [3 Bullet Points]
**Nächster Schritt:** [Was sollte man als erstes tun?]
**Bewertung:** [1-10, mit Begründung in einem Satz]
```

Die Geschäftsidee: Eine App, die lokale Bauernhöfe mit Restaurants verbindet.

Das Modell füllt dein Template aus. Punkt für Punkt. Das ist besonders praktisch, wenn du mehrere Dinge nach dem gleichen Schema analysieren willst – du schickst einfach das gleiche Template mit einer anderen Idee.

## Mehrere Formate in einem Prompt

Du kannst auch verschiedene Formate kombinieren:

Erstelle eine Übersicht über die 3 beliebtesten Programmiersprachen für Anfänger.

Für jede Sprache:

1. **\*\*Überschrift\*\*** mit dem Namen
2. **\*\*Kurzbeschreibung\*\*** (2 Sätze)
3. **\*\*Tabelle\*\*** mit Vor- und Nachteilen (je 3)
4. **\*\*Code-Beispiel\*\*** (Hello World)
5. **\*\*Empfehlung\*\*** in einem Satz: Für wen ist diese Sprache am besten?

## Praxis-Trick: Das Format vorab zeigen

Wenn du ein ganz bestimmtes Format im Kopf hast, zeig dem Modell ein Beispiel:

Erstelle 3 Flashcards zum Thema Fotosynthese.  
Format wie in diesem Beispiel:

```
---  
**Frage:** Was ist Fotosynthese?  
**Antwort:** Der Prozess, bei dem Pflanzen Licht in Energie  
umwandeln.  
**Merkhilfe:** Foto = Licht, Synthese = Zusammensetzen  
**Schwierigkeit:** Leicht  
---
```

Erstelle 3 weitere in diesem Format zum Thema Zellteilung.

Das Modell kopiert dein Format exakt. Das nennt man “Few-Shot Prompting” – du zeigst ein Beispiel und das Modell folgt dem Muster. Dazu mehr in Band 2.

## Das Wichtigste aus diesem Kapitel

- Du kannst Listen, Tabellen, JSON, Markdown und Fließtext anfordern
- Länge steuerst du über Wort-, Satz- oder Absatzanzahl
- Tonalität (formell, locker, akademisch) ist Teil des Formats
- Templates erzwingen eine konsistente Struktur
- Ein Beispiel zu zeigen ist der sicherste Weg zum gewünschten Format

---

## Übung

**Lass dir dasselbe Thema in 5 verschiedenen Formaten ausgeben.**

Wähle ein beliebiges Thema. Zum Beispiel: “Die Vorteile von regelmäßiger Bewegung.”

Jetzt fordere es in diesen 5 Formaten an:

1. Eine nummerierte Liste mit 7 Punkten
2. Eine Tabelle (Vorteil | Erklärung | Wie oft pro Woche)
3. Ein kurzer Absatz (maximal 80 Wörter) im akademischen Stil
4. Ein Instagram-Post (locker, mit Emoji-Vorschlägen)
5. Ein JSON-Objekt mit den Feldern: vorteil, kategorie, evidenz

Vergleiche die Ergebnisse. Achte darauf, wie das gleiche Wissen komplett anders präsentiert wird – nur durch die Format-Anweisung.

# Kapitel 9: Iteratives Prompting – Warum der erste Versuch selten reicht

---

Ich muss dir was sagen, das viele Prompt-Engineering-Anfänger nicht hören wollen: Dein erster Prompt wird fast nie perfekt sein. Und das ist völlig in Ordnung.

Die besten Ergebnisse entstehen nicht durch einen einzigen genialen Prompt. Sie entstehen durch einen Prozess: Schreiben, Ergebnis lesen, korrigieren, verfeinern, nochmal probieren. Genau wie ein Bildhauer nicht mit einem einzigen Hammerschlag eine Statue formt, formst du dein Ergebnis Schritt für Schritt.

Das nennt sich iteratives Prompting. Und es ist die vielleicht wichtigste Fähigkeit, die du in diesem Buch lernst.

