

# Große Sprachmodelle revolutionieren Inhaltsanalyse in der Marktforschung

How can AI be used sensibly within market research?

Paul Simmering  
Senior Data Scientist  
Q Agentur für Forschung

EPHMRA Chapter Meeting Germany 2024

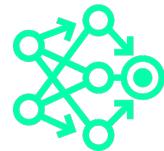


AGENTUR FÜR FORSCHUNG

## Wie ist KI so viel praktischer geworden?



Bessere Basismodelle



Fine-Tuning ist optional



Prompt Engineering



Multimodale Modelle

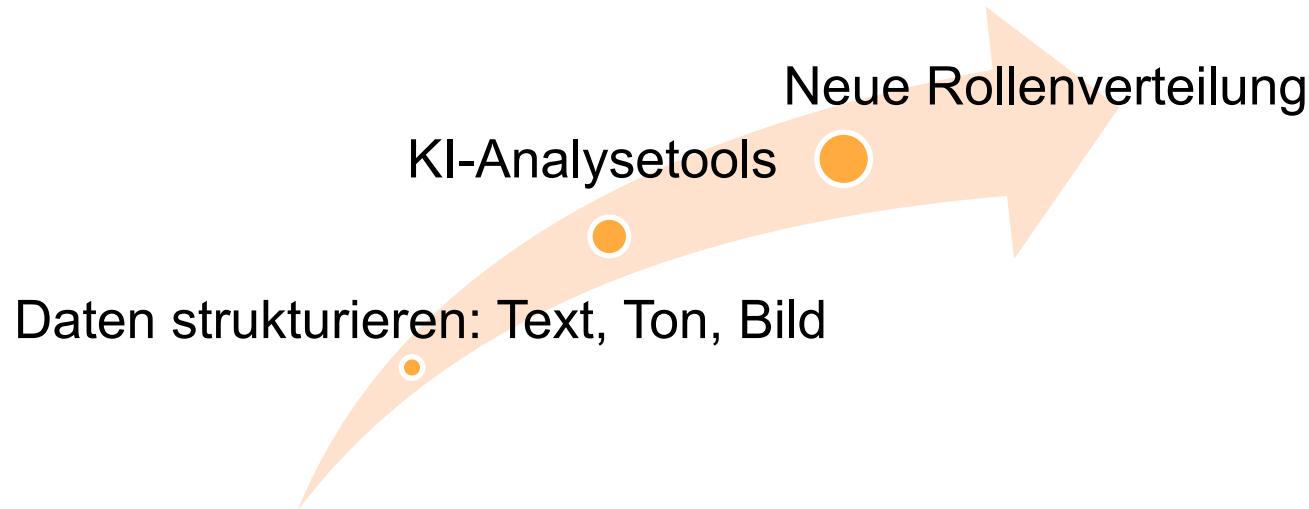
Multi-task Language Understanding

2019: GPT-2-XL  34%

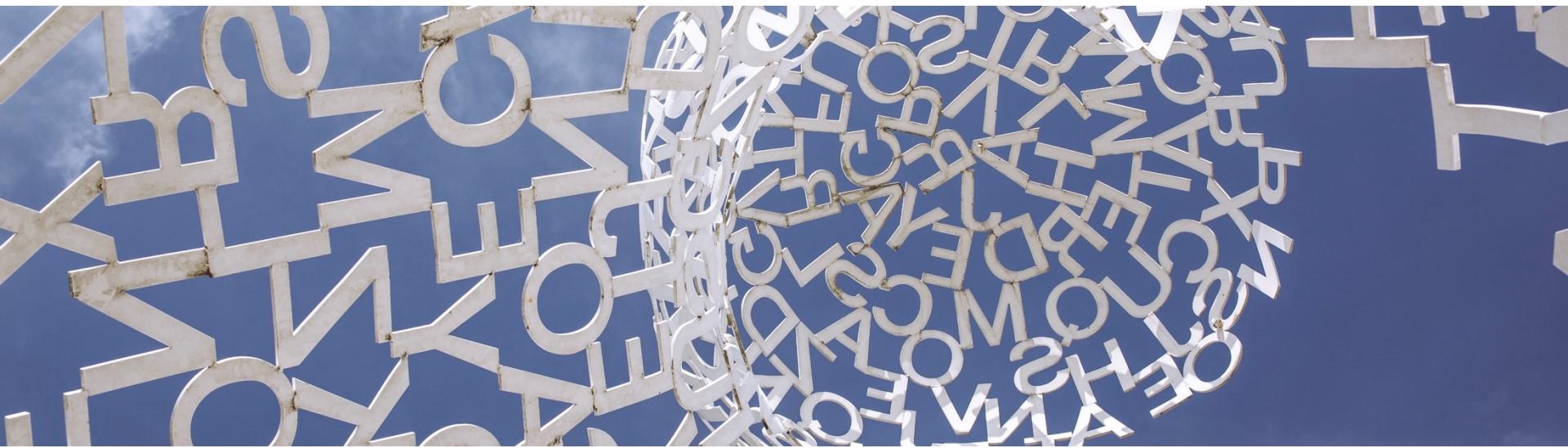
2024: Claude 3 Opus  88%

