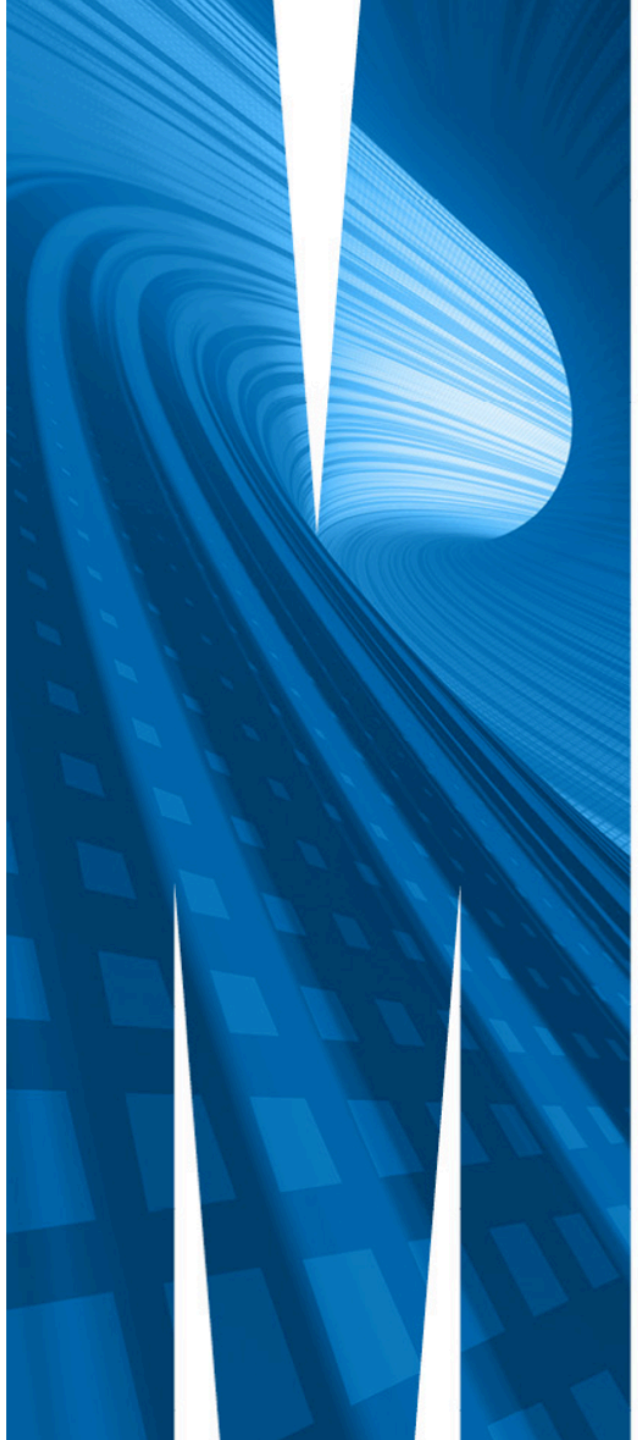
MONASH University

LLMs for Preparing Data in R

Guest lecture for ETC5512: Wild Caught Data

Cynthia A. Huang

19 May 2025

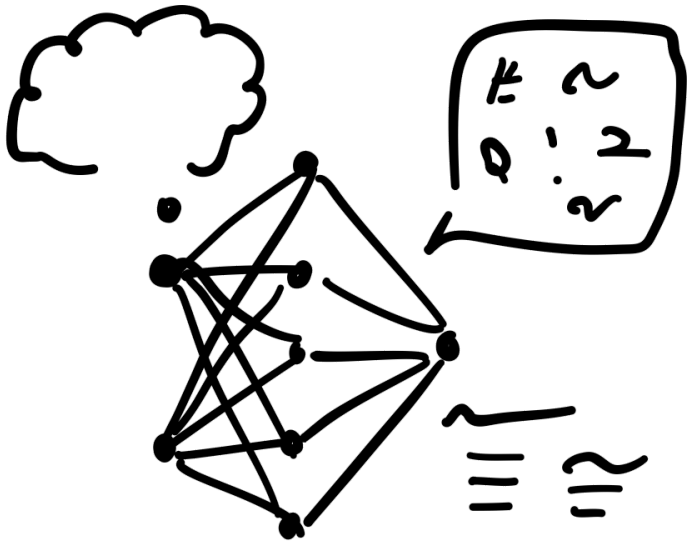


Introduction

Learning goals

- develop your understanding of:
  - what LLMs are,
  - different types of “wild caught data” tasks that LLMs can help with,
  - how to use and check LLMs for specific data prepration tasks,
  - how to interact with LLMs in R using {ellmer}

