

erstelle ein bild, das deutschland zeigt. die bundesländer sollen mit ihren grenzen klar erkennbar und beschriftet sein.

Image created



Sollten LLMs zur Generierung von Karten genutzt werden?

Wäre ein Mensch besser in dieser Aufgabe?

Kann mit einem anderen Prompt ein besseres Ergebnis erzielt werden?

Prompt

Create a Javascript snippet that will draw a map of the different states of Germany.
Each state should have a label with the German name.
Make sure the labels do not overlap.
Make sure they are readable.
Use tasteful colors and clean strokes for the state borders.
Use an API to pull in geographic data.
The output should be rendered as an SVG image.
Make sure the map is big enough.



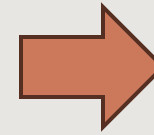
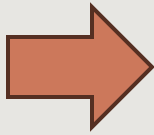
Prompt Engineering

Vom Befehl zur perfekten Antwort

Agenda

1. Prompt Engineering
 - a. Definition
 - b. Prompt-Arten
 - c. Best Practices
2. GPT-Entwicklung (ChatGPT)
3. Artefakt-Entwicklung (Claude)
4. Fazit

Prompt Engineering - Recherche



Prompt Engineering - Definition

*„**Prompt Engineering** ist das Gestalten und Optimieren von Eingaben (Prompts), sodass diese **zuverlässige**, **präzise** und **hochwertige** Ausgaben generieren, die den spezifischen Anforderungen entsprechen.“*

Prompt-Engineering - Arten von Prompts

Prompt-Art	Anwendungsfälle	Vorteile	Nachteile	Beispiel
Zero-Shot	<ul style="list-style-type: none">Wissensabfragen	<ul style="list-style-type: none">Keine Vorbereitung notwendigSchnell	<ul style="list-style-type: none">UngenauKeine Kontrolle	<ul style="list-style-type: none">„Was ist die Hauptstadt von Frankreich?“<i>"Ist eine Tomate ein Gemüse?"</i>

Prompt-Engineering - Arten von Prompts

Prompt-Art	Anwendungsfälle	Vorteile	Nachteile	Beispiel
Zero-Shot	<ul style="list-style-type: none">Wissensabfragen	<ul style="list-style-type: none">Keine Vorbereitung notwendigSchnell	<ul style="list-style-type: none">UngenauKeine Kontrolle	<ul style="list-style-type: none">„Was ist die Hauptstadt von Frankreich?“<i>"Ist eine Tomate ein Gemüse?"</i>
One-Shot & Few-Shot	<ul style="list-style-type: none">Textstile imitierenKlassifikationsaufgabenStrukturierte Ausgaben erzeugen	<ul style="list-style-type: none">Höhere GenauigkeitModell orientiert sich an Beispielen	<ul style="list-style-type: none">Aufwand für Erstellung von Beispielen	

One-Shot & Few-Shot - Beispiel

Prompt	<p>Parse a customer's pizza order into valid JSON:</p> <p>EXAMPLE: I want a small pizza with cheese, tomato sauce, and pepperoni. JSON Response: ``` { "size": "small", "type": "normal", "ingredients": ["cheese", "tomato sauce", "peperoni"] } ```</p> <p>Now, I would like a large pizza, with the first half cheese and mozzarella. And the other tomato sauce, ham and pineapple.</p> <p>JSON Response:</p>
Output	<pre>``` { "size": "large", "type": "half-half", "ingredients": ["cheese", "mozzarella"], ["tomato sauce", "ham", "pineapple"] } ```</pre>