## Warum der erste Prompt selten perfekt ist

Drei Gründe:

### **1. Du weißt nicht immer, was du willst**

Klingt komisch, ist aber wahr. Oft merkst du erst, wenn du die Antwort siehst, dass du eigentlich etwas anderes wolltest. Das ist kein Versagen – das ist ein normaler Denkprozess. Die erste Antwort hilft dir, dein eigenes Ziel zu schärfen.

## 2. Du kannst nicht alles im Voraus bedenken

Selbst mit den 5 Bausteinen aus Kapitel 4 vergisst du manchmal einen wichtigen Kontext. Oder du merkst, dass der Ton nicht stimmt. Oder die Länge passt nicht. Kein Problem – dafür gibt es Folgeprompts.

## 3. LLMs interpretieren anders als du denkst

Du schreibst “kurz”, aber dein “kurz” sind 3 Sätze und das “kurz” des Modells sind 3 Absätze. Oder du schreibst “professionell” und meinst “freundlich-businesslike”, aber das Modell versteht “steif und distanziert”. Missverständnisse passieren – und du korrigierst sie im Dialog.

# Die Kunst des Folge-Prompts

Ein Folge-Prompt ist dein Werkzeug, um die Antwort zu verfeinern. Hier sind die häufigsten Situationen und was du schreiben kannst:

## Die Antwort ist zu lang

Das ist zu ausführlich. Kürze es auf die Hälfte und behalte nur die wichtigsten Punkte.

Oder noch präziser:

Kürze den Text auf maximal 100 Wörter. Behalte Punkt 1, 3 und 5.  
Die anderen können weg.

## Die Antwort ist zu oberflächlich

Gut als Überblick, aber geh bei Punkt 2 und 4 mehr ins Detail.  
Ich brauche konkrete Beispiele und Zahlen.

## Der Ton stimmt nicht

Das klingt zu formell. Schreib es um – lockerer, als würdest du es einem Freund bei einem Kaffee erklären. Keine Fachbegriffe.

## Die Antwort geht in die falsche Richtung

Das geht an meiner Frage vorbei. Ich meinte nicht [X], sondern [Y].  
Lass uns nochmal von vorn anfangen: [Klarstellung]

## Du willst einen bestimmten Teil vertiefen

Punkt 3 ist sehr interessant. Kannst du dazu mehr schreiben?  
Speziell: Welche konkreten Schritte müsste ich unternehmen?

## Du willst das Format ändern

Gut, aber mach daraus eine Tabelle statt Fließtext.  
Spalten: Schritt, Beschreibung, Zeitaufwand, benötigte Ressourcen.

## Ein vollständiges Beispiel: 5 Iterationen

Lass mich dir zeigen, wie ein iterativer Prozess in der Praxis aussieht. Ich wollte neulich einen LinkedIn-Post über KI im Bildungswesen schreiben.

### Iteration 1:

Schreib mir einen LinkedIn-Post über KI im Bildungswesen.

*Ergebnis: Generisch, lang, voller Buzzwords. Nicht brauchbar.*

### Iteration 2:

Das ist mir zu generisch und zu lang. Ich möchte einen konkreten  
Aspekt: Wie Lehrer KI als Werkzeug nutzen können. Maximal  
150 Wörter.  
Kein Buzzword-Bingo.

*Ergebnis: Besser. Fokussierter. Aber immer noch etwas langweilig.*

### Iteration 3:

Der Inhalt stimmt jetzt, aber der Einstieg ist schwach.  
Starte mit  
einer konkreten Szene oder einem Beispiel, nicht mit einer  
allgemeinen  
Aussage. Und der Schluss braucht einen konkreten Call-to-  
Action.

*Ergebnis: Guter Einstieg, besserer Schluss. Fast da.*

### Iteration 4:

Gut! Aber Satz 3 ist zu lang – teile ihn in zwei kürzere Sätze auf.  
Und ersetze "transformieren" durch ein normaleres Wort.

*Ergebnis: Fast perfekt.*

### **Iteration 5:**

Letzter Feinschliff: Füge nach dem zweiten Absatz eine Leerzeile ein  
und setze einen kurzen Satz in Klammern, der eine persönliche Meinung einbringt. So wirkt es authentischer.

*Ergebnis: Fertig. Ein Post, den ich veröffentlichen kann.*

Fünf Iterationen. Der erste Versuch war meh. Der fünfte war gut. So funktioniert iteratives Prompting.