Lecture outline



OpenAI

ANTHROP\C

- About Me!
- About LLMs
- Using LLMs for “Wild Caught Data” tasks
- Using LLMs in R with `{ellmer}`

About Me!

Who am I?

What do I work on?

About Large Language Models

Generative AI and LLMs

Generative AI refers to:

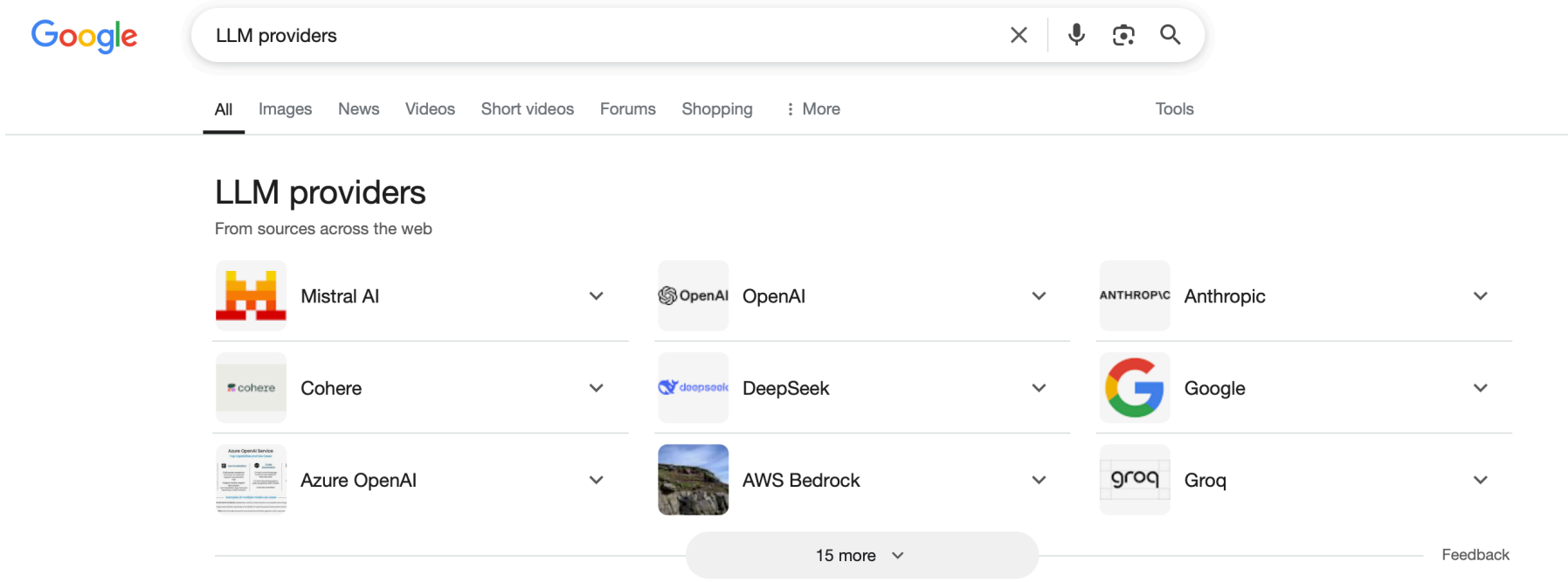
- computer algorithms and systems
- that can generate content such as text, images and sound
- based on patterns learnt from existing data

What are Large Language Models?

LLMs are…

Who makes LLMs?