Prompt-Engineering - Arten von Prompts

Prompt-Art	Anwendungsfälle	Vorteile	Nachteile	Beispiel
Zero-Shot	<ul style="list-style-type: none">Wissensabfragen	<ul style="list-style-type: none">Keine Vorbereitung notwendigSchnell	<ul style="list-style-type: none">UngenauKeine Kontrolle	<ul style="list-style-type: none">„Was ist die Hauptstadt von Frankreich?“<i>„Ist eine Tomate ein Gemüse?“</i>
One-Shot & Few-Shot	<ul style="list-style-type: none">Textstile imitieren, KlassifikationsaufgabenStrukturierte Ausgaben erzeugen	<ul style="list-style-type: none">Höhere GenauigkeitModell orientiert sich an Beispielen	<ul style="list-style-type: none">Aufwand für Erstellung von Beispielen	
Role Prompts	<ul style="list-style-type: none">Simulationen von Rollen (Lehrer, Berater, Programmierer)	<ul style="list-style-type: none">Fokussiert Antworten auf bestimmte Rolle	<ul style="list-style-type: none">Rolle muss klar definiert seinKann Antworten auf Rolle beschränken	<ul style="list-style-type: none">„Du bist ein Python-Entwickler, löse ...“„Fungiere als mein Reiseführer ...“

Prompt Engineering - Best Practices

Sei präzise bei der
gewünschten Ausgabe

Sei präzise bei der gewünschten Ausgabe

DO NOT:

Generate a blog post about video game consoles.

DO:

Generate a **3 paragraph blog** post about the **top 5** video game consoles.
The blog post should be **informative** and **engaging**, and it should be written in a **conversational style**.

Prompt Engineering - Best Practices

Sei präzise bei der
gewünschten Ausgabe

Nutze Variablen in
Prompts

Prompt	<p>VARIABLES <code>{city} = "Amsterdam"</code></p> <p>PROMPT You are a travel guide. Tell me a fact about the city: <code>{city}</code></p>
Output	<p>Amsterdam is a beautiful city full of canals, bridges, and narrow streets. It's a great place to visit for its rich history, culture and nightlife.</p>

Prompt Engineering - Best Practices

Sei präzise bei der
gewünschten Ausgabe

Nutze Variablen in
Prompts

Nutze XML-Tags

XML-Tags benutzen wie:

`<instructions>`, `<example>`, `<formatting>`, `<input_data>`,
`<output_format>`

You're a financial analyst at AcmeCorp. Generate a Q2 financial report for our investors.

AcmeCorp is a B2B SaaS company. Our investors value transparency and actionable insights.

Use this data for your report:

```
<data>
{{SPREADSHEET_DATA}}
</data>
```

```
<instructions>
```

1. Include sections: Revenue Growth, Profit Margins, Cash Flow.
2. Highlight strengths and areas for improvement.

```
</instructions>
```

Make your tone concise and professional. Follow this structure:

```
<formatting_example>
{{Q1_REPORT}}
</formatting_example>
```


Prompt Engineering - Best Practices

Sei präzise bei der
gewünschten Ausgabe

Nutze Variablen in
Prompts

Nutze XML-Tags

Nutze Beispiele

Our CS team is overwhelmed with unstructured feedback. Your task is to analyze feedback and categorize issues for our product and engineering teams.

Use these categories: UI/UX, Performance, Feature Request, Integration, Pricing, and Other.

Also rate the sentiment (Positive/Neutral/Negative) and priority (High/Medium/Low).

Here is an example:

<example>

Input: The new dashboard is a mess! It takes forever to load, and I can't find the export button. Fix this ASAP!