## **Das Prompt-Protokoll**

Hier kommt ein Tipp, der dich langfristig besser macht: Führe ein Prompt-Protokoll.

Was ist das? Eine Sammlung deiner besten Prompts. Schreib dir die Prompts auf, die besonders gut funktioniert haben – inklusive der Iterationen, die nötig waren, um dahin zu kommen.

Das klingt nach Mehraufwand, und am Anfang ist es das auch. Aber nach ein paar Wochen hast du eine persönliche Prompt-Bibliothek, auf die du immer zurückgreifen kannst. Du musst das Rad nicht jedes Mal neu erfinden.

Ich mache das mit einer einfachen Textdatei. Andere benutzen Notion, Obsidian oder Google Docs. Das Tool ist egal – Hauptsache, du machst es.

Ein Eintrag in meinem Protokoll sieht so aus:

Datum: 15.03.2026  
Zweck: LinkedIn-Post über KI im Bildungswesen  
Modell: Claude 3.5 Sonnet  
Iterationen: 5

Finaler Prompt:

"Du bist ein Bildungsexperte, der auf LinkedIn aktiv ist. Schreib einen Post (maximal 150 Wörter) darüber, wie Lehrer KI als Werkzeug im Unterricht nutzen können. Starte mit einer konkreten Szene aus dem Schulalltag. Kein Buzzword-Bingo, kein Corporate-Sprech. Inklusive einem konkreten Call-to-Action am Ende.  
Ton: enthusiastisch aber realistisch."

Notizen: Erster Versuch war zu generisch. Key Learning: Immer einen konkreten Aspekt wählen statt ein breites Thema.

## Aus Fehlern lernen

Jede schlechte Antwort ist eine Lektion. Wenn das Modell nicht liefert, was du willst, liegt das fast immer am Prompt – nicht am Modell.

Frag dich:

- War ich spezifisch genug?
- Fehlte Kontext?
- Hätte eine Rolle geholfen?
- War das gewünschte Format klar?
- Habe ich zu viel auf einmal verlangt?

Meistens liegt es an einem dieser Punkte. Und wenn du den Fehler erkennst, wird dein nächster Prompt besser. Prompt Engineering ist eine Fähigkeit, die mit jedem Versuch wächst.

## Wann du aufhören solltest zu iterieren

Auch das ist eine berechnete Frage. Iteratives Prompting heißt nicht, dass du endlos hin und her korrigieren sollst. Irgendwann gibt es Diminishing Returns – die Verbesserungen werden kleiner und der Aufwand größer.

Meine Faustregel: Wenn du nach 5-7 Iterationen nicht da bist, wo du hinwillst, dann ist das Modell für diese Aufgabe nicht das richtige Werkzeug. Oder du musst deinen Ansatz komplett ändern – nicht nur einzelne Wörter austauschen.

Manchmal ist es effizienter, den ganzen Prompt zu löschen und von vorn anzufangen, als eine schlechte Antwort durch 20 Korrekturen zu retten.

## Das Wichtigste aus diesem Kapitel

- Der erste Prompt muss nicht perfekt sein – iteriere
  - Folge-Prompts sind dein Werkzeug zur Verfeinerung
  - Sei spezifisch in deinen Korrekturen (“Satz 3 kürzen”, nicht “mach es besser”)
  - Führe ein Prompt-Protokoll für langfristiges Lernen
  - Nach 5-7 Iterationen: Entweder passt es oder du brauchst einen neuen Ansatz
- 

## Übung

**Starte mit einem einfachen Prompt und verbessere ihn in 5 Iterationen.**

1. Wähle ein Thema, das dich interessiert
2. Schreib einen bewusst einfachen, vagen ersten Prompt
3. Lies die Antwort und notiere, was nicht passt

4. Schreib einen Folge-Prompt, der genau eine Sache verbessert
5. Wiederhole Schritt 3 und 4 noch dreimal (insgesamt 5 Iterationen)
6. Vergleiche die erste und die letzte Antwort

Schreib dir den finalen Prompt auf – er kommt in dein Prompt-Protokoll.

Bonus-Übung: Versuche dieselbe Aufgabe, aber schreibe direkt einen detaillierten Prompt (ohne Iteration). Vergleiche, ob das Ergebnis besser oder schlechter ist als nach 5 Iterationen. Du wirst überrascht sein, wie oft der iterative Weg gewinnt.