LLM providers develop and offer access to large language models and systems



Why are there so many different models?

LLM providers offer paid and free access to multiple models:

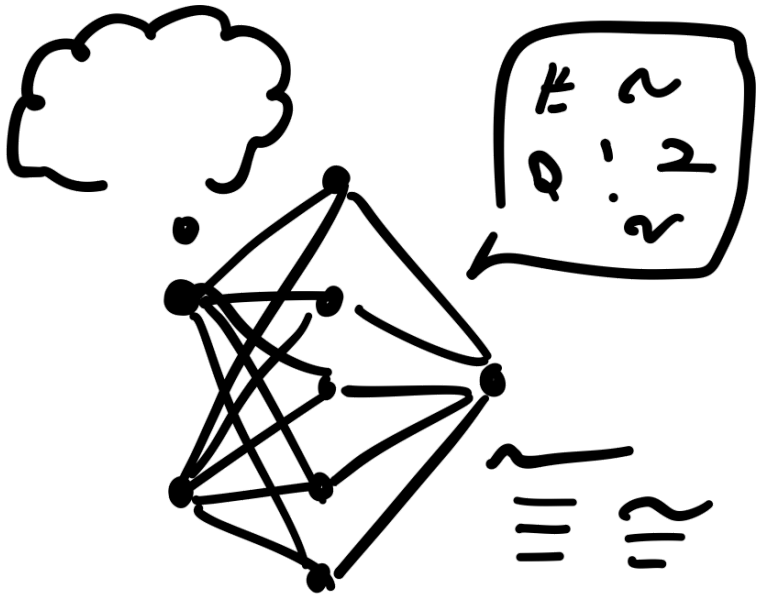
- **OpenAI**: GPT-3 and 4, o-series,
- **Anthroptic**: Claude Haiku, Sonnet and Opus
- **Google**: Gemini Flash and Pro
- **Meta**: Llama 3, Llama 4
- **Alibaba**: Gwen 2.5, 3, Max, Plus and Turbo

Model differentiation

Learn more about picking the right tool:

- 🎙️ Beyond ChatGPT: THE RAPIDLY EVOLVING LANDSCAPE OF AI
- 💡 Suggestions on provider/model choice in ‘ellmer’ docs
- 📖 OpenAI Model Selection Guide
- 📖 Anthropic Claude Model Comparison Table

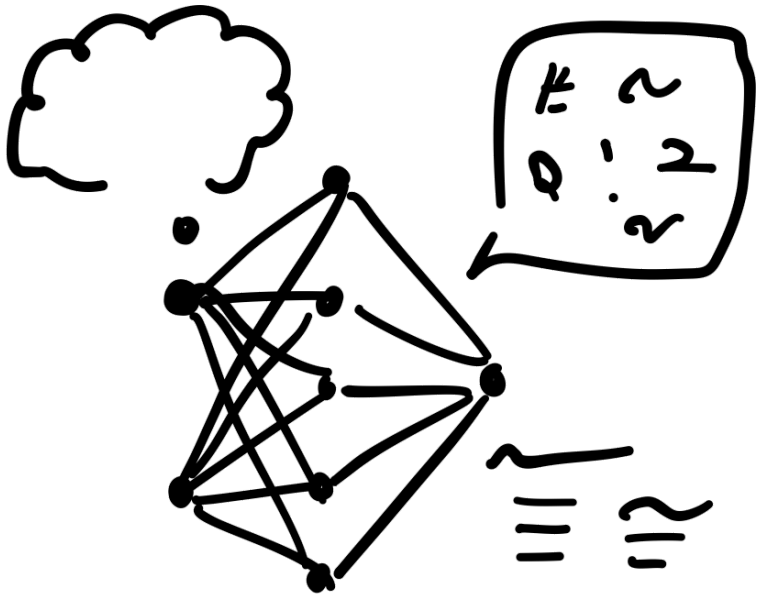
How can we interact with LLMs?



OpenAI

ANTHROP\C

How can we interact with LLMs?