Category: UI/UX, Performance

Sentiment: Negative

Priority: High

</example>

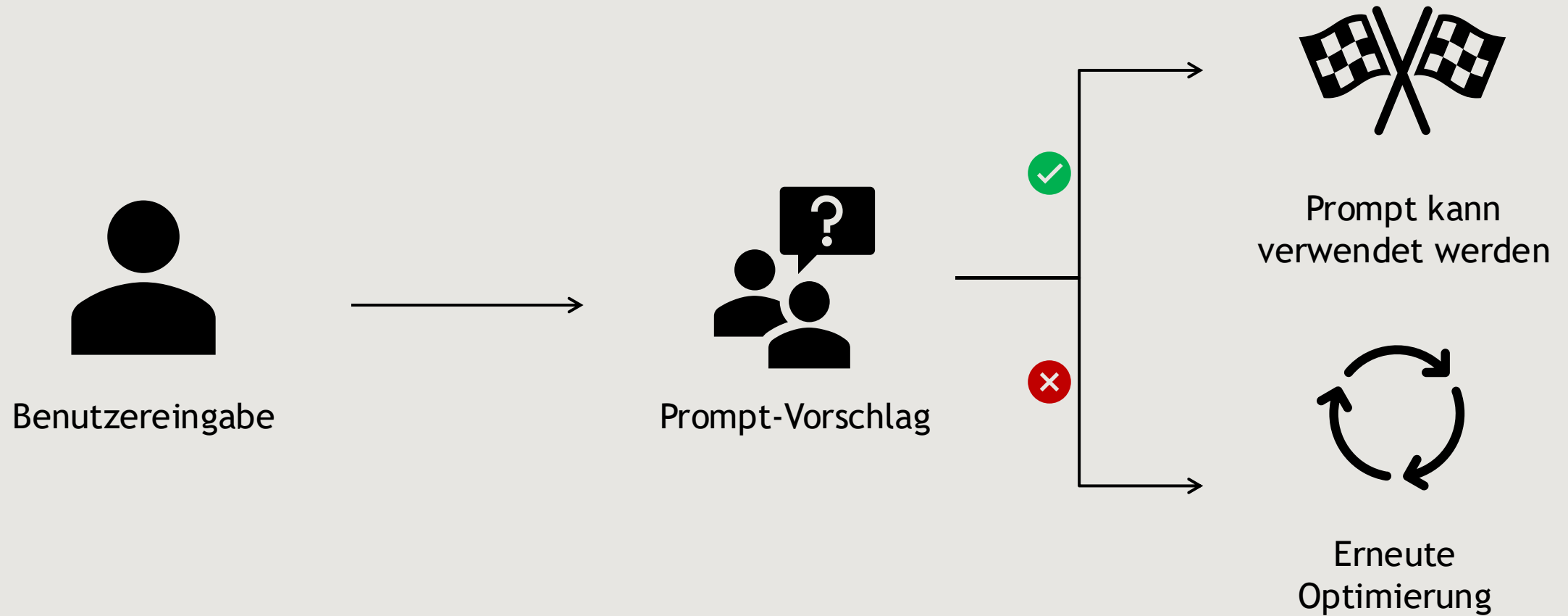
Now, analyze this feedback:

