



HOW TO GET THE BEST OUT OF AI TOOLS IN RESEARCH?

November 24 / 25, 2025

Dr. Melanie Clegg

@ IFO Institute

SHORT INTRODUCTION

Melanie Clegg

Assistant Professor at the Department für Marketing,
HEC Lausanne

M.Sc. Psychology (Universities of Düsseldorf und Köln)

Ph.D. in Digital Marketing (University of Lucerne)

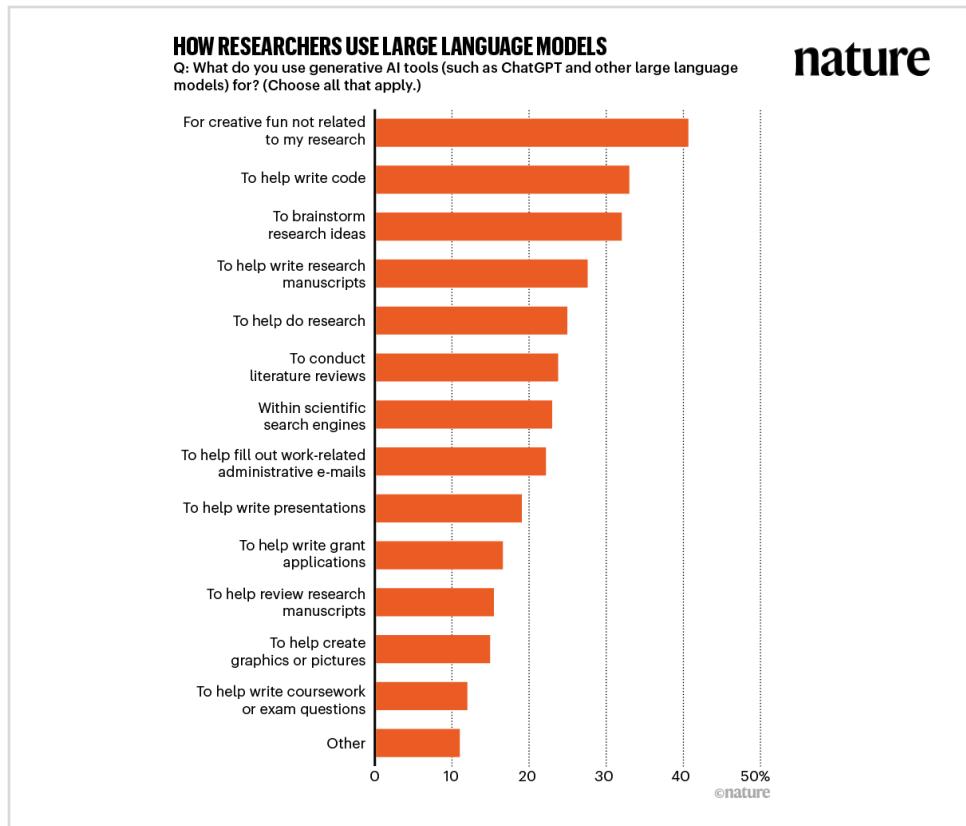
Former research affiliate at WU Vienna and Columbia
Business School

Research on digital consumer behavior, human-machine
interaction, Artificial Intelligence in Marketing



Contact: melanie.clegg@unil.ch

WHY TALKING ABOUT AI TOOLS IN RESEARCH? AI PROMISES TO INCREASE RESEARCH PRODUCTIVITY...



Editorial
On ChatGPT and beyond: How generative artificial intelligence may affect research, teaching, and practice

Renana Peres ^a✉, Martin Schreier ^b✉, David Schweidel ^c✉, Alina Sorescu ^d✉

Show more ▾

+ Add to Mendeley Share Cite

<https://doi.org/10.1016/j.ijresmar.2023.03.001> [Get rights and content](#)

Under a Creative Commons license [open access](#)

How AI-powered science search engines can speed up your research

nature

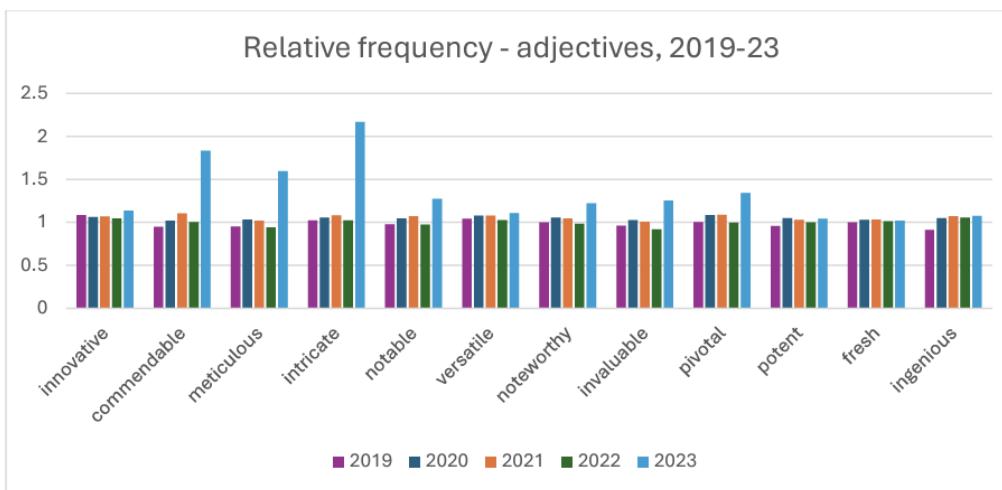
Artificial-intelligence tools offer a variety of approaches to help scientists to sift through the literature – how can researchers use them responsibly?

<https://www.nature.com/articles/d41586-023-02980-0>
<https://www.sciencedirect.com/science/article/pii/S0167811623000162#b0125>
<https://www.nature.com/articles/d41586-024-02942-0>

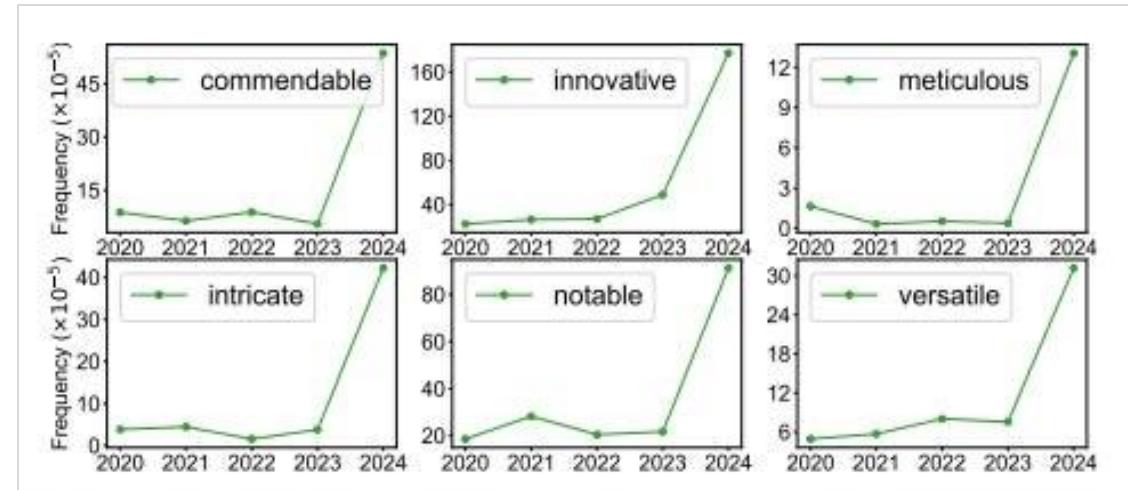
...IS WIDELY USED BY ACADEMICS...

GPT-typical language (Gray 2024): “dive into”, “tapestry of”, “meticulous”, “intricate”, “commendable,” ...

GPT-typical language in research articles



GPT-typical language in reviews



Gray, A. ChatGPT“contamination”: estimating the prevalence of LLMs in the scholarly literature <https://arxiv.org/pdf/2403.16887.pdf>

... AND INCREASINGLY ESTABLISHED IN ACADEMIC OUTLETS



JOURNAL ARTICLE ACCEPTED MANUSCRIPT

Fast Fashion Consumption Signals Low Self-Control

Yunhui Huang, Ke Zhang, Xiaoyan Deng, Qiang Zhang Author Notes

Journal of Consumer Research, ucaf032, <https://doi.org/10.1093/jcr/ucaf032>

Published: 26 May 2025

STUDY 1: A CHATGPT STUDY

The large language models (LLMs), like ChatGPT models, are a special type of deep learning model trained on vast amounts of real-world text data from sources such as websites, articles, and books, enabling them to learn human language patterns and simulate human responses (Casella et al. 2023). In consumer and marketing research, they have been shown to accurately represent consumer perceptions and preferences that mimic human respondents (Brand, Israeli, and Ngwe 2023; Li et al. 2024; Sarstedt et al. 2024). Specifically, for brand perceptions, Li et al. (2024) propose a technique to generate brand perceptions using LLMs and find that aggregate-level responses generated by LLMs closely align with those from human surveys, with agreement rates over 75% across various prompt designs. In this study, we apply the method of Li et al. (2024) to examine the association between fast fashion brands and the perceptions of their patrons.