OpenAI

ANTHROP\C

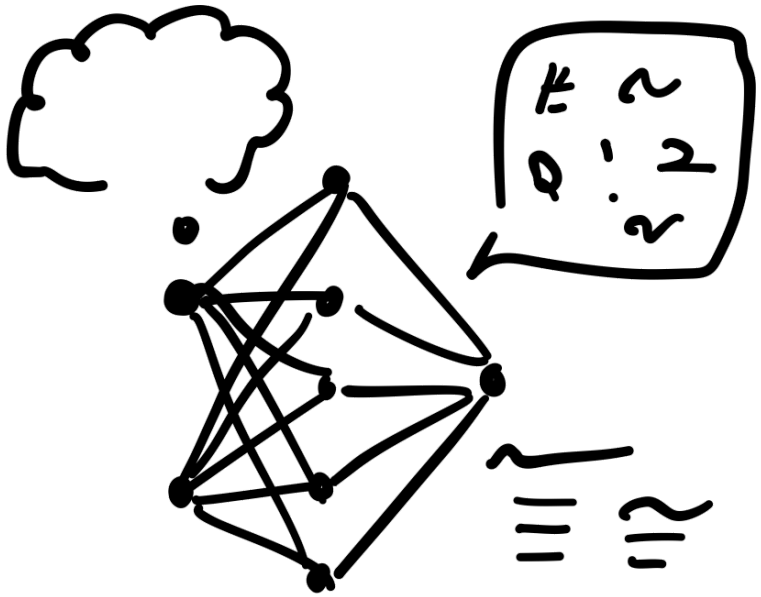
1. Web-based Chat Interface

2. Programmatic Access

3. Other interfaces

- mobile (chat) applications
- voice assistants
- embedded LLMs (e.g. suggestions in Gmail)

How can we interact with LLMs?



OpenAI

ANTHROP\C

1. Web-based Chat Interface

2. Programmatic Access

3. Other interfaces

Today, we will use web-based chat via:

- chatgpt.com
- claude.ai,

as well as programmatic access via:

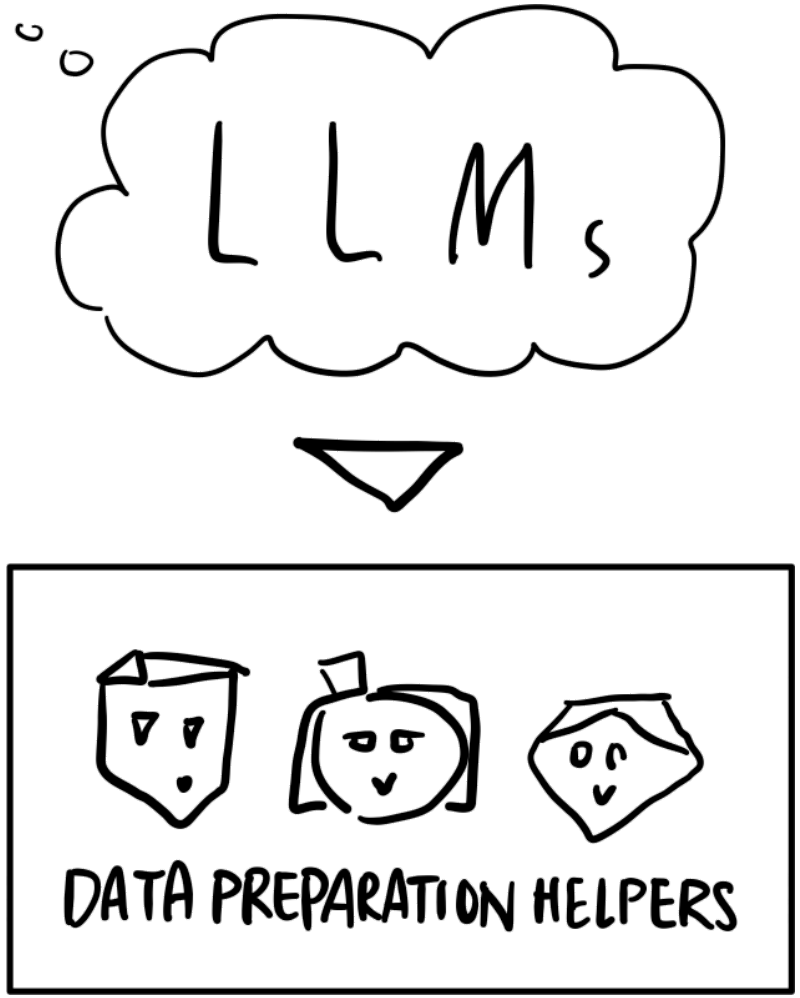
- `{ellmer}`

How do we verify what LLMs are doing?