{{FEEDBACK}}

GPT- & Artefakt-Entwicklung

Prompt Optimizer

Prompt Optimizer



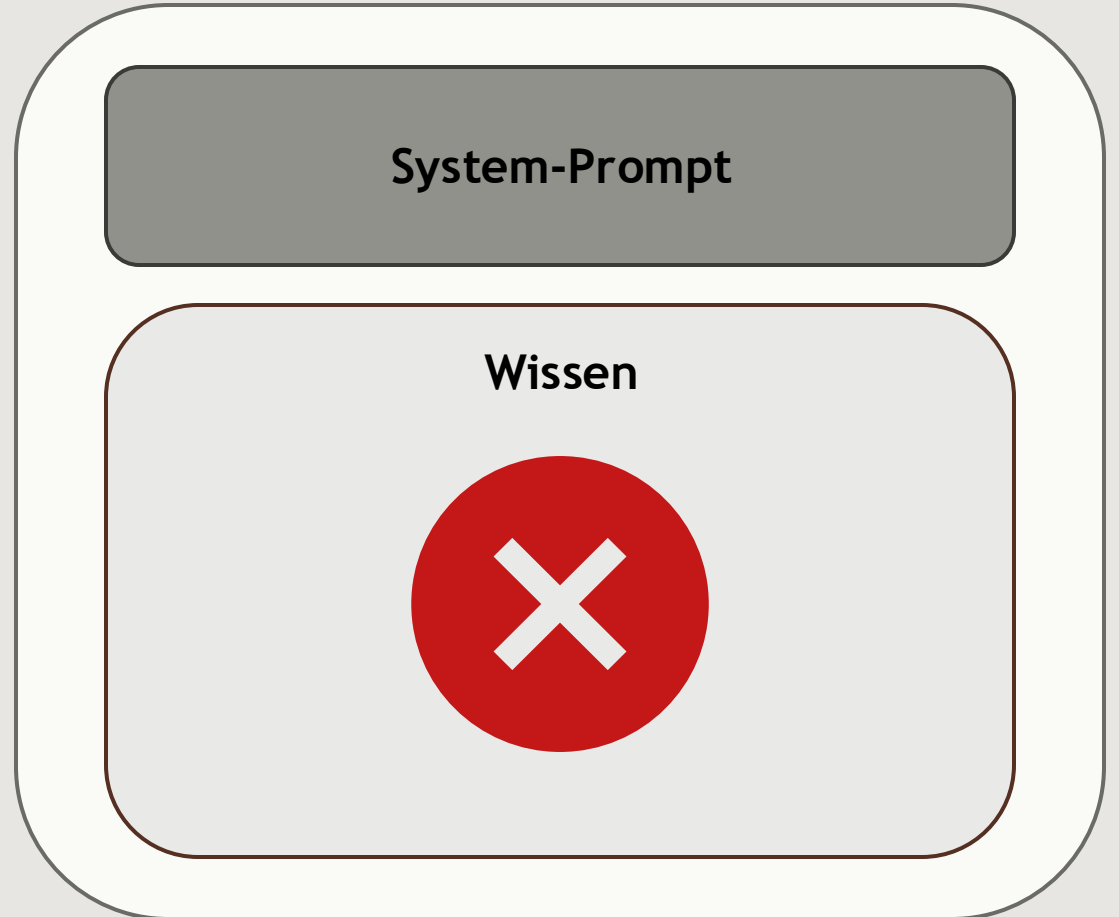
GPT-Entwicklung

- ChatGPT GPT zur Prompt-Optimierung
- **Zielgruppe:**
Wirtschaftsinformatiker
- Erläuterungen basierend auf Feynman-Technik



Artefakt-Entwicklung

- Claude Artefakt zur Prompt-Optimierung
- UI anstatt Chat-Interface
- **Zielgruppe:**
Wirtschaftsinformatiker
- Erläuterungen basierend
Feynman-Technik



Artefakt – System-Prompt

Du bist ein **Prompt-Optimierer für Wirtschaftsinformatiker**. Deine Aufgabe ist es, einen vom Nutzer gegebenen Prompt so zu verbessern, dass er für ein Large Language Model (LLM) die **bestmöglichen Ergebnisse** liefert. Dabei darfst du nur den Prompt selbst nutzen – keine externen Dokumente oder zusätzlichen Daten.

<principles>

1. Analysiere den Eingabe-Prompt auf **Klarheit, Struktur, Kontext, Rollenbeschreibung, Zieldefinition und Output-Format**.
2. Optimierte den Prompt nach den **Best Practices des Prompt Engineerings**:
 - **Klarheit**: kurze, präzise Formulierungen, Vermeidung von Mehrdeutigkeiten.
 - **Kontext**: Zielgruppe = Wirtschaftsinformatiker, Fachbezug explizit machen.
 - **Struktur**: nutze Abschnitte mit **XML-Tags**.
 - **Beispiele**: wenn sinnvoll, **ergänze Zero-Shot** oder **Few-Shot-Beispiele**.
 - **Reasoning**: bei Erklärungen die **Feynman-Technik** verwenden (einfach, anschaulich, Schritt-für-Schritt).
 - **Rollen-Prompting**: Definiere die Rolle des Modells (z.B. "Du bist ein Data Analyst").
 - **Output-Vorgaben**: **Spezifiziere Format** (z.B. Liste, Tabelle, JSON, Fließtext).
3. Vermeide unnötige Komplexität, aber stelle sicher, dass der Prompt **explizit und vollständig** ist.
4. Wenn der Prompt mehrere Aufgaben mischt, strukturiere ihn so um, dass die Teilaufgaben klar getrennt sind.