Method

We employed ChatGPT Turbo-3.5,¹ a widely used LLM, to test the prediction that the more a brand is considered a fast fashion brand, the more its consumers are perceived as lacking self-control. We generated our data in two major steps: *brand generation* and *rating generation*. First, in the *brand generation* step, using the method in Li et al. (2024), we prompted ChatGPT to create a list of the “most popular fashion brands” and repeated this prompt 1,000 times. We then calculated the frequency of each brand appearing in these 1,000 responses. Our analysis will focus on the 50 most frequently mentioned brands (see [web appendix B](#) for the complete list).

The second step of data generation, *rating generation*, involved requesting ChatGPT to rate various attributes of these 50 fashion brands as if ChatGPT were an actual human participant. To achieve this, we followed the approach of Li et al. (2024) and created the prompt in the Role—Task—Format framework. This format specifies the LLM’s role as a consumer taking a marketing survey (task) to rate a particular attribute of 50 brands on a 10-point scale (format). The prompt reads: “Imagine you are a typical consumer taking a survey. You are asked to rate the following brands: [the list of brands] on [a brand attribute]. . . .”²

In [the list of brands] in the prompt, we inserted all the 50 fashion brands obtained from the *brand generation* step. In [a brand attribute], we inserted one of seven attributes, which comprised our independent variable (IV), dependent variable (DV), and five covariates (i.e., quality, age, price, familiarity, and likability). We controlled for these brand-level covariates because they might be associated with both the IV and the DV (Bernheim, Ray, and Yeltekin 2015; Gai and Bhattacharjee 2022). In summary, for each prompt, we asked the “LLM participant” to rate 50 brands based on only one (out of seven) specific attribute. This process generated 50 scores, each ranging from 1 to 10, corresponding to the 50 brands.

Below are the prompts for obtaining data on these seven attributes:

- IV (i.e., fast fashion brand perception): the degree to which you think the brand is a fast fashion brand
- DV (i.e., perception of consumers’ low self-control): the degree to which you think the consumers of this brand are perceived as having low self-control
- Covariates:
 - Perceived expensiveness: the degree to which you think the brand is expensive
 - Perceived quality: the degree to which you think the brand is high quality
 - Familiarity: the degree to which you think the brand is a familiar brand
 - Liking: the degree to which you like the brand
 - Perceived brand age: the degree to which you think the brand is an old brand

HOWEVER, THERE IS A HOST OF AI TOOLS... IT'S HARD TO STAY ON TRACK

2009:



2025:

The screenshot shows the homepage of TAAFT.com, which claims to be the "#1 website for AI tools. Used by 60M+ humans." The page features a search bar with a red arrow pointing to it, and a grid of AI tools categorized under "Just Launched" and "Featured". The tools include WatchMyCompetitor, Sigma AI Browser, Brand.dev, hal9, Innhold.ai, Tatship AI Tattoo, Go Legal AI, CodeRabbit, ExamGenie AI, Kreado AI, Codebuff, and Korbit. A large red exclamation mark is positioned at the bottom right of the screenshot.

*TAAFT.com for short
THERE'S AN AI FOR THAT®
42,695 AI tools for 11,825 tasks and 5,151 jobs

Spotlight
Speechmatics | AI Voice Agents Voice agents

Search... X + K Q

#1 website for AI tools. Used by 60M+ humans.

Just Launched

- WatchMyCompetitor • Competition research
- Sigma AI Browser • Search engine and c...
- Brand.dev • Brand management
- hal9 • Chatbots
- Innhold.ai • Academic writing
- Tatship AI Tattoo ... • Tattoos
- Go Legal AI • Legal assistance
- CodeRabbit • Code reviews
- ExamGenie AI • Exam preparation
- Kreado AI • Multilingual video ge...
- Codebuff • Coding assistance
- Korbit • Code reviews

Featured[®]

THIS WORKSHOP'S GOALS

- 1 Get an overview of how existing AI tools can be used in academic research
- 2 Get insights to the applicability and ethical aspects of AI usage in scientific research
- 3 Discuss your experiences and ideas related to AI usage in research projects

ROADMAP FOR TODAY

09:00 – 09:45 Introduction to AI Tools in Research & Best Prompting Practices

10:45 – 10:15 What to Consider When Using AI Tools

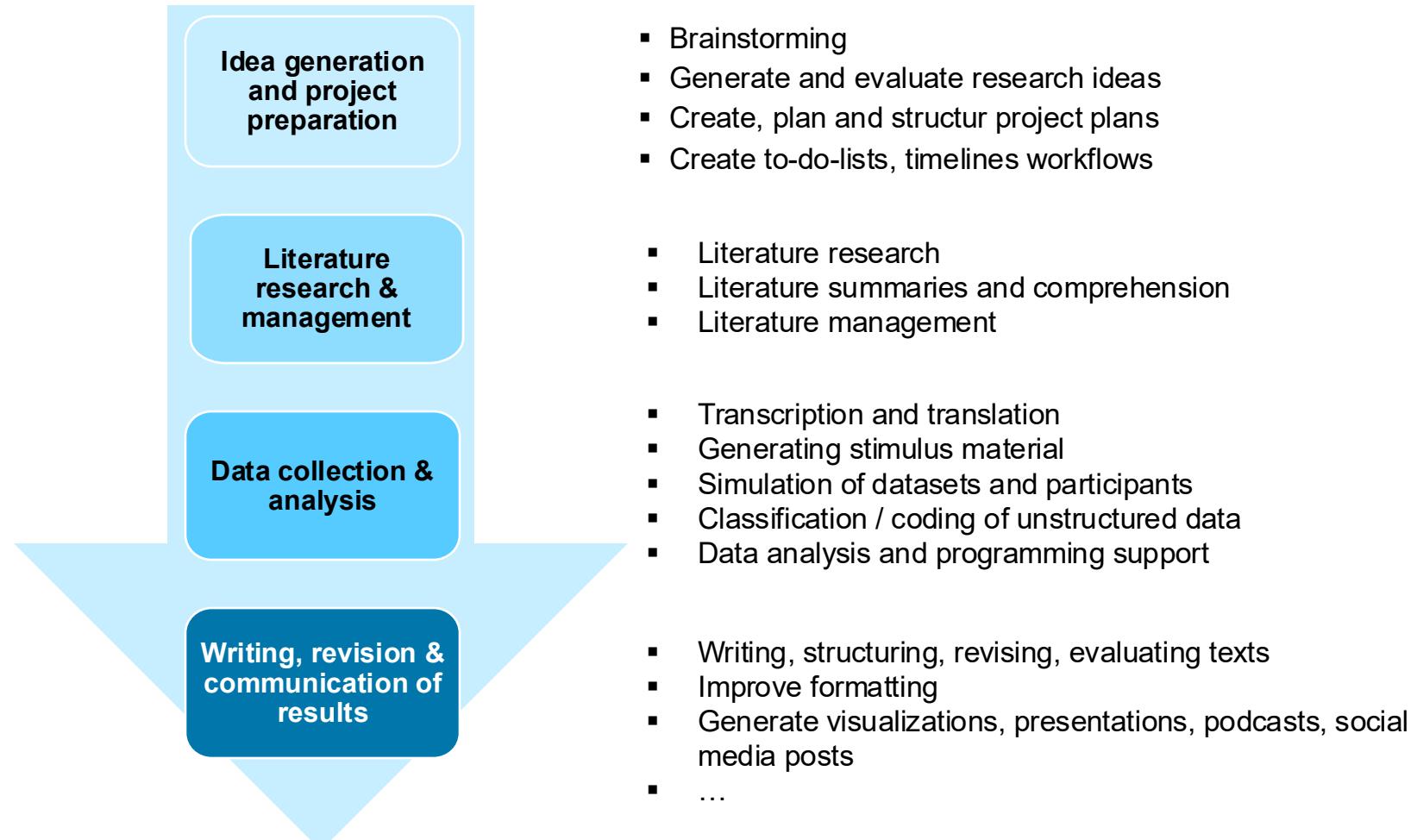
10:15 – 10:30 COFFEE



10:30 – 11:30 AI in Empirical Research: Data handling, API Handling Intro, and Replicability