For a dishwasher, we consider:

Using LLMs for “Wild Caught Data” Tasks

What are common WCD tasks?



Discuss in groups [5 mins]

- What tasks are involved in preparing Wild Caught Data for analysis? *List at least 3.*
- Which of these might be more or less suitable for addressing with LLMs? Why?

Using LLMs for Data Preparation Tasks

Using LLMs for Data Preparation Tasks

1. Generating data wranging code
2. Transformating and generating new data

Modifying and augmenting data

Prompts for Data Preparation Tasks

Starting Prompts

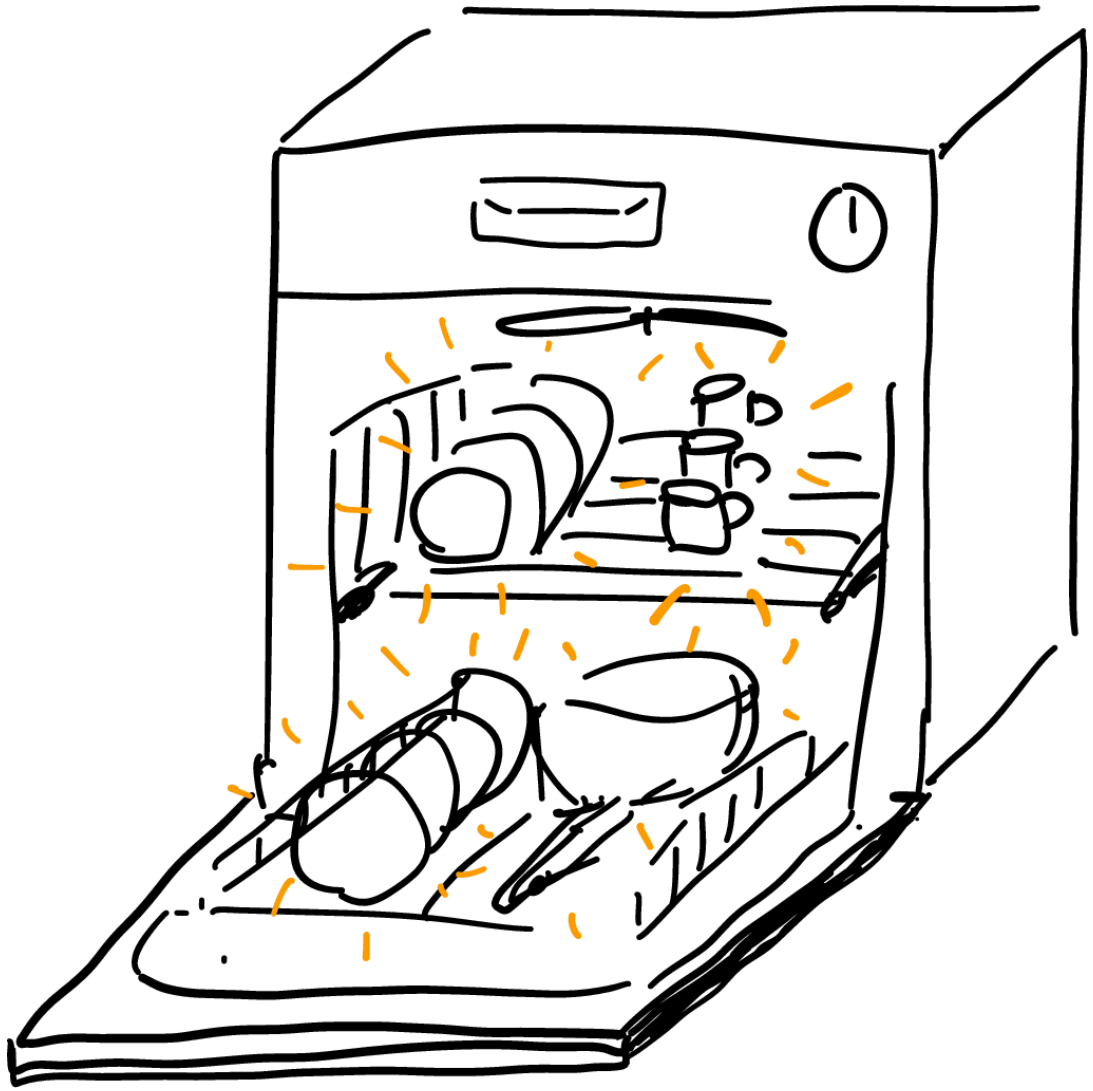
- **'look up' facts:** "What nationality is *the author* `<author name>`?"
- **suggesting values:** "What is the likely volume unit *of a beverage* `<can>` *of with* a volume of `<300>`?"
- **harmonise different abbreviations:** "Convert the following list of Australian states to all use three-letter state codes: `<list>`"
- **comparison:** "How similar are these two occupations: `<occupation A>`, `<occupation B>`?"
- **summary:** "Summarise the following survey response: `<text>`"
- **extraction**: "What movie is the follow review about": `<review text>`

