UNIVERSITÄT
LEIPZIG

# GetTogetheR!

## ChatGPT und R

Jonas Frost
jonas.frost@studserv.uni-leipzig.de

Peter Kannewitz
peter.kannewitz@uni-leipzig.de

08. Juni 2023

# Motivation

- ChatGPT und andere KI-Tools haben in letzter Zeit stark an Bedeutung gewonnen
  - mehr Anwendungsmöglichkeiten
  - schnellere Benutzung
  - einfacher Zugang
  - große Trainingsdatensätze
- Wir können diese Tools für unsere Arbeit mit R nutzen!
- ChatGPT kann R-Code ...
  - generieren
  - korrigieren
  - erklären
  - kommentieren
  - optimieren
- **Wir müssen wissen, wie wir am besten mit ChatGPT kommunizieren, um die gewünschten Ergebnisse zu bekommen (Prompting)**

# Ablauf

1. Kurzeinstieg: ChatGPT
2. ChatGPT nutzen
3. Austausch

# Kurzeinstieg: ChatGPT

# Kleine Demonstration

- https://platform.openai.com/tokenizer
- https://platform.openai.com/playground

# Was ist GPT?

- **G**enerative **P**re-trained **T**ransformer
  - generatives Sprachmodell mit Transformer Architektur
  - Zusammensetzung von künstlichen neuronalen Netzen
- Bezeichnet eine Familie von Large Language Models (LLM)
  - Ziel: möglichst viel Trainingsdaten, um Sprache “generieren” zu können
- Erste Version GPT-1 2018 veröffentlicht (117 Millionen Parameter) [1]
  - aktuell: GPT-4, 2023 veröffentlicht
  - Vorgänger: GPT-3 mit 175 Billionen Parametern
- ChatGPT ist eine für Dialog optimierte Variante von GPT