<https://paperswithcode.com/sota/multi-task-language-understanding-on-mmlu>

## Mit KI Inhaltsanalyse auf ein neues Level heben



# Textanalysen



# Flexible Textanalyse per Prompt

## Beitrag

Bei mir wurde Lungenkrebs diagnostiziert. Am Anfang war ich geschockt. Die Behandlung war hart, mit vielen Arztbesuchen und Nebenwirkungen der Chemotherapie. Aber ich habe nicht aufgegeben. Mit der Unterstützung meiner Familie konnte ich die Krankheit besiegen.

## Prompt

Codierschema:

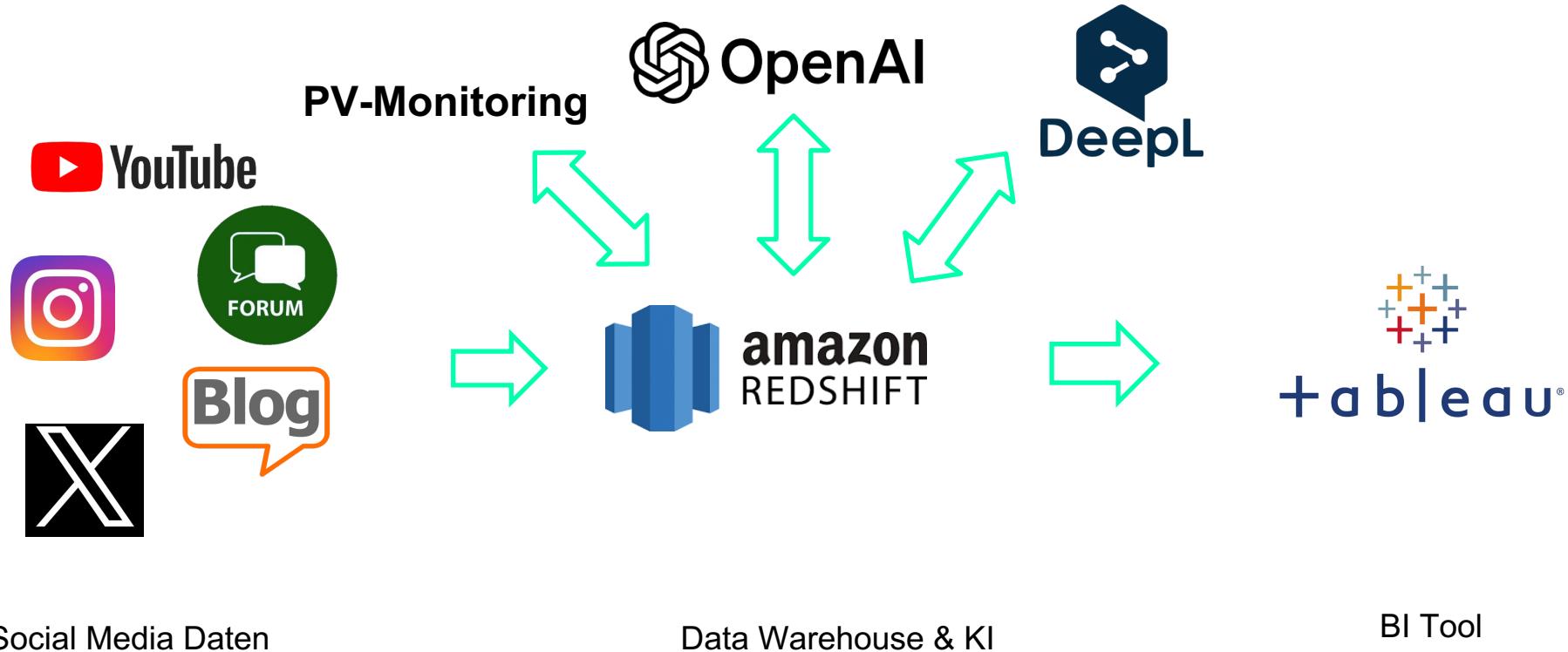
- Relevant?
- Autorensegmente:
  - Patienten
  - Angehörige
  - ...
- Themen:
  - Diagnose
  - Remission
  - ...
- Zusammenfassung