Requesting different output formats

Verifying success

Verification is the most important skill when using LLMs, but it requires:

- Clearly defined tasks and expected outcomes
- Ways of checking the outcomes have been achieved



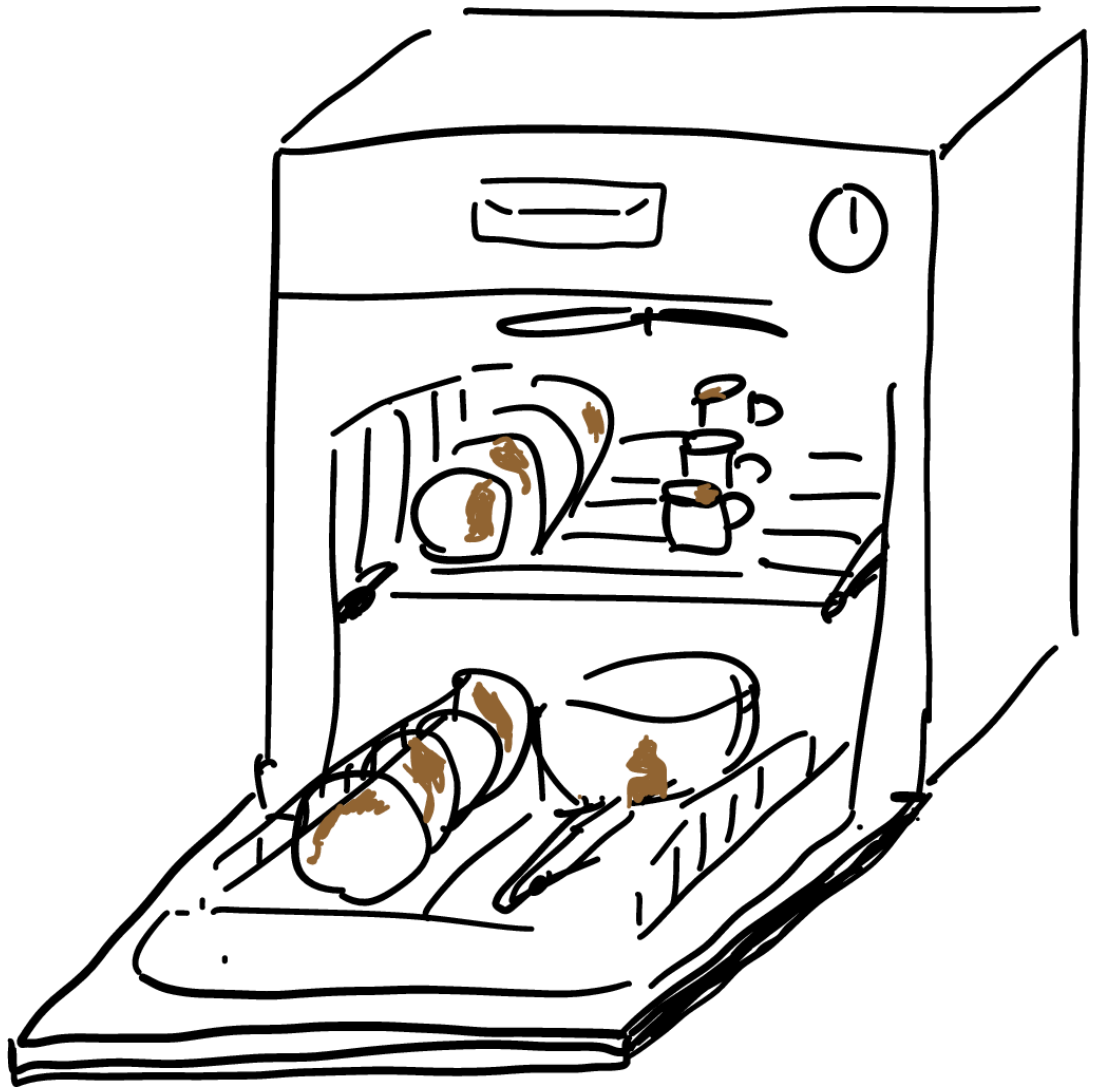
Approaches to verification

There are multiple ways to verify outcomes match expectations.

Checking on the dishes

There are multiple ways to verify outcomes match expectations.

- **Positive verification:**
- **Negative verification:**
- **Trust-based verification:**



Using LLMs in R with {ellmer}

Beyond web-based interfaces

Different interfaces mean different data preparation workflows:

Using the {ellmer} R package we can:

Connecting {ellmer} to an LLM

The basic steps:

LIVE DEMO: Chatting via {ellmer}

```r
library(ellmer)
## A session is like a chat conversation
session <- chat_openai()

question <- "
  How can I pick a random letter from A-Z.
"

## send a question to the 'chat'
session$chat(question)

## clarify your request
session$chat("Return R code only")

## inspect all turns in the session so far
session
```

LIVE DEMO: System Prompts

```r
library(ellmer)

session_tidy_expert <- chat_openai(system_prompt = "
  You are an expert R programmer