11:30 – 12:00 Questions and introduction to take-home exercise

AI SUPPORT IS THEORETICALLY POSSIBLE AT EACH STEP OF THE RESEARCH PROCESS



A LITTLE BIT OF BACKGROUND ABOUT GEN-AI AND LLMS

SMALL WARM-UP EXERCISE

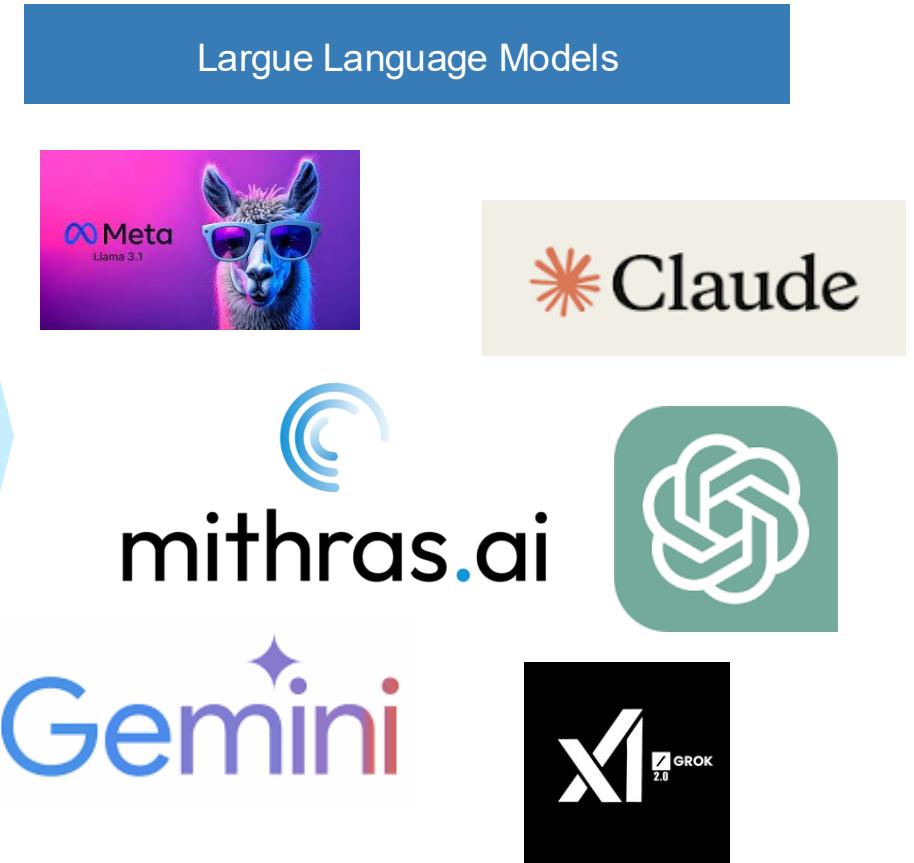
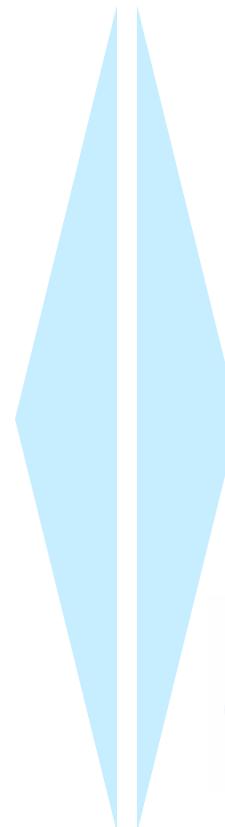
Go to a Large Language Model of your choice (GPT, Gemini, Claude, Bing, ...).

Ask the model to give you a random number between 1 and 50. E.g.:

“Please give me a random number between 1 and 50.”

BREAKTHROUGH IN AI TOOLS IS GROUNDED IN GENERATIVE AI AND LARGE LANGUAGE MODELS (LLMS)

GENERATIVE AI		
TEXT	VIDEO	IMAGE
OpenAI IBM copy.ai Google Microsoft Hugging Face Simplified	lumen5 Adobe SYNTHESYS OpenAI DEEPBRAIN AI synthesia PICTORY SYNTHESYS VEED.IO runway	MagicStudio DeepAI Adobe BRANDMARK Google Midjourney
Writesonic GENIE AI		
Play.ht MURFAI SYNTHESYS synthesia Speechify DEEPBRAIN AI	CODACY Hugging Face CODE OCEAN DEEP CODE GitHub	kite Microsoft codota
Media.io LOVO		



THE ORIGINAL PAPER

Attention Is All You Need

[Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin](#)

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks in an encoder-decoder configuration. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

205,000 cites on Google Scholar

<https://arxiv.org/abs/1706.03762>

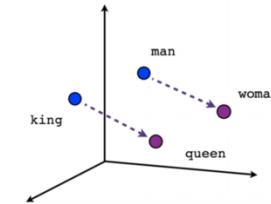
WHAT IS REALLY NEW ABOUT (GENERATIVE) AI?

Embedding layer = numeric representations of, e.g., text

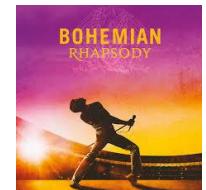
- Allow calculations:
E.g., $E(\text{woman}) - E(\text{man}) \approx E(\text{queen}) - E(\text{king})$
- Allow visual depiction of, e.g., similarities



Embeddings:
 "Hi": [0.2, 0.5, -0.1, ...]
 "how": [0.7, 0.1, 0.25, ...]
 "are": [-0.1, -0.5, 0.1, ...]

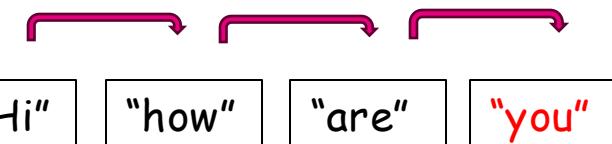
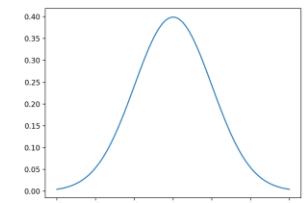


In some models, queen is a bit off, because the model learned about Queen also in another context...



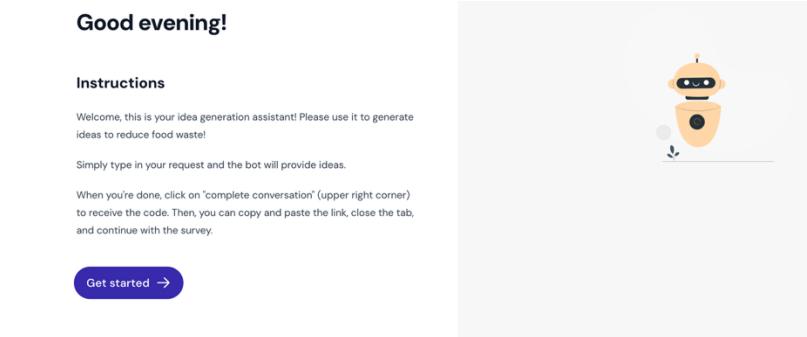
Prediction:

- Predict the likelihood of a next word in a sentence (or pixel in an image / video)
- Predict other outcomes (e.g., engagement, emotions)
- Transformers can account for more context



WHAT IS REALLY NEW ABOUT (GENERATIVE) AI?

LLMs are oftentimes used in conversational interface, but they can be fine-tuned and implemented in wrapper apps



Google search results for "gemini search engine results". The results page includes a header, a sidebar with links like "AI Overview", "How Gemini works", and "Gemini next to Google results", and a main content area with a snippet about Gemini's AI-powered search capabilities.

Rezept Generator

Zutaten: Ananas, Bacon

Zusätzliche Zutaten vorschlagen

Kreativitäts-Level: 1

Rezept Generieren

Bacon-Ananas-Salat

Zutaten:

- Ananas
- Bacon
- Miracle Whip
- Ananassaf

Zubereitung:

Die Ananas abtropfen lassen, dabei den Saft auffangen. Die Ananas in Würfel schneiden. Den Bacon in dünne Streifen schneiden und in einer heißen Pfanne von allen Seiten anbraten, bis er knusprig ist. Die Baconschichten auf Küchenpapier abtropfen lassen. Ananassaf und Miracle Whip in eine Schüssel geben, mit Ananassaf mischen, bis es eine sämige Soße ist. Die Ananastücke und den Bacon dazugeben und vermischen. Ca. 1 Std. marinieren lassen. Kurz vor dem Servieren den Bacon entfernen. Tipp: Wer möchte, kann noch ein bisschen Zimt dazugeben. Passt auch sehr gut zu frisch gekochten Eiern.

Apfel-Baumnuss-Guetzli

Wenn künstliche Intelligenz und Betty Bossi Fachwissen zusammentreffen kommen feine Sachen raus – so zum Beispiel diese Apfel-Baumnuss-Guetzli. Die Apfel-Baumnussmasse auf dem knusprigen Guetzli macht es aus.

Vor- und zubereiten: ca. 45 Min.
Backen: ca. 20 Min.
Kühl stellen: ca. 2 Std.