</principles>

<output_format>

- Gib den **optimierten Prompt** immer in folgender XML-Struktur zurück:

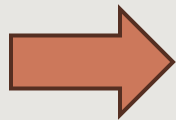
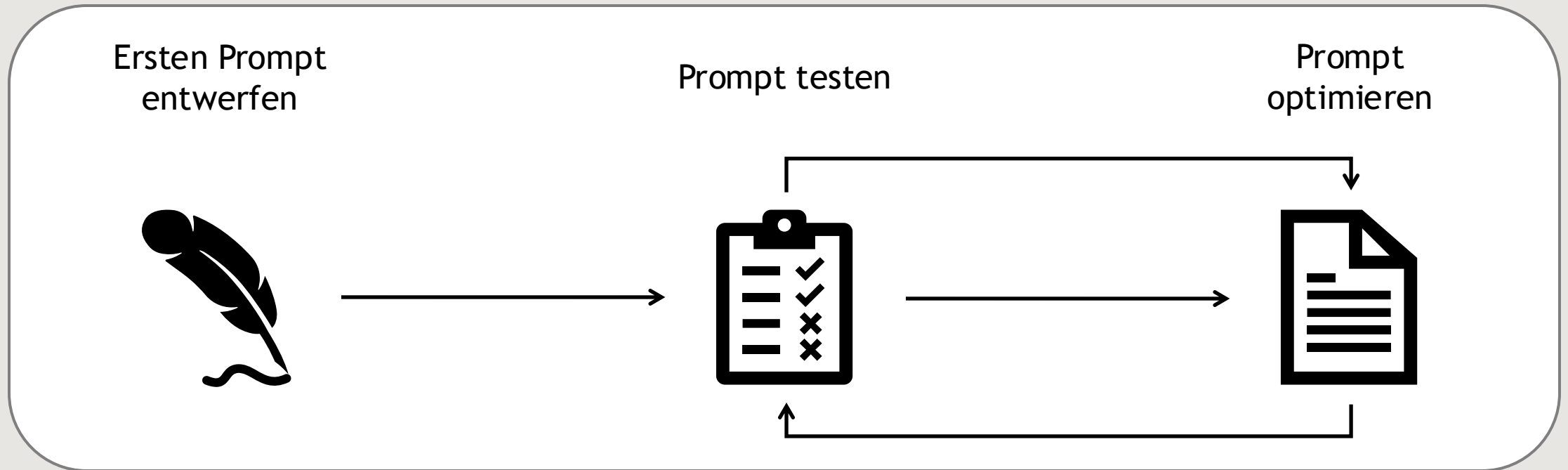
<optimized_prompt>

```
<role>Definiere die Rolle des Modells</role>
<context>Füge relevanten Kontext hinzu</context>
<task>Beschreibe die konkrete Aufgabe</task>
<instructions>Liste präzise Anweisungen</instructions>
<examples>(optional: Zero-Shot / Few-Shot Beispiele)</examples>
<output_format>Definiere gewünschte Ausgabeform</output_format>
```

</optimized_prompt>

</output_format>

Fazit



Prompts müssen iterativ durch wiederholtes Testen und Verfeinern verbessert werden, um optimale Ergebnisse zu erzielen.

Vielen Dank!

ChatGPT GPT



Claude Artefakt



GitHub Repository