+ JSON Outputformat

## Antwort von GPT-4

```
{  
    "relevant": true,  
    "segment": "Patienten",  
    "themen": [  
        "Diagnose",  
        "Psychologie",  
        "Chemotherapie",  
        "Nebenwirkungen",  
        "Remission"  
    ],  
    "zusammenfassung": "..."  
}
```

# KI als Baustein in einer Social Media Pipeline

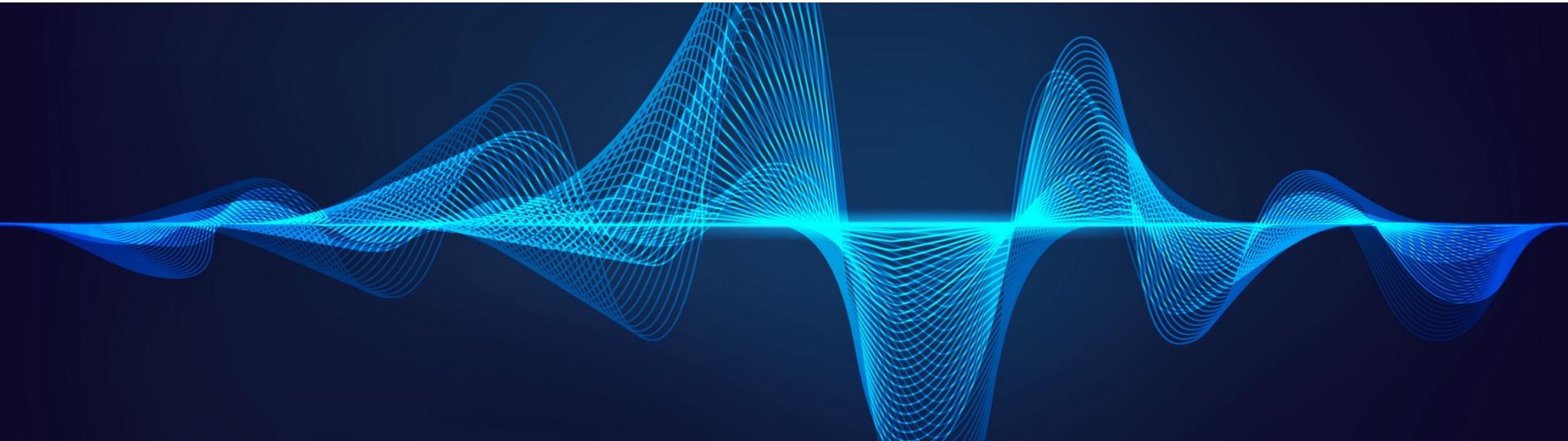


Social Media Daten

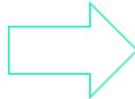
Data Warehouse & KI

BI Tool

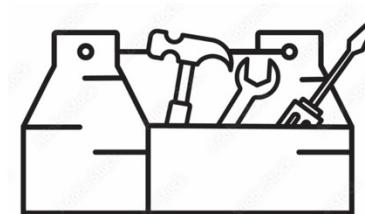
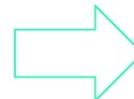
# Audioanalyse