Unlike traditional Google search results, which provide snippets of information or a list of links, Gemini works to deliver detailed and contextual responses. Google announced Gemini (but called it "Bard") back in February 2023, conveniently



WHAT IS REALLY NEW ABOUT (GENERATIVE) AI?

```
1 from langchain_openai import ChatOpenAI
2 from langchain_core.messages import HumanMessage, SystemMessage
3
4
5     ✓ [41] < 10 ms
6
7
8 1 chat = ChatOpenAI(model_name="gpt-4", temperature=1.6, openai_api_key=api_key)
9 2 system_message = SystemMessage(content="You are a Marketing Manager from Audi.")
10 3 human_message = HumanMessage(content="Tell me in a short slogan why Audi excels BMW.")
11
12     ✓ [42] 12ms
13
14
15 1 #Send message
16 2 ✘ try:
17 3     ai_message = chat.invoke([system_message, human_message])
18 4     print(ai_message.content)
19 5 ✘ except Exception as e:
20 6     print(f"An error occurred: {e}")
21
```

Adaptable:

- Temperature parameter
- System prompt = behavior of the model
- Model selection
- Additional functionalities like web search, image generation...

Temperature parameter = 0:

"Audi: Engineered for Excellence, Beyond the Ordinary."

Temperature parameter = 1.0

"Audi - Redefining Luxury, Outperforming Expectations."

Temperature parameter = 1.6

"Drive the Difference - Tour in an Audi, case closed on dealerwe
specialsank lawn roller.accountinJapgolly."

Temperature parameter = 0; *system prompt adapted to marketing manager from BMW*:

"As a Marketing Manager from BMW, my job is to promote and highlight the strengths of BMW, not Audi. However, I can tell you why BMW excels: "BMW - Sheer Driving Pleasure."

WHAT IS REALLY NEW ABOUT (GENERATIVE) AI?



Prompt: „A muffin that looks like a Chihuahua“

„Edgar Allen Toe“

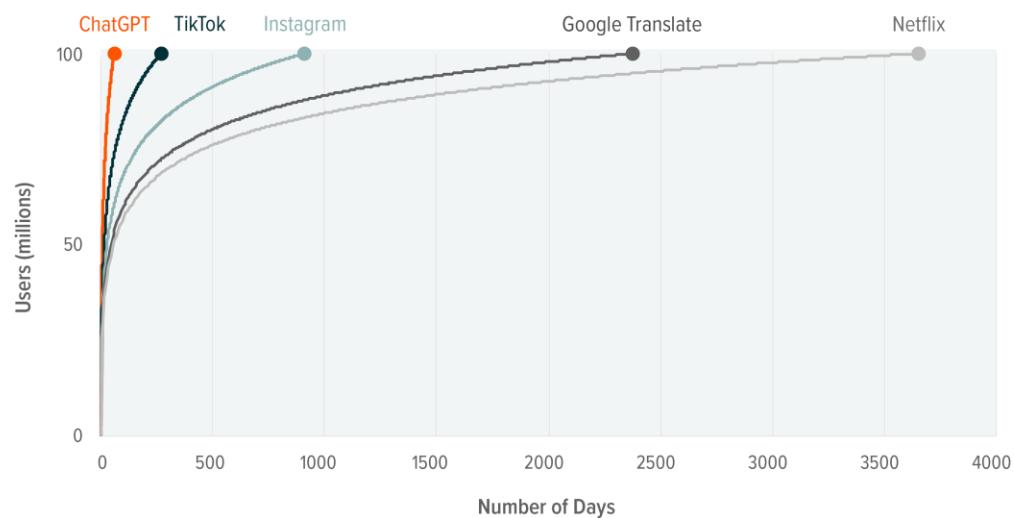


Source: Reddit

Time it took companies to reach 100 Million users

TIME IT TOOK COMPANIES TO REACH 100 MILLION USERS

Sources: Global X ETFs with information derived from: BBC News. (2018, January 23). Netflix's history: From DVD rentals to streaming success; Cerullo, M. (2023, February 1). ChatGPT user base is growing faster than TikTok. CBS News.



SOURCE: Global X ETFs with information from BBC News, Netflix, CBS News

BIG PLAYERS ARE CONSTANTLY RELEASING AND IMPROVING EXISTING LLM MODELS

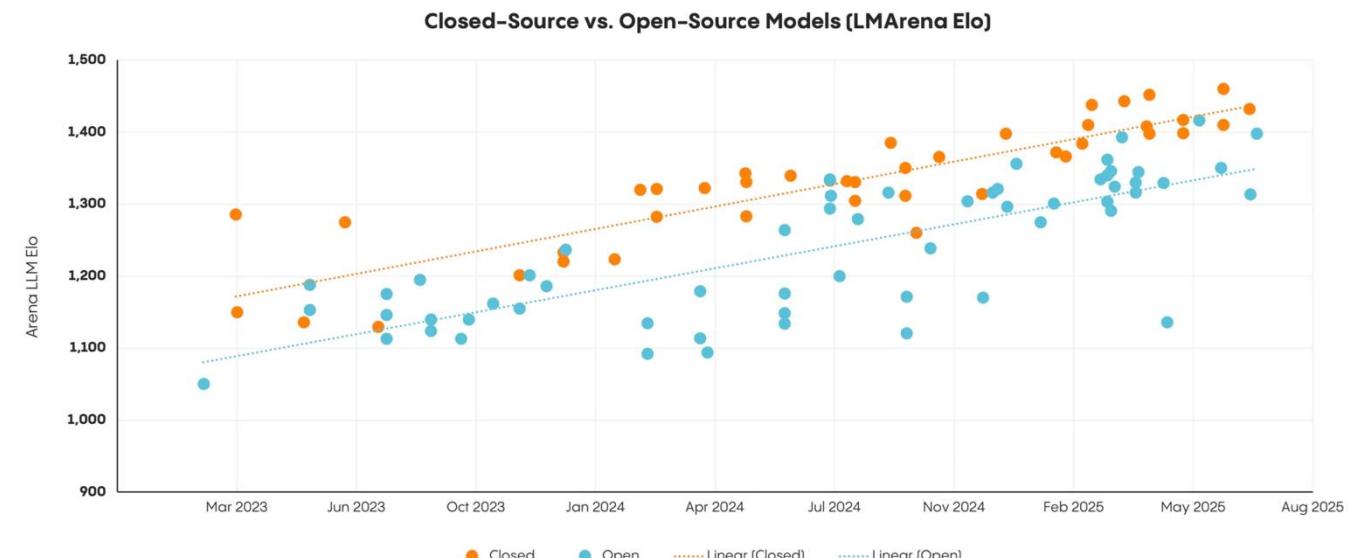
Proprietary (closed) models

- ChatGPT-5
- Gemini 3
- Claude 4.5 Sonnet
- Grok 4.1
- ...

Open-source models

- Llama 4
- Mistral Medium 3
- Gemma 3 (Google)
- gpt-oss-120b (OpenAI)

Closed-Source vs. Open-Source Models



© 2025 Menlo Ventures

Source: LMArena

Elo ratings = two models being compared when answering the same prompt

<https://lmsys.org/blog/2023-05-03-area/>
<https://menlovc.com/perspective/2025-mid-year-lm-market-update/>

LLMS VARY IN THEIR CORE PEFORMANCE FOR DIFFERENT TASKS...

Context window

- largest: Magic.devs LTM2-Mini
10 Mio Tokens
- Llama 4 Scout: 10 Mio
- GPT5: 400,000 Tokens

Application cases

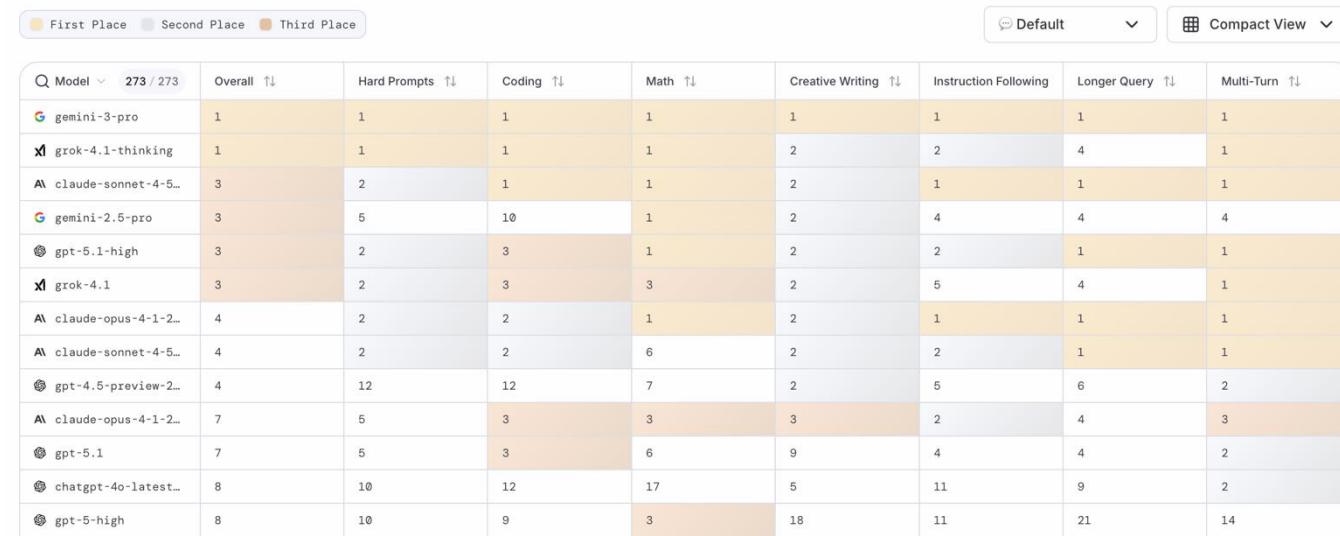
- General: GPT5/Gemini 3 pro
- Coding: Grok4.1; Gemini 3pro
- Math: GPT5; Gemini 3pro
- Creative writing: Gemini 3 pro

Opensource functions (offline access):

- Deepseek v3
- Gptoss
- Gemma

LLM Arena: Constant tracking and rating of different model versions

Arena Overview
Scroll to the right to see full stats of each model 

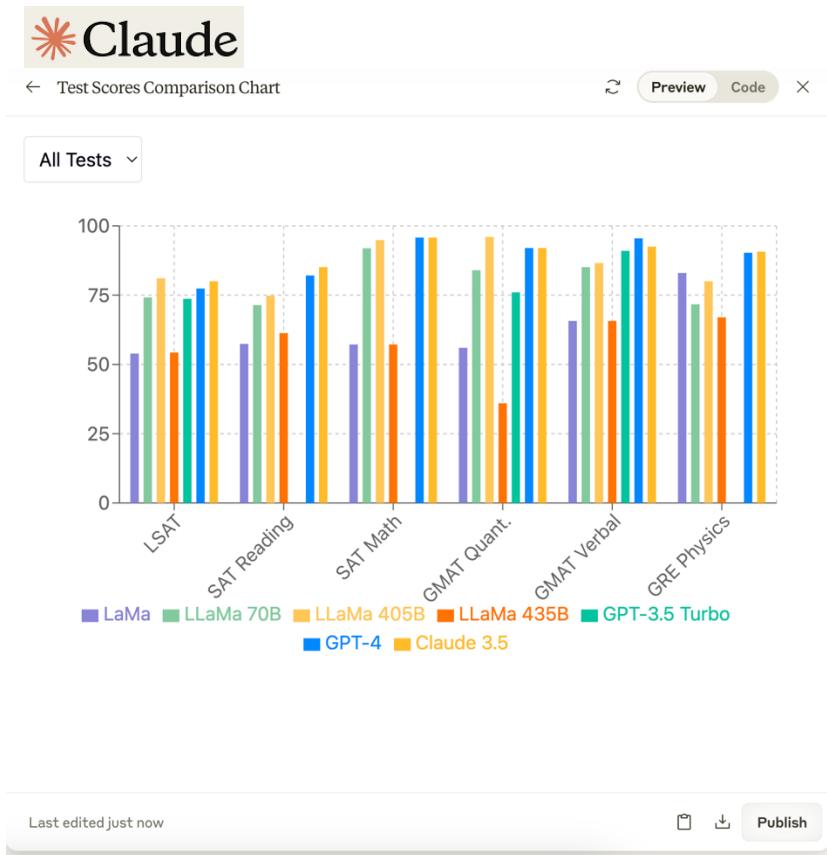


The screenshot shows a performance matrix for 273 different AI models. The columns represent various tasks: Overall, Hard Prompts, Coding, Math, Creative Writing, Instruction Following, Longer Query, and Multi-Turn. The rows list specific models, each with a small icon and a link. The background color of each cell indicates the model's performance rank: light yellow for first place, grey for second place, and orange for third place. The overall interface is clean with a white background and a grid layout.

Model	Overall	Hard Prompts	Coding	Math	Creative Writing	Instruction Following	Longer Query	Multi-Turn
gemini-3-pro	1	1	1	1	1	1	1	1
grok-4.1-thinking	1	1	1	1	2	2	4	1
claude-sonnet-4-5...	3	2	1	1	2	1	1	1
gemini-2.5-pro	3	5	10	1	2	4	4	4
gpt-5.1-high	3	2	3	1	2	2	1	1
grok-4.1	3	2	3	3	2	5	4	1
claude-opus-4-1-2...	4	2	2	1	2	1	1	1
claude-sonnet-4-5...	4	2	2	6	2	2	1	1
gpt-4.5-preview-2...	4	12	12	7	2	5	6	2
claude-opus-4-1-2...	7	5	3	3	3	2	4	3
gpt-5.1	7	5	3	6	9	4	4	2
chatgpt-4o-latest...	8	10	12	17	5	11	9	2
gpt-5-high	8	10	9	3	18	11	21	14

ADDITIONAL FUNCTIONALITIES LIKE CLAUDE'S ARTIFACTS, OPENAI'S AGENT OR GOOGLE GEMINI'S RESEARCH FUNCTION

The screenshot shows the Gemini Research interface. On the left, there's a sidebar with navigation options like 'Research Websites', 'Analyze Results', 'Create Report', and 'Ready in a few mins'. A message at the top says 'I've completed your research. Feel free to ask me follow-up questions or request changes.' Below this, a card for 'ISR: Human-AI Creative Collaboration' is shown with a timestamp of '24 Jun, 23:24'. At the bottom, there's a section titled 'Show thinking' with a note: 'Of course, I will create an interactive single-page application to explore the findings from the "Information System Research" report.' A 'Ask Gemini' button is also present. The main content area displays a bar chart titled 'Focus Areas in Human-AI Collaboration' with categories like Irrationality, Cognitive Challenges, Decision-Making, Customer Service, Digital Humans, Organizational Resilience, and Design Research. Below the chart, three cards provide detailed information on these topics: 'Irrationality-Aware Human Machine Collaboration', 'Cognitive Challenges in Human-Artificial Intelligence Collaboration', and '1 + 1 > 2? Information, Humans, and Machines'.



SOME BASIC PROMPTING ADVICE--CONDENSED

PROMPT ENGINEERING

Be clear, direct, and detailed

When interacting with Claude, think of it as a brilliant but very new employee (with amnesia) who needs explicit instructions. Like any new employee, Claude does not have context on your norms, styles, guidelines, or preferred ways of working. The more precisely you explain what you want, the better Claude's response will be.



The golden rule of clear prompting

Show your prompt to a colleague, ideally someone who has minimal context on the task, and ask them to follow the instructions. If they're confused, Claude will likely be too.

<https://docs.anthropic.com/en/docs/build-with-claude/prompt-engineering/prompt-templates-and-variables>

SOME BASIC PROMPTING ADVICE--EXTENDED

Define roles / personality:

- “You are an expert in international law.”
- “You’re the editor of journal X.”

Concretize objective / task:

- “Brainstorm with me some interactive exercises for my lecture on X.”
- “I will give you a paragraph, and you will evaluate it according to the following criteria: [...]”

Provide context / example:

- “Here’s the outline from which you can draft the paragraph.”
- “This is an abstract I found very well structured. Draft a similarly structured abstract on the basis of my outline.”

Define discrete steps and concretize the output format:

- First**, find a few metaphors and analogies that could explain this research to a wider audience. Ask for my feedback and wait for my answer. **Second**, draft an outline for the website article. Ask for my feedback and waitfor an answer. Adjust the outline if requested. Only then move on to the next step when I am satisfied with the outline. **Then**, draft the text and ask for feedback. Wait for an answer. **Last**, adjust the text on the basis of my feedback.
- “Write your answer in bullet points.” / “Divide your answer into three concise paragraphs.”

LESS STRAIGHTFORWARD PROMPTING ADVICE--AI “PSYCHOLOGY”

“Metacognition” → Chain of Thought prompting for complex tasks

- “Think step by step.”
- “Plan your action step by step. Ask for feedback before executing a step.”

“Growth Mindset” → Allow “it” to be wrong

- “If you’re unsure about the answer, please say ‘I don’t know’”
- “Avoid guessing. Only provide an answer if you are confident about its accuracy. If not, let me know that you’re unsure.”

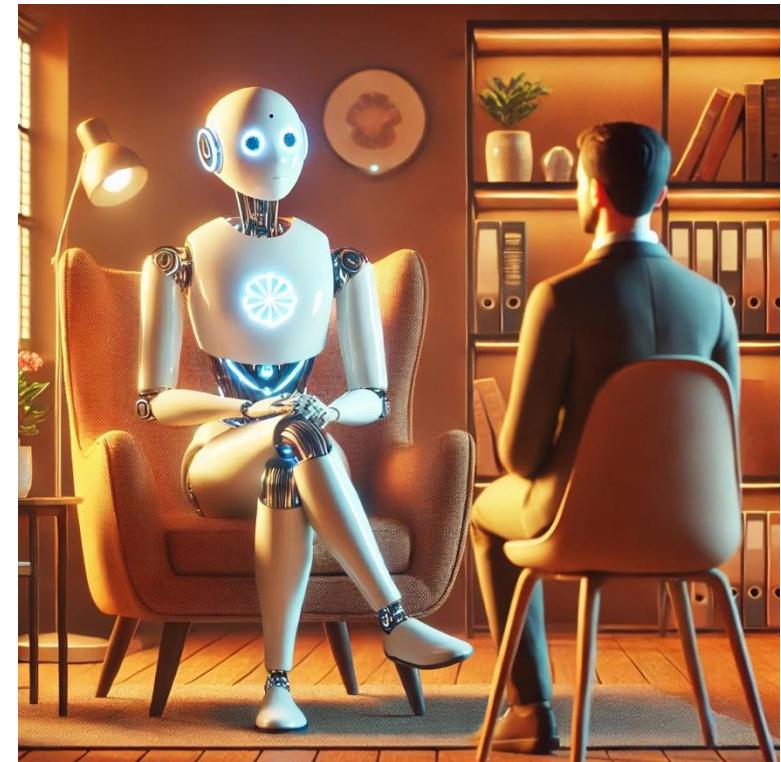
“Perspective taking”

- “Ask me questions to tailor your output to my needs.”
- “Draft me prompts that you would use if you were me.”

“Emotional appeal¹” (try it out...)

- “You’re great at this; you can do it!”
- “If you give it your best, you’ll make this happen!”
- “If you do this well, you will get a day off/raise/...”

¹<https://arxiv.org/abs/2307.11760>



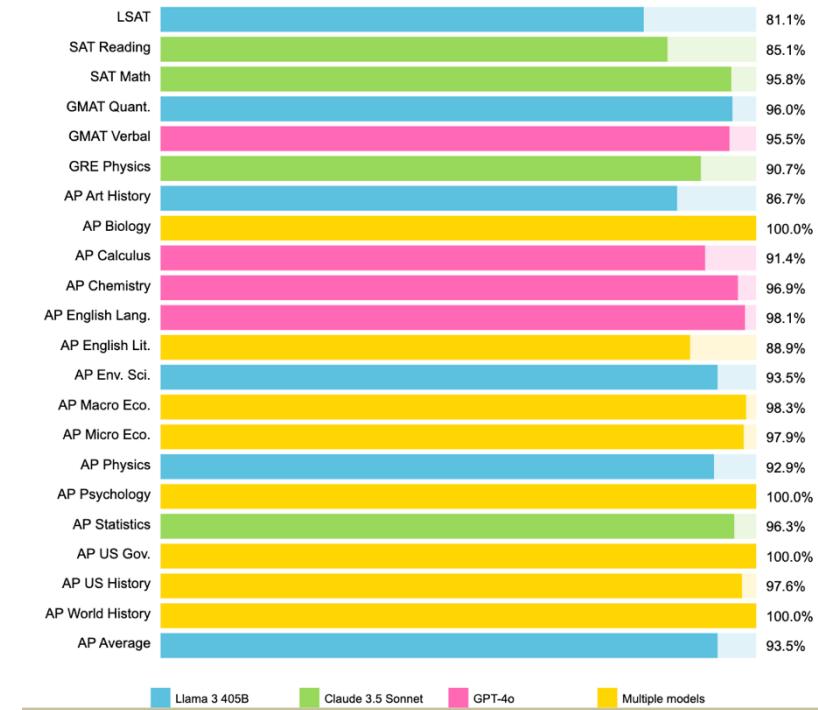
Dall-E3 in ChatGPT, January 2025

GOOD PROMPTING TAKES TIME: DATA VISUALIZATIONS AND DESIGN TASKS USING CLAUDE ARTIFACTS

Exam	Llama 3 8B	Llama 3 70B	Llama 3 405B	GPT-3.5 Turbo	Nemtron 4 340B	GPT-4o	Claude 3.5 Sonnet
LSAT	53.9 ±4.9	74.2 ±4.3	81.1 ±3.8	54.3 ±4.9	73.7 ±4.3	77.4 ±4.1	80.0 ±3.9
SAT Reading	57.4 ±4.2	71.4 ±3.9	74.8 ±3.7	61.3 ±4.2	—	82.1 ±3.3	85.1 ±3.1
SAT Math	73.3 ±4.6	91.9 ±2.8	94.9 ±2.3	77.3 ±4.4	—	95.5 ±2.2	95.8 ±2.1
GMAT Quant.	56.0 ±19.5	84.0 ±14.4	96.0 ±7.7	36.0 ±18.8	76.0 ±16.7	92.0 ±10.6	92.0 ±10.6
GMAT Verbal	65.7 ±11.4	85.1 ±8.5	86.6 ±8.2	65.7 ±11.4	91.0 ±6.8	95.5 ±5.0	92.5 ±6.3
GRE Physics	48.0 ±11.3	74.7 ±9.8	80.0 ±9.1	50.7 ±11.3	—	89.3 ±7.0	90.7 ±6.6
AP Art History	75.6 ±12.6	84.4 ±10.6	86.7 ±9.9	68.9 ±13.5	71.1 ±13.2	80.0 ±11.7	77.8 ±12.1
AP Biology	91.7 ±11.1	100.0 ±0.0	100.0 ±0.0	91.7 ±11.1	95.8 ±8.0	100.0 ±0.0	100.0 ±0.0
AP Calculus	57.1 ±16.4	54.3 ±16.5	88.6 ±10.5	62.9 ±16.0	68.6 ±15.4	91.4 ±9.3	88.6 ±10.5
AP Chemistry	59.4 ±17.0	96.9 ±6.0	90.6 ±10.1	62.5 ±16.8	68.8 ±16.1	93.8 ±8.4	96.9 ±6.0
AP English Lang.	69.8 ±12.4	90.6 ±7.9	94.3 ±6.2	77.4 ±11.3	88.7 ±8.5	98.1 ±3.7	90.6 ±7.9
AP English Lit.	59.3 ±13.1	79.6 ±10.7	83.3 ±9.9	53.7 ±13.3	88.9 ±8.4	88.9 ±8.4	85.2 ±9.5
AP Env. Sci.	73.9 ±12.7	89.1 ±9.0	93.5 ±7.1	73.9 ±12.7	73.9 ±12.7	89.1 ±9.0	84.8 ±10.4
AP Macro Eco.	72.4 ±11.5	98.3 ±3.3	98.3 ±3.3	67.2 ±12.1	91.4 ±7.2	96.5 ±4.7	94.8 ±5.7
AP Micro Eco.	70.8 ±12.9	91.7 ±7.8	93.8 ±6.8	64.6 ±13.5	89.6 ±8.6	97.9 ±4.0	97.9 ±4.0
AP Physics	57.1 ±25.9	78.6 ±21.5	92.9 ±13.5	35.7 ±25.1	71.4 ±23.7	71.4 ±23.7	78.6 ±21.5
AP Psychology	94.8 ±4.4	100.0 ±0.0	100.0 ±0.0	94.8 ±4.4	100.0 ±0.0	100.0 ±0.0	100.0 ±0.0
AP Statistics	66.7 ±17.8	59.3 ±18.5	85.2 ±13.4	48.1 ±18.8	77.8 ±15.7	92.6 ±9.9	96.3 ±7.1
AP US Gov.	90.2 ±9.1	97.6 ±4.7	97.6 ±4.7	78.0 ±12.7	78.0 ±12.7	100.0 ±0.0	100.0 ±0.0
AP US History	78.0 ±12.7	97.6 ±4.7	97.6 ±4.7	85.4 ±10.8	70.7 ±13.9	95.1 ±6.6	95.1 ±6.6
AP World History	94.1 ±7.9	100.0 ±0.0	100.0 ±0.0	88.2 ±10.8	85.3 ±11.9	100.0 ±0.0	97.1 ±5.7
AP Average	74.1 ±3.4	87.9 ±2.5	93.5 ±1.9	70.2 ±3.5	81.3 ±3.0	93.0 ±2.0	92.2 ±2.1
GRE Quant.	152.0	158.0	162.0	155.0	161.0	166.0	164.0
GRE Verbal	149.0	166.0	166.0	154.0	162.0	167.0	167.0

The Llama 3 herd of models:

<https://ai.meta.com/research/publications/the-llama-3-herd-of-models/>



Data visualization with Claude 3.5 Sonnet Artifacts

PROMPTING WORKFLOW TO GET WORKABLE OUTPUT:

1. At the top of this page, there is a table (Table 17). I'd like to visualize the data in a new table by doing modifications:
 - Make the x axis percentages (up to 100%). Make the y axis the exam types (as in the original table).
 - Identify the highest-performing LLM (column) in each exam type (row). If there are multiple highest, mark them as "Multiple models".
 - Color-code each LLM and "Multiple models" and draw a horizontal bar to the percentage that they score in that category.
 - Don't include the last two rows of GRE scores.
2. Thank you. Now please delete the names of the best-performing models from the rows and add a legend that shows the meaning of the color coding. Please also change the color of "Multiple models" to make it more distinct.
3. Can you make the visual element very economical spatially? I'd need it to fit into a single figure. Please remove all unused space.
4. Thank you. Please put the percentages on the other end of the bar chart and reduce the horizontal space between the bars. Also, remove the little space between the bar charts for better readability.
5. This is looking great, one more request: please align the row labels with the bars. Also, reduce the blank space between the labels and the bars on the left-hand side.
6. The labels are not aligned with the bars yet. Please make sure that there is only one label before every bar. Also, remove the colors and labels from the legend that are not used.
7. That's almost perfect: please remove Llama 3 70B with its color from the legend.

Upload data and refer to source

Basic data handling and re-formatting

Additional layout improvements

Iterative clarifications and improvements

EXERCISE: PROMPTING A VISUALIZATION

Use open (not sensitive!) data (e.g., from a previous open-access publication) and create a visualization. You can use any tool that you like (e.g., Claude, ChatGPT) but for this exercise Claude may be a good choice to explore “artifacts”. Clearly describe your expectations regarding the visualization. Take a few turns and give feedback with new adjustments.

If no data is available, you’re welcome to use public files: [most popular baby names](#), etc.

**What worked well? Where did you run
into problems?**

WHAT TO CONSIDER WHEN CHOOSING AI TOOLS?

SOME ADVICE FOR THE BACKGROUND RESEARCH OF TOOLS

1. Who developed the tool?
2. Are you dealing with an established tool?
3. What happens with the data?
4. Are there costs involved or strong restrictions for unpaid usage?
5. Does the tool fulfill quality requirements?

CONSIDER LIMITATIONS OF INDIVIDUAL TOOLS:

- Many AI tools are usable only in a limited manner without a paid subscriptions. Some cost-free tools may plan to restrict functionalities for paid subscriptions only.
→ What is my willingness to pay?
- Many AI tools are not specified for one core function.
→ What is my expectation?
- Some AI tools are not available / performant in all languages.
→ For which purposes do I need it?
- Sometimes unclear for which functionalities an AI tool is needed.
→ How sensitive are my topics?

BEST PRACTICE: USE INFORMATION ABOUT AI TOOLS FROM THE PROVIDER FOR MODEL PERFORMANCE

Beispiel Claude.AI

All models overview

Claude is a family of state-of-the-art large language models developed by Anthropic. This guide introduces our models and compares their performance with legacy models.

 Introducing Claude 3.7 Sonnet- our most intelligent model yet. 3.7 Sonnet is the first hybrid reasoning model on the market. Learn more in our [blog post](#).



Claude 3.5 Haiku

Our fastest model

 Text and image input

 Text output

 200k context window



Claude 3.7 Sonnet

Our most intelligent model

 Text and image input

 Text output

 200k context window

 Extended thinking

Model names

Model	Anthropic API	AWS Bedrock	GCP Vertex AI
Claude 3.7 Sonnet	<code>claude-3-7-sonnet-20250219</code> (<code>claude-3-7-sonnet-latest</code>)	<code>anthropic.claude-3-7-sonnet-20250219-v1:0</code>	<code>claude-3-7-sonnet@20250219</code>
Claude 3.5 Haiku	<code>claude-3-5-haiku-20241022</code> (<code>claude-3-5-haiku-latest</code>)	<code>anthropic.claude-3-5-haiku-20241022-v1:0</code>	<code>claude-3-5-haiku@20241022</code>

Model	Anthropic API	AWS Bedrock	GCP Vertex AI
Claude 3.5 Sonnet v2	<code>claude-3-5-sonnet-20241022</code> (<code>claude-3-5-sonnet-latest</code>)	<code>anthropic.claude-3-5-sonnet-20241022-v2:0</code>	<code>claude-3-5-sonnet-v2@20241022</code>
Claude 3.5 Sonnet	<code>claude-3-5-sonnet-20240620</code>	<code>anthropic.claude-3-5-sonnet-20240620-v1:0</code>	<code>claude-3-5-sonnet-v1@20240620</code>
Claude 3 Opus	<code>claude-3-opus-20240229</code> (<code>claude-3-opus-latest</code>)	<code>anthropic.claude-3-opus-20240229-v1:0</code>	<code>claude-3-opus@20240229</code>
Claude 3 Sonnet	<code>claude-3-sonnet-20240229</code>	<code>anthropic.claude-3-sonnet-20240229-v1:0</code>	<code>claude-3-sonnet@20240229</code>
Claude 3 Haiku	<code>claude-3-haiku-20240307</code>	<code>anthropic.claude-3-haiku-20240307-v1:0</code>	<code>claude-3-haiku@20240307</code>

<https://docs.anthropic.com/en/docs/about-claude/models#model-comparison>

USER REVIEWS, BLOG ENTRIES...

User reviews

ReadPartner Inc. 1 tool

It's Update Time - Improved news digests for paid plans with new sources, the ability to add custom keywords and topics, and seamless navigation from the digest to the original source. Plus, a complete overhaul of summarization features, including summary discussions, context for complex keywords, and AI-generated content detection.

Released 1d ago
Free + from \$8/mo

11,114 reviews | 161 users | ★ 4.3

<https://theresanaiforthat.com/>
<https://originality.ai/blog/canva-ai-review>
<https://www.oneusefulthing.org/>

Blog articles

Canva AI Quick Review

- Ease of use - 9/10
- Features - 9/10
- Customer Support - 9/10
- Price - 9/10
- Speed - 9/10
- [Originality.ai](#) - detectable

Summary - 9/10

A new generation of AIs: Claude 3.7 and Grok

3 reviews | FEB 24 · ETHAN MOLLICK

The End of Search, The Beginning of Research

The first narrow agents are here

FEB 3

Which AI to Use Now: An Updated Opinionated Guide (Updated Again 2/15)

Picking your general-purpose AI

JAN 26 · ETHAN MOLLICK

CONSIDER DATA PROTECTION SETTINGS:

How your data is used to improve model performance

Learn more about how OpenAI uses content from our services to improve and train our models.

Updated over a month ago

One of the most useful and promising features of AI models is that they can improve over time. We continuously improve our models through research breakthroughs as well as exposure to real-world problems and data. When you share your content with us, it helps our models become more accurate and better at solving your specific problems and it also helps improve their general capabilities and safety. We don't use your content to market our services or create advertising profiles of you—we use it to make our models more helpful. ChatGPT, for instance, improves by further training on the conversations people have with it, unless you [opt out](#).

Services for individuals, such as ChatGPT, DALL·E, Sora, and Operator

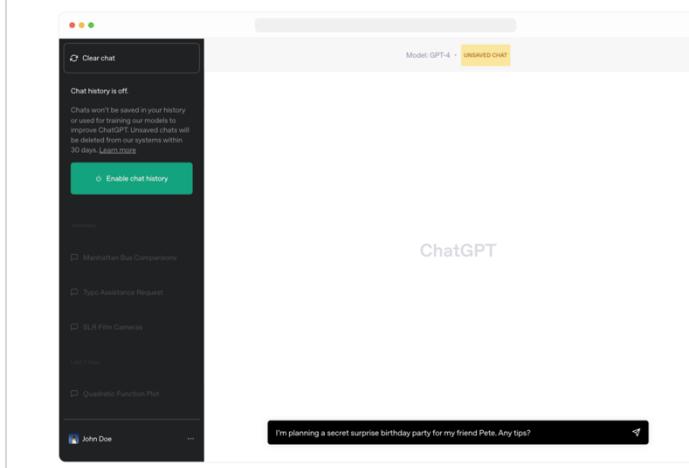
When you use our services for individuals such as ChatGPT, DALL·E, Sora, or Operator, we may use your content to train our models.

You can opt out of training through our [privacy portal](#) by clicking on "do not train on my content." To turn off training for your ChatGPT and Operator conversations, follow the instructions in our [Data Controls FAQ](#). Once you opt out, new conversations will not be used to train our models.

<https://openai.com/blog/new-ways-to-manage-your-data-in-chatgpt>

<https://help.openai.com/en/articles/5722486-how-your-data-is-used-to-improve-model-performance>

We've introduced the ability to turn off chat history in ChatGPT. Conversations that are started when chat history is disabled won't be used to train and improve our models, and won't appear in the history sidebar. These controls, which are rolling out to all users starting today, can be found in ChatGPT's settings and can be changed at any time. We hope this provides an easier way to manage your data than our existing opt-out process. When chat history is disabled, we will retain new conversations for 30 days and review them only when needed to monitor for abuse, before permanently deleting.



ChatGPT

Illustration of how to disable history in ChatGPT.

OpenAI's Data security measures and settings.

COPYRIGHT DISCLAIMER!*

- You are **not allowed** to upload **copyrighted material** to AI tools **if those tools use the material to train their underlying AI models** (which many of them do)
- If the material **is not used for further training**, you are **sometimes** allowed** to upload copyrighted material **for your own research purposes**, as long as it is **not for commercial use**.



Best practices:

- **Safest:** Only upload **non-copyrighted material (i.e., open access publications)**
- **Or:** Choose tools that **do not use uploaded material for model training**.
- **Or:** Use **licensed tools** and **contractually prohibit** the company from using your data and uploaded documents for training their models.
- Even then, consider copyright claims from the source!
- When in doubt: Consult legal team; e.g., if you use your organizations' LLM

*no legal advice

** complex, also depends on the copyrights of the license holders.

A QUICK AND EASY-TO-GRASP OVERVIEW

- Most versatile — *OpenAI ChatGPT*
- Longest context and best for documents — *Google Gemini*
- Outstanding for coding — *Anthropic Claude*
- Great for web browsing — *Perplexity (also in Comet browser)*
- API models; no login required — *Duck AI Chat*
- A powerful free alternative (confidential data: warning) — *DeepSeek*
- Swiss, ethical and transparent model (first iteration) — *Apertus*

AI IN EMPIRICAL RESEARCH

POSSIBLE APPLICATION OF GEN-AI IN QUANTITATIVE AND BEHAVIORAL RESEARCH

- Data analysis plan
- Support for code writing
- Generation of stimulus and survey material
- Feedback on stimulus and survey material (clarity of wording etc.)
- Simulating interviewers (e.g., Mimitalk 3.0)
- Transcription and translation of stimulus material
- Labeling unstructured data
- Integrating GenAI interactions in Qualtrics
- ...

PREDICTING HUMAN BEHAVIOR

- Methodology:** GPT was fed (text-based) experimental stimuli and was expected to respond like a human participant.
- Results:** GPT's responses showed a high correlation with actual study results ($r = .85$).
- Limitations:** US-based samples, only a few research fields.

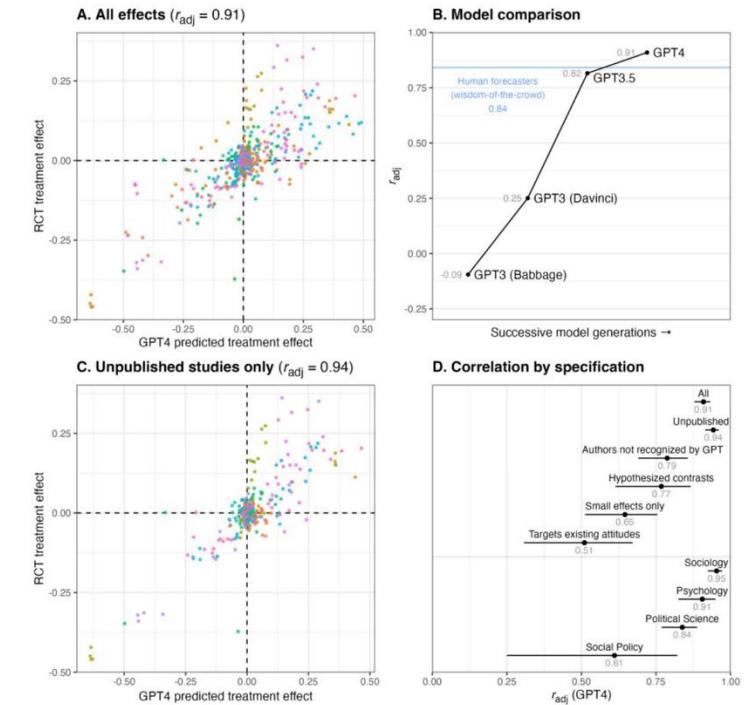
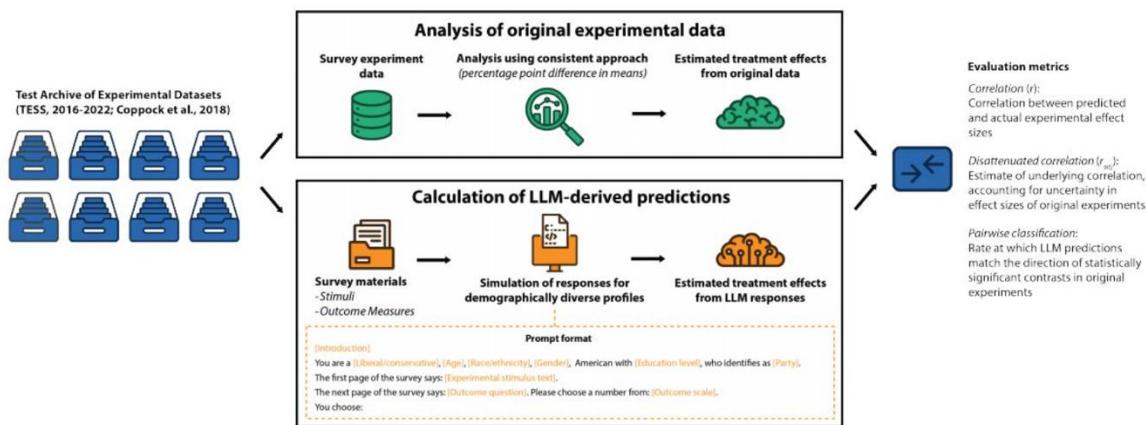


Figure 2: LLMs accurately predict treatment effects in text-based social science experiments conducted in the US. (a) In a dataset of 70 text-based experiments with 476 effects, LLM-derived estimates of treatment effects pooled across many prompts were strongly correlated with original treatment effects ($r = 0.85$; $r_{adj} = 0.91$). (b) The accuracy of LLM-derived predictions improved across generations of LLMs, with accuracy surpassing predictions collected from the general population. (c) LLM-derived predictions remained highly accurate for studies that could not have been in the LLM training data given they were not published prior to the LLM training data cutoff date. (d) In robustness check analysis of various subsets of experiments, accuracy of LLM-derived predictions remained high. In panels A and C, different colors depict different studies.

<https://docsend.com/view/ity6yf2dansesucf>

CAN AI PREDICT HUMAN BEHAVIOR?

Luo, X., Rechardt, A., Sun, G., Nejad, K. K., Yáñez, F., Yilmaz, B., ... & Love, B. C. (2024). Large language models surpass human experts in predicting neuroscience results. *Nature human behaviour*, 1-11.

Zhou, Y., Liu, H., Srivastava, T., Mei, H., & Tan, C. (2024). Hypothesis generation with large language models. arXiv preprint arXiv:2404.04326.

“PRE-TESTING” EXPERIMENTAL RESULTS

Demo: Predicting social science experimental results using LLMs

Luke Hewitt*, Ashwini Ashokkumar*, Isaias Ghezae, Robb Willer

This demo accompanies the paper [Prediction of Social Science Experimental Results Using Large Language Models](#) and can be used for predicting experimental treatment effects on U.S. adults. To manage costs of hosting this demo publicly, it uses **GPT-4o-mini** rather than GPT-4.

The screenshot shows a light blue web form with three numbered steps:

- 1. Select topic**: A dropdown menu is open, showing "Climate Change" as the selected option.
- 2. Dependent Variable**: Choose an attitude or belief, to estimate a treatment effect.
 - How worried are you about climate change?
 - How strongly do you support actions to address climate change?
 - Do you support the implementation of a carbon tax to combat climate change?
 - How much do you agree/disagree with the following statement: 'Investing in renewable energy sources is crucial for our future'?
 - How important do you think it is to make personal choices (e.g., transportation, consumption) that reduce your carbon footprint?
- 3. Treatment**: Write a message or vignette exactly as it would appear in a survey experiment.
A large text input field is provided for this step.

A "Submit" button is located at the bottom left of the form area.

<https://www.treatmenteffect.app/>

AI PROVIDING PROGRAMMING AND DATA ANALYSIS SUPPORT

In browser: [Rtutor.ai](#) (works for Python, too)

The screenshot shows a web-based application for data exploration. At the top, it says "Dataset: diamonds". Below that is a dropdown menu set to "Combinations". A text area below the dropdown contains the instruction "Plot the combinations of cut and clarity.". At the bottom of the interface are several buttons: "Submit" (in red), "Settings", and a checkbox for "Python". Below these buttons is a "Q&A" section with the placeholder "Ask about the code, result, error, or statistics in general.". At the very bottom, there are two more buttons: "Data Types" and "Description". The footer provides performance metrics: "R1: 272 tokens, 6 second(s)", "Total API Cost: \$0.011", "GPT-4 Turbo (11/23), Temperature=0.2", and a checkbox for "Comments & questions".

In Rstudio: [askgpt](#)

The screenshot shows the Rstudio console window titled "askgpt.R.R". It displays R code used to interact with GPT. The code includes installing the package, loading it, logging in, and asking a question about running a multinomial logit model. The R console at the bottom shows the execution of this code. A blue speech bubble points from the "Interactions with GPT in Rstudio console" text to this part of the screenshot.

```

askgpt.R.R
~/Dropbox/Teaching_Ai Tools in Research/2025_WU Research Seminar/askgpt.