# Kapitel 10: Zusammenfassung und Ausblick

---

Du hast es geschafft. Band 1 ist durch.

Lass mich kurz zusammenfassen, was du in den letzten neun Kapiteln gelernt hast – und dann zeige ich dir, wohin die Reise in Band 2 geht.

## Was du jetzt kannst

Wenn du die Übungen gemacht hast (und ich hoffe wirklich, dass du das hast), dann kannst du jetzt:

1. **Erklären, was KI ist** – und was sie nicht ist. Du weißt, dass LLMs keine denkenden Wesen sind, sondern statistische Modelle, die Wort für Wort vorhersagen.
2. **Verstehen, wie LLMs funktionieren** – Token, Kontextfenster, Training, Temperatur. Du musst kein Ingenieur sein, aber du verstehst die Grundlagen.
3. **Prompts schreiben** – nicht nur einfache Fragen, sondern strukturierte Anweisungen mit Kontext, Rolle, Format und Einschränkungen.
4. **Die 5 Bausteine anwenden** – Aufgabe, Kontext, Format, Ton, Einschränkungen. Nicht jeder Prompt braucht alle fünf, aber du weißt, wann welcher Baustein hilft.

5. **Fehler erkennen** – Du weißt, warum “Hilf mir” kein guter Prompt ist, warum Kontext alles verändert und warum du Ergebnisse hinterfragen solltest.
6. **Kontext gezielt einsetzen** – Die 5 Kontext-Kategorien und der Goldene Satz helfen dir, dem Modell genau die Informationen zu geben, die es braucht.
7. **Rollen und Personas nutzen** – Vom einfachen “Du bist ein Experte” bis zu detaillierten Personas mit Persönlichkeit.
8. **Output-Formate steuern** – Listen, Tabellen, JSON, Markdown, Templates. Du bekommst die Antwort in dem Format, das du brauchst.
9. **Iterativ arbeiten** – Der erste Prompt ist ein Startpunkt, nicht das Endprodukt. Durch gezielte Folge-Prompts kommst du zum Ergebnis.

## Checkliste: Bin ich bereit für Band 2?

Geh diese Checkliste durch. Wenn du bei mindestens 7 von 10 Punkten “Ja” sagen kannst, bist du bereit.

- Ich kann erklären, was ein LLM ist und wie es grundsätzlich funktioniert
- Ich habe mindestens 2 verschiedene LLMs ausprobiert
- Ich kann die 5 Bausteine eines guten Prompts benennen
- Ich habe einen Prompt geschrieben, der eine Rolle enthält
- Ich habe ein Output-Format gesteuert (Tabelle, Liste, etc.)
- Ich weiß, was Halluzinationen sind und warum ich Ergebnisse prüfen muss
- Ich habe mindestens einmal iterativ einen Prompt verbessert
- Ich habe angefangen, ein Prompt-Protokoll zu führen

- [ ] Ich kenne den Unterschied zwischen einem Google-Suchbegriff und einem Prompt
- [ ] Ich habe Spaß daran, mit KI zu experimentieren

Wenn du bei weniger als 7 Punkten “Ja” sagst, geh nochmal die entsprechenden Kapitel durch. Es gibt keine Eile. Nimm dir die Zeit.

## Was dich in Band 2 erwartet

Band 2 heißt “Prompt-Frameworks – Strukturiert zum perfekten Prompt” und baut direkt auf dem auf, was du hier gelernt hast.

Bisher hast du Prompts eher intuitiv geschrieben – mit den 5 Bausteinen im Kopf, aber ohne feste Struktur. In Band 2 lernst du Frameworks kennen, die dir eine klare Schritt-für-Schritt-Anleitung geben:

- **CRAFT** – Context, Role, Action, Format, Tone. Ein Framework, das die 5 Bausteine in ein System packt.
- **RTF** – Role, Task, Format. Für schnelle Prompts, wenn es nicht komplex sein muss.
- **RISEN** – Role, Instructions, Steps, End goal, Narrowing. Für komplexe Aufgaben mit vielen Teilschritten.