# Akkurate Transkription



05:21 A: Was ist Ihnen  
besonders wichtig?  
05:24 B: Das sich Ärzte Zeit  
nehmen

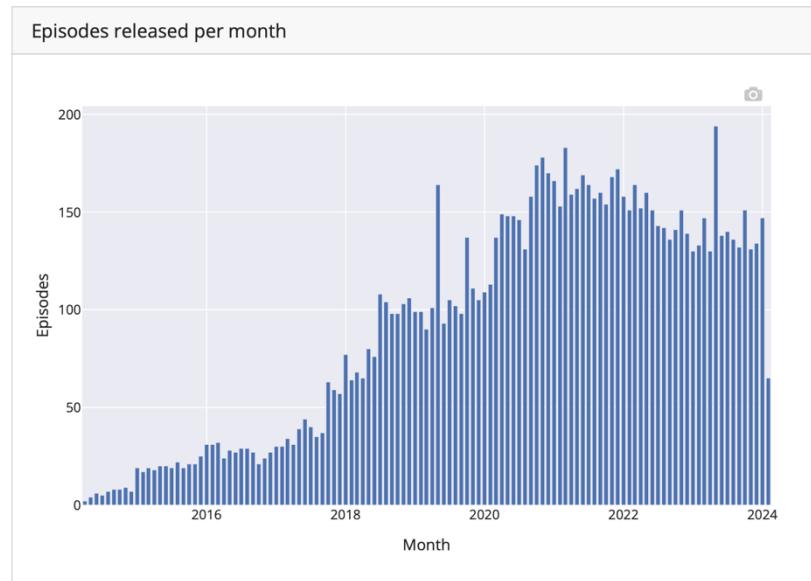
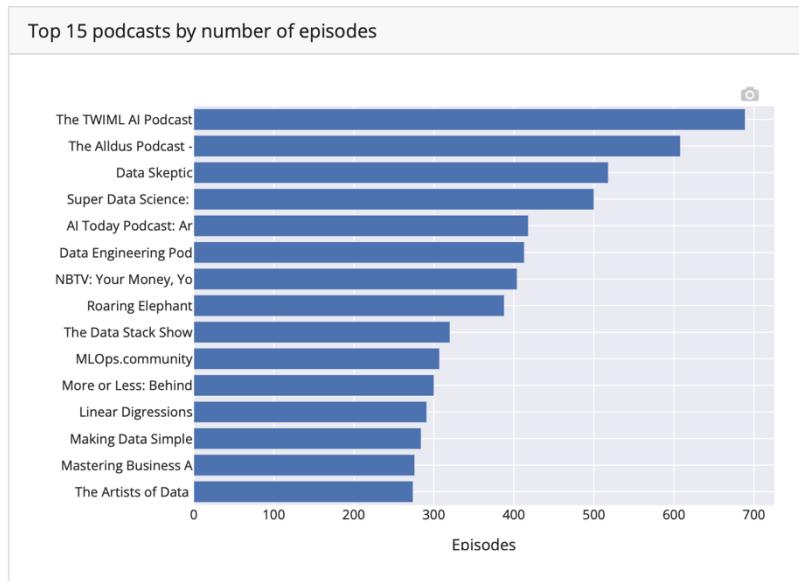
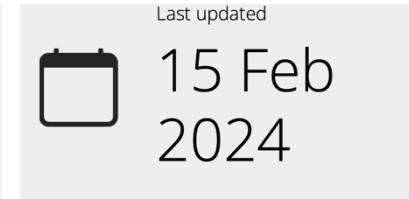


Interviews  
Audio-Tagebücher  
Gruppendiskussionen

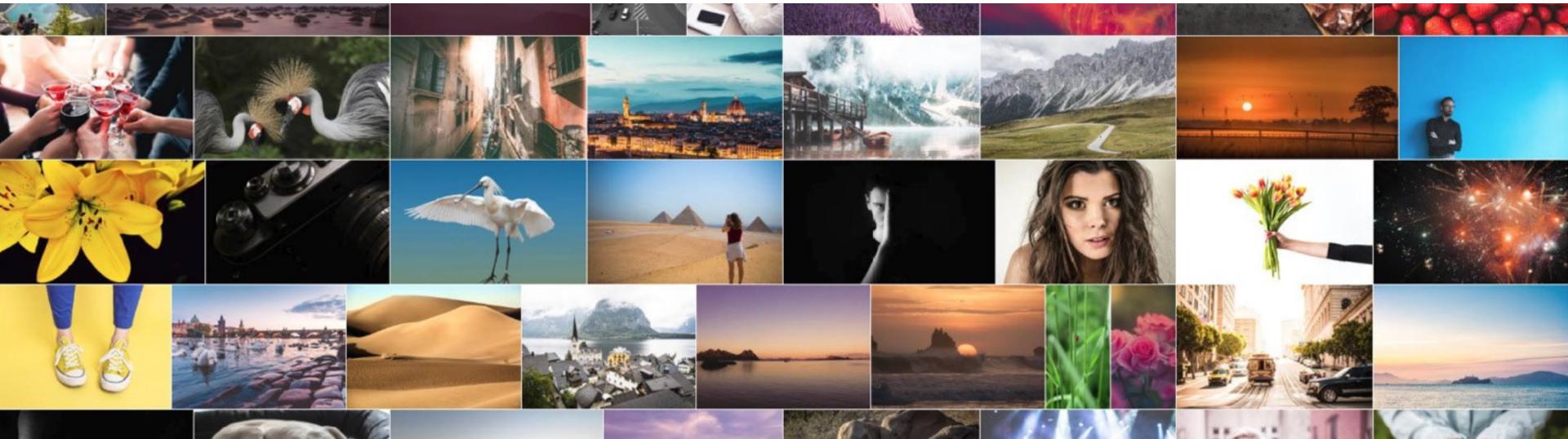
Transkription  
Cloud & Lokal  
4 bis 6% Wortfehlerrate  
Speaker Diarization  
<https://paperswithcode.com/sota/speech-recognition-on-common-voice-german>

Textanalyse-Toolkit  
Zusammenfassungen

# Projektbeispiel: Podcast-Analyse



# Bildanalyse



# Bilder befragen, als wären sie Text



“Extract a list of people and objects and their descriptions from this image. Answer in JSON format. Describe characteristics with keywords rather than full sentences. Focus on the foreground.”

```
{
  "people": [
    {
      "person1": {
        "position": "sitting",
        "activity": "receiving vaccination",
        "gender": "male",
        "mask": "wearing",
        "clothing": "grey t-shirt"
      },
      "person2": {
        "position": "standing",
        "activity": "administering vaccination",
        "gender": "female",
        "mask": "wearing",
        "clothing": "blue scrubs"
      }
    }
  ],
  "objects": [
    {
      "table": {
        "color": "white",
        "content": "vaccination supplies"
      }
    },
    {
      "sign": {
        "text": "COVID-19 CORONAVIRUS VACCINATION",
        "color": "yellow with black and red text"
      }
    }
  ]
}
```

GPT4-V

# Bildanalyse

Fiktive Studie  
1000 TN

Aufgabe:  
Mach ein Foto von dem  
Ort, wo du deine Medi-  
kamente aufbewahrst



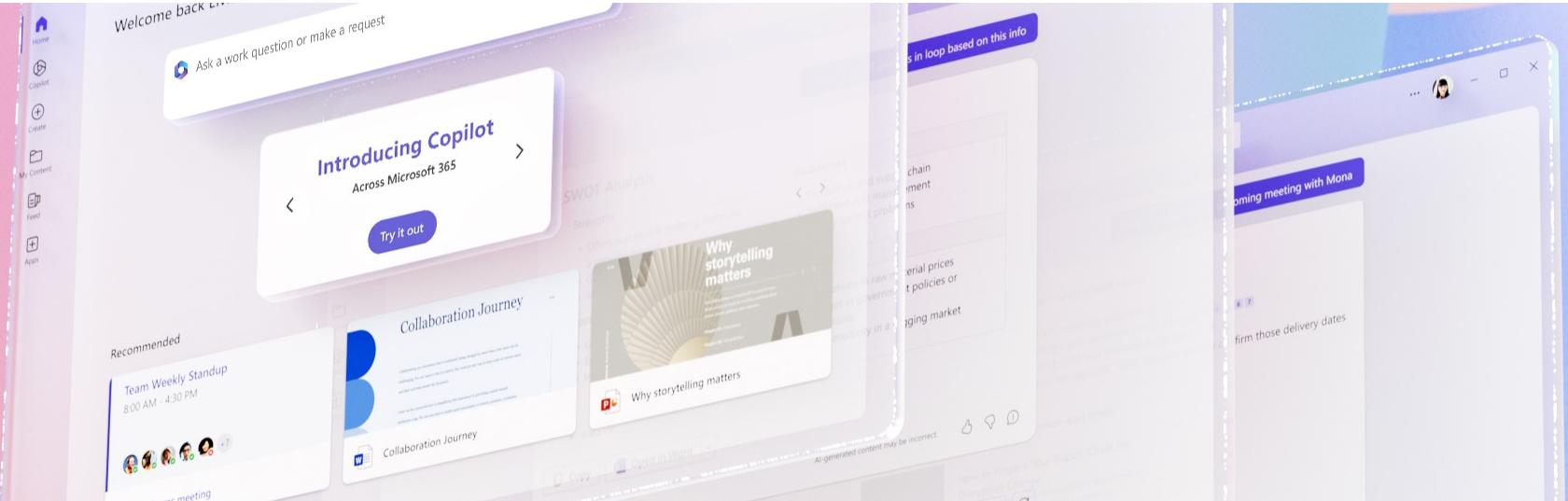
Projekt

1000 Fotos

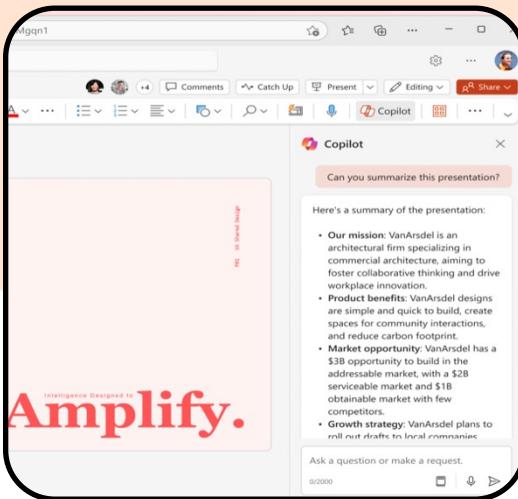
Insights

- Welche Produkte werden gelagert?
- Wo wird gelagert?
- Wie viel lagert?
- Was wird dort noch gelagert?

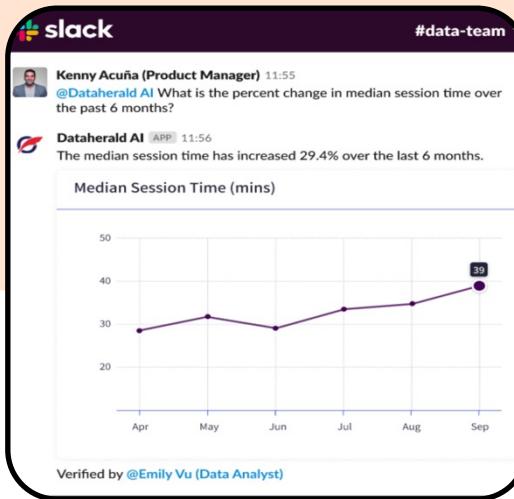
# KI-Analysetools



# Copiloten beschleunigen die Analyse



Microsoft Copilot



Datenbanken in natürlicher Sprache befragen (Text-to-SQL)