[1] https://en.wikipedia.org/wiki/Generative_pre-trained_transformer

# Trainingsdaten

- Riesiger Textkorpus -> 410 Billionen Tokens

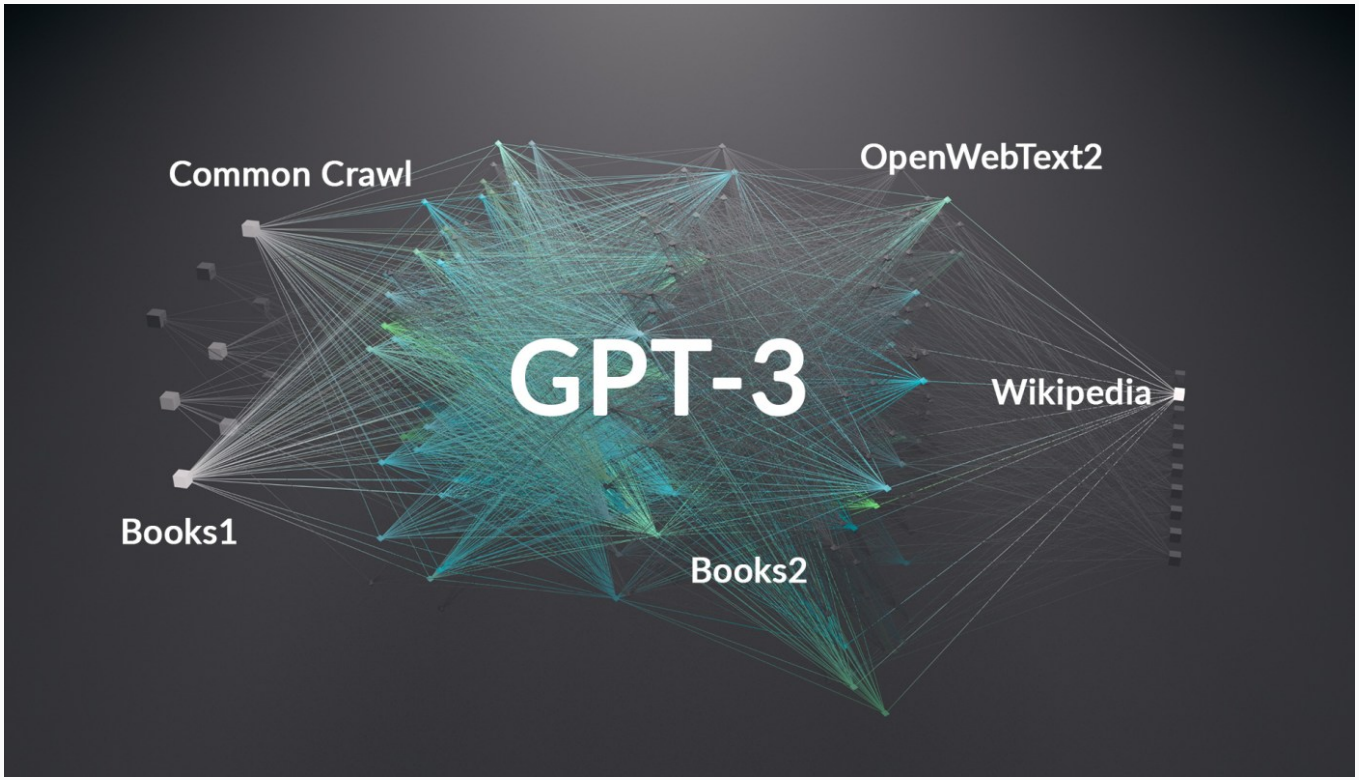


**Figure 1:** Quelle: https://katzlberger.ai/2021/04/12/mit-diesen-daten-wurde-gpt-3-trainiert/

# Intuition: Wie funktioniert ein LLM?

- Unupervised machine learning
  - keine explizite “Ground Truth”
  - Muster in den Trainingsdaten werden im Netz abgebildet
- Ziel: Gegeben eine Vorgeschichte $\longrightarrow$ Was ist das nächst Wort?

*The best thing about AI is its ability to*

| | |
|---|---|
| learn | 4.5% |
| predict | 3.5% |
| make | 3.2% |
| understand | 3.1% |
| do | 2.9% |