Außerdem lernst du die drei Prompting-Typen kennen, die dir in fast jedem fortgeschrittenen Artikel über Prompt Engineering begegnen:

- **Zero-Shot Prompting** – Du gibst dem Modell eine Aufgabe ohne Beispiel. Das haben wir in Band 1 die ganze Zeit gemacht.
- **One-Shot Prompting** – Du gibst ein Beispiel, an dem sich das Modell orientieren kann.
- **Few-Shot Prompting** – Du gibst mehrere Beispiele. Das Modell erkennt das Muster und folgt ihm.

Und schließlich lernst du, wie du dir eine eigene Template-Bibliothek aufbaust – eine Sammlung von erprobten Prompt-Vorlagen, die du immer wieder verwenden und anpassen kannst.

## Wie die ganze Reihe aufgebaut ist

Falls du dich fragst, wie es nach Band 2 weitergeht – hier der Überblick:

### **Anfänger (Band 1-3)**

- Band 1: Grundlagen (*das hast du gerade gelesen*)
- Band 2: Prompt-Frameworks
- Band 3: Fortgeschrittene Basics

### **Fortgeschritten (Band 4-6)**

- Band 4: Reasoning-Techniken (Chain-of-Thought, Tree-of-Thought, etc.)
- Band 5: Kreatives Prompting (Texte, Bilder, Multimedia)
- Band 6: Spezialisiertes Prompting (Marketing, Bildung, Datenanalyse)

### **Profi (Band 7-9)**

- Band 7: Prompting für Entwickler (Code, APIs, Automatisierung)
- Band 8: Business & Produktivität
- Band 9: Sicherheit, Ethik und Verantwortung

### **Experte (Band 10)**

- Band 10: Die Zukunft (Agenten, Multimodal, Automation)

Jeder Band baut auf den vorherigen auf. Du kannst sie der Reihe nach lesen oder nach Bedarf in die Bände springen, die dich am meisten interessieren. Ich empfehle aber, zumindest Band 1-3 in der Reihenfolge zu lesen – die Grundlagen braucht alles Weitere.

## Ein letztes Wort

Prompt Engineering ist keine Raketenwissenschaft. Es ist eine Fähigkeit, die jeder lernen kann. Ja, auch du. Vor allem du, wenn du dieses Buch bis hierhin gelesen hast.

Was mich persönlich am meisten an diesem Thema begeistert: Es demokratisiert den Zugang zu KI. Du brauchst keinen Dokortitel in Informatik, um LLMs effektiv zu nutzen. Du brauchst keine teuren Tools oder spezielle Hardware. Du brauchst nur die Fähigkeit, klar zu kommunizieren, was du willst.

Und genau das hast du in Band 1 gelernt.

Geh raus und prompt. Experimentiere. Mach Fehler. Lern daraus. Und wenn du bereit bist, schnapp dir Band 2.

Wir sehen uns dort.

*Belkis Aslani*

---

## Ressourcen und weiterführende Links

Hier sind ein paar Ressourcen, die ich empfehlen kann, wenn du tiefer einsteigen willst:

### **Zum Nachlesen:**

- Prompt Engineering Guide ([promptingguide.ai](https://promptingguide.ai)) – Die umfassendste kostenlose Ressource im Netz
- IBM's Guide to Prompt Engineering – Gut strukturiert, viele Beispiele
- LearnPrompting.org – Interaktiver Online-Kurs

**Zum Ausprobieren:**

- ChatGPT (chat.openai.com) – Kostenloser Account verfügbar
- Claude (claude.ai) – Kostenloser Account verfügbar
- Gemini (gemini.google.com) – Kostenlos mit Google-Account

**Zum Weiterlesen:**

- Band 2 dieser Reihe: “Prompt-Frameworks – Strukturiert zum perfekten Prompt”
- “The Quick Guide to Prompt Engineering” (O’Reilly) – Englischsprachig, aber sehr gut
- Diverse YouTube-Channels zum Thema – such einfach nach “Prompt Engineering Tutorial Deutsch”

**Wissenschaftlich:**

- “Attention Is All You Need” (Vaswani et al., 2017) – Das Paper, das alles gestartet hat
- “A Systematic Survey of Prompt Engineering in Large Language Models” (Sahoo et al., 2024) – Umfassende akademische Übersicht