monalisa  
Write a unit test for this function

GitHub Copilot

```
import unittest
import datetime

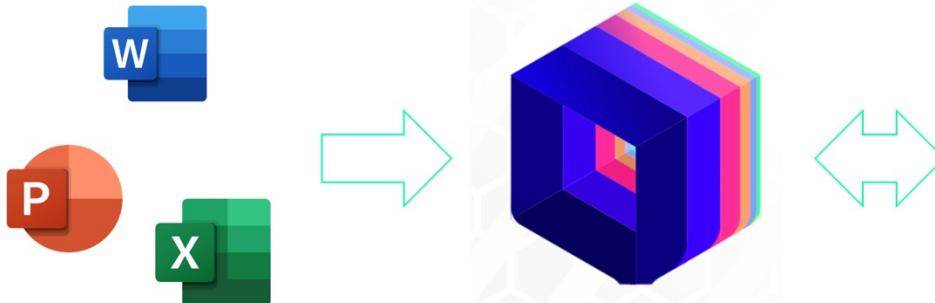
class TestParseExpenses(unittest.TestCase):
    def test_parse_expenses_with_valid_input(self):
        expenses_string = '2023-01-02 -34.01 USD\n2023-01-03 -12.50 EUR\n2023-01-04 -5.00 GBP'
        expected_output = [(datetime.datetime(2023, 1, 2), -34.01, 'USD'), (datetime.datetime(2023, 1, 3), -12.5, 'EUR'), (datetime.datetime(2023, 1, 4), -5.0, 'GBP')]
        self.assertEqual(parse_expenses(expenses_string), expected_output)

    def test_parse_expenses_with_empty_input(self):
        expenses_string = ''
        expected_output = []
        self.assertEqual(parse_expenses(expenses_string), expected_output)

if __name__ == '__main__':
    unittest.main()
```

Code Generation

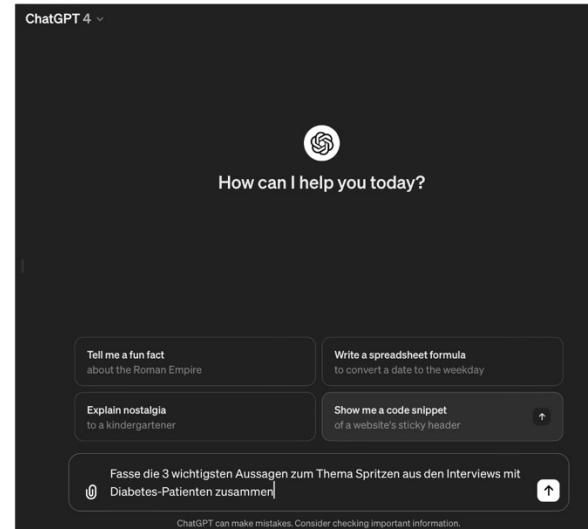
# Retrieval Augmented Generation (RAG)



Dokumente

Vektordatenbank

Chatbot



# Relevantere und präzisere Antworten durch Customization

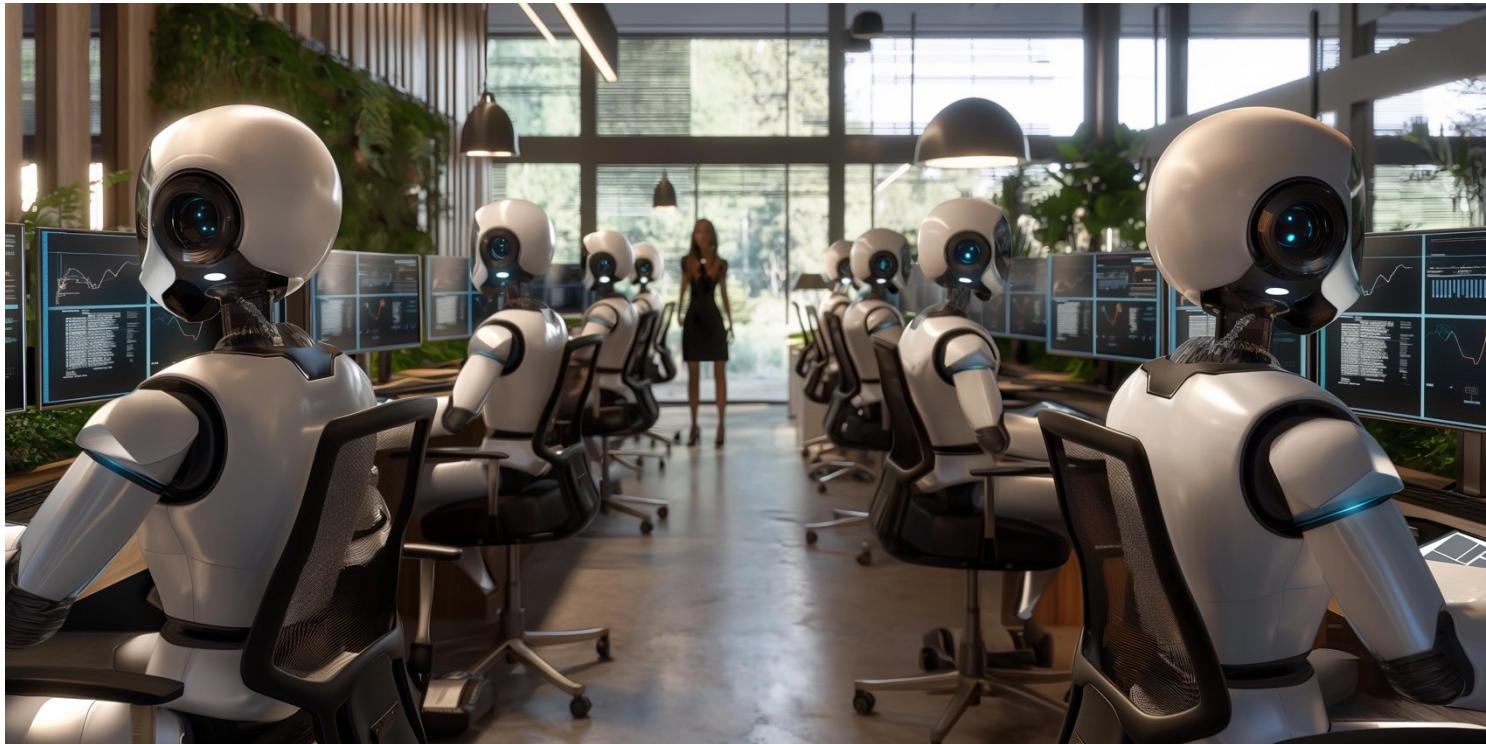
METHODE	ZWECK	WISSEN & TOOLS	
	Prompt Engineering	Aufgabe und Vorgehen definieren	30min Kurs* & in ChatGPT probieren
	Retrieval Augmented Generation	Große Datenmengen durchsuchen	MS Copilot, Adobe Acrobat, eigener Code
	Fine-Tuning	Anpassung des Modells selbst	Trainingsdaten, Machine Learning, Hardware

\*Prompting Kurs: <https://www.linkedin.com/learning/prompt-engineering-how-to-talk-to-the-ais/designing-a-prompt>

## Neue Rollenverteilung

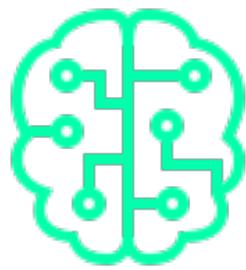


# LLMs als Armee von Analysten



## Was ist der Mehrwert?

### Größer denken



Mehr Daten analysieren.  
Mehr Projekte realisieren.

### Geprüfte Genauigkeit



Präzise Antworten.  
Keine Probleme mit  
Inter-Coder-Reliabilität.

### Mehr Zeit für das Big Picture



Strategie und  
Kommunikation statt  
kleinteilige Analysen.

# Qualitätssicherung ist notwendig

## Probleme

- ✗ Ungenaue, unvollständige, veraltete und halluzinierte Antworten
- ✗ Misstrauen
- ✗ Falsche Erwartungen

## Maßnahmen

- ✓ Genauigkeit quantifizieren mit einem Test-Set
- ✓ Input & Output-Monitoring
- ✓ Automatisierte Daten-Checks
- ✓ Modelle via Prompt Engineering, RAG und Fine-Tuning anpassen
- ✓ Selbst ausprobieren und Intuition entwickeln

**Vielen Dank!**

Paul Simmering

Q Agentur für Forschung

Web: [teamq.de](http://teamq.de)

Mail: [paul.simmering@teamq.de](mailto:paul.simmering@teamq.de)

X: [@Q\\_InsightAgency](https://twitter.com/@Q_InsightAgency)

LinkedIn: [Q Agentur für Forschung](https://www.linkedin.com/company/q-agentur-fuer-forschung/)