**Figure 2:** Quelle: https://writings.stephenwolfram.com/2023/02/what-is-chatgpt-doing-and-why-does-it-work/

# Intuition: künstliche Neuronale Netze

- Inspiriert vom menschlichen Gehirn

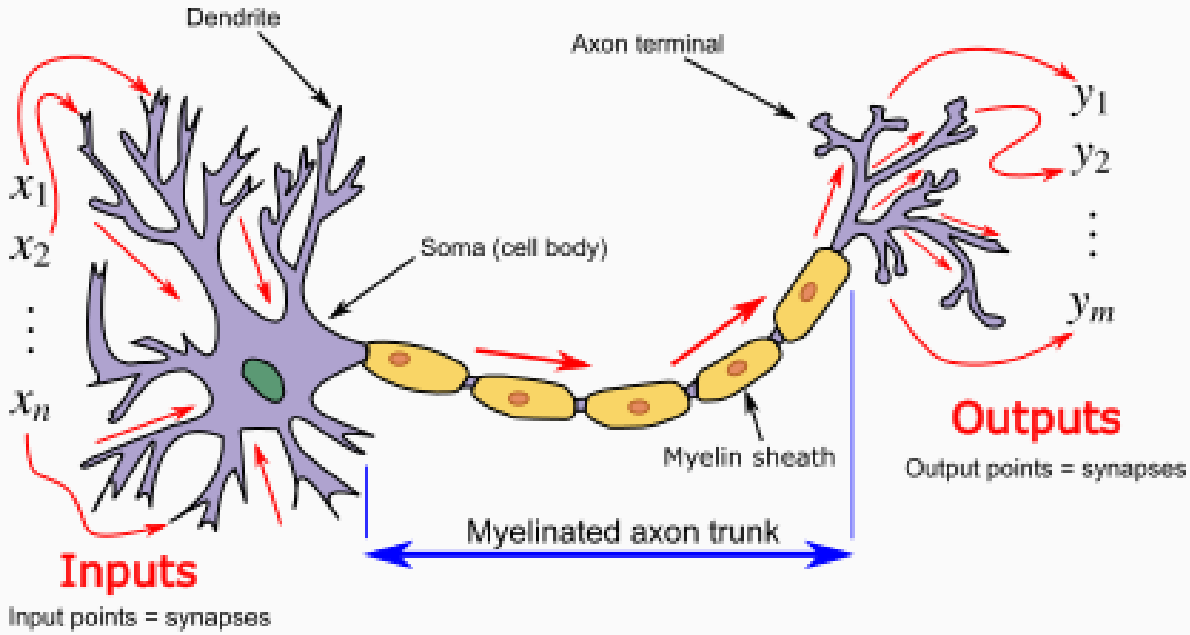


**Figure 3:** Quelle: https://upload.wikimedia.org/wikipedia/commons/4/44/Neuron3.png

- Architektur eines künstlichen neuronalen Netzwerks

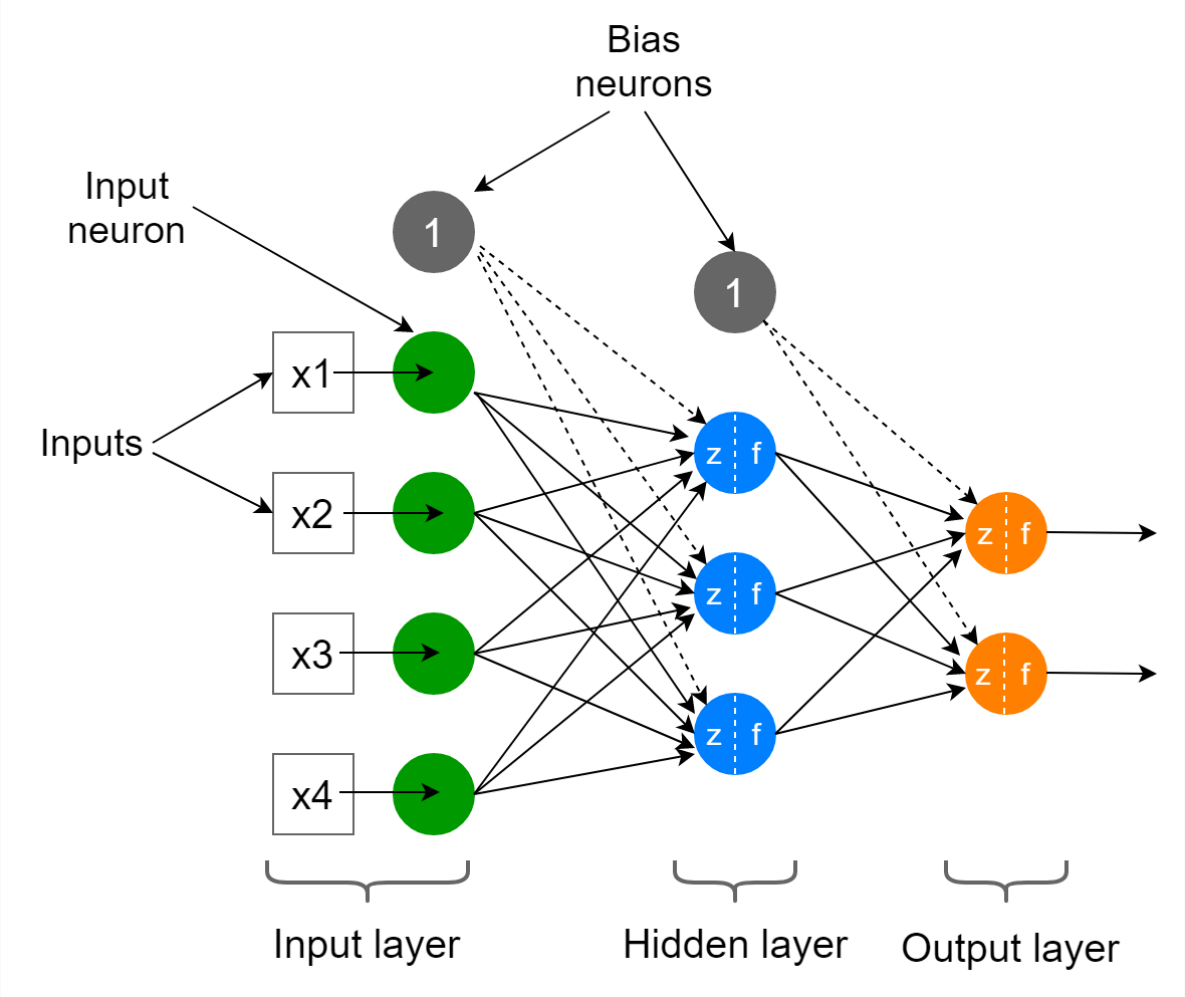


**Figure 4:** Quelle: https://medium.com/data-science-365/one-hidden-layer-shallow-neural-network-architecture-d45097f649e6

# Probleme von LLMs

- Mehrere “Layer”, die nicht linear miteinander Interagieren (96 bei GPT-3)
- Outputs von mehrschichtigen neuronalen Netzen sind kaum bis gar nicht Rückführbar auf einzelne Teile des Netzwerkes
  - $\longrightarrow$ Blackbox
- Sprachmodelle sind sehr rechenintensiv (sowohl Training als auch Nutzung)
  - ChatGPT kann nur serverseitig genutzt werden (Datenschutz?)

# ChatGPT nutzen

# Möglichkeiten ChatGPT in einen R-Worflow zu integrieren

- WebApp
  - chat.openai.com
  - Voraussetzungen: OpenAI Konto
- R-Packages
  - Voraussetzungen: Konto, gültigen API-Key
  - Einschränkung: Rate-Limit
  - R-Package und und RStudio-Plugin: `chatgpt::`

# Wir zeigens Euch: WebApp

## Beispiel 1

”Given a dataset with 3 variables ”name”, ”group” and ”status”, where name is a character, and group and status are dummy variables. Generate an R code that generates a cross table of the two variables group and status. Also, the code should filter cases with status 1 and print the names of persons with status 1. Use the tidyverse package and comment the code.”

## Beispiel 2

”Now, using the same dataset write an R code using the ggplot2 package that generates a barchart showing the relative frequencies of the status variable. Color the bars using the group variable.”

# Tipps für’s Prompting

- gebt ChatGPT so spezifische Anweisungen wie möglich:
  - Welche Packages?
  - Welche Datentypen?
  - Wie soll das Ergebnis aussehen?
  - Wie lautet die Fehlermitteilung?
- Oft ist Codegeneration mit ChatGPT iterativ:
  1) Initialer Prompt
  2) erster generierter Code
  3) Code testen
  4) ChatGPT auf Fehler hinweisen oder Prompt entsprechend anpassen
  5) Schritt 3) und 4) wiederholen, bis Code wie erwartet läuft

# Was ist eine API?

- API -> Application Programming Interface
  - ist eine vom Betreiber der Anwendung bereitgestellte Schnittstelle
- Oft an einen sogenannten API-Key gebunden, über welchen man sich authentifiziert

# OpenAI API Zugriff

- Anmeldung bei OpenAI erforderlich
- $5 Startguthaben, aber nur für die ersten 3 Monate
- Rate Limits für Startguthaben: 3 requests per minute
- Danach muss Zahlungsmittel hinterlegt werden, für weiter Nutzung. Kosten[2]:
  - GPT-4, 8K context, Prompt: $0.03/1K tokens
  - GPT-3.5-turbo: $0.002/1K tokens
- Für R-Package `chatgpt` ist API-Key erforderlich

[2] https://openai.com/pricing

# Wir zeigens Euch: Plugin `chatgpt`

- API-Key als globale Systemvariable **OPENAI_API_KEY** abspeichern
- Plugin und `chatgpt`-Funktionen nun nutzbar
- Funktionsweise: Package sendet markierten Code und vorgefertigten Promt über die API an ChatGPT
- Einfaches Beispielscript mit Plugin bearbeiten

# Austausch

# Austausch

- Welche Erfahrung habt Ihr mit ChatGPT in Bezug auf R gemacht?
- Welche Nachteile könnte ein aktiver Gebrauch von Chat-GPT haben?
- Wie Reproduzierbar sind Antworten von ChatGPT, was bedeutet das für die Forschung?

Danke fürs Teilnehmen!