  who prefers the tidyverse.
  Only return code without explanation.
")

session_tidy_expert$chat(question)

session_tidy_expert
```

Example adapted from ellmer docs

Sessions and system prompts

Revisiting Author Nationalities

WCD 2025, S1, Week 10 - Slide 29

```
# A tibble: 11 × 3
   author_name
author_links                              nationality
   <chr>                                     <chr>
<chr>
 1 Rick Warren
https://en.wikipedia.org/wiki/R… <NA>
 2 William Griffith Wilson
https://en.wikipedia.org/wiki/B… <NA>
 3 F. Scott Fitzgerald
https://en.wikipedia.org/wiki/F… <NA>
 4 John Green
https://en.wikipedia.org/wiki/J… <NA>
 5 Sam McBratney
```

LIVE DEMO: Extract Nationalities

```r
text <- "Jane Austen (/ˈɒstɪn, ˈɔːstɪn/ OST-in, AW-stin; 16 December 1775 – 18 July 1817) wa

session_read <- chat_openai(system_prompt = "You are a data entry assistant.")

session_read <- chat_openai()

nationality_prompt <- "Nationality of person"

session_read$extract_data(text, type = type_string(description = nationality_prompt))

std_prompt <- "Extract structured data of the nationality of person. Return only ISO 3-digit

session_read$extract_data(text, type = type_string(description = std_prompt))
```

LIVE DEMO: Ask for Nationalities

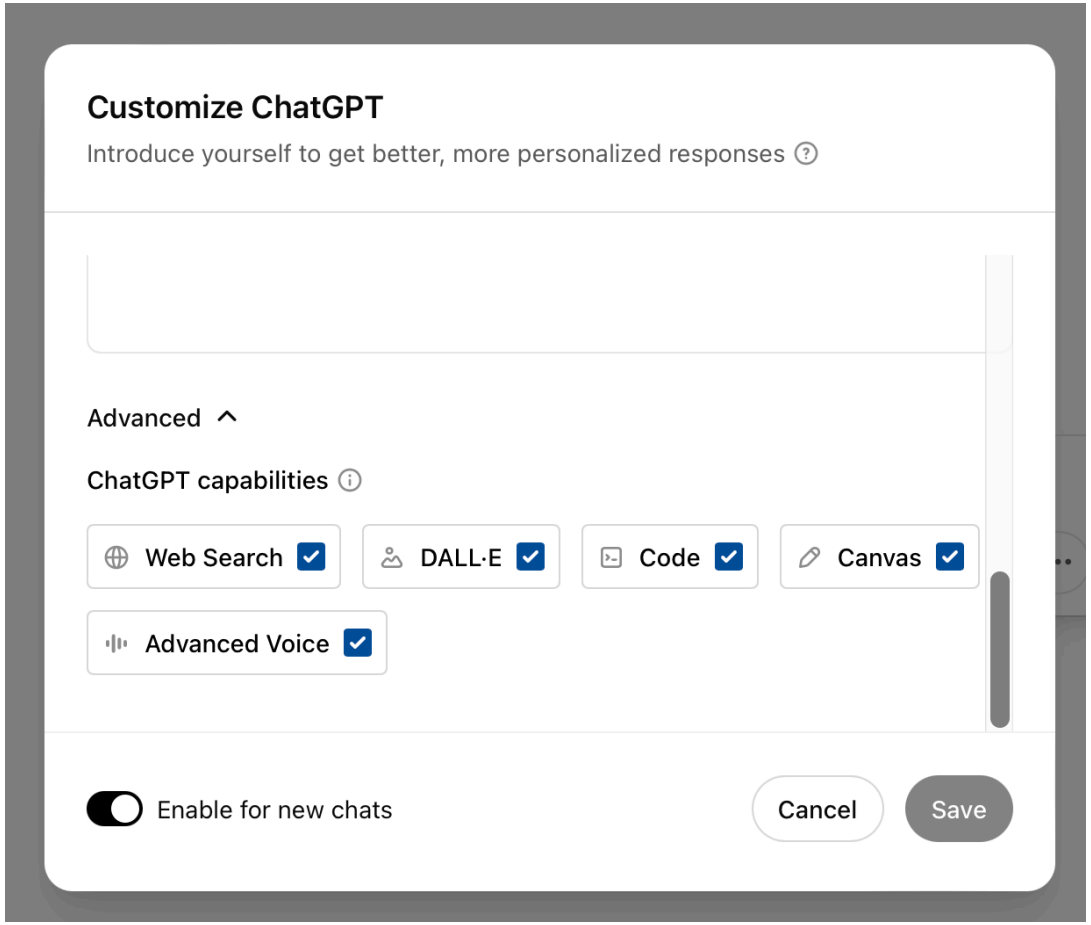
```r
library(dplyr)

author_df <- readr::read_csv('example_data/week10-author_df.csv')

short_prompt <- "Nationality of person only"
session_lib <- chat_openai(system_prompt = "You are a librarian with expert knowledge of pop

## let's ask about multiple authors
author_df |>
  tail(6) |>
  rowwise() |>
  mutate(nationality_llm =
           session_lib$clone()$extract_data(author_name,
                                            type = type_string(short_prompt))
    )
```

Evaluation through agreement

Another way to verify data quality is via **consensus**.

How much do LLMs ‘know’?

LLM agents, web search and more



Final Comments

Ethics and AI safety

Generative AI acknowledgement

I used AI in the following ways:

i. generate definitions and suggested explanations for key concepts covered in this lecture. I used Claude AI to suggest definitions for terms like ‘Generative AI’, and ‘System Prompt’, and to generate lists of “top LLM providers in 2025” and “ways of interacting with LLMs”.

Key takeaways

What we've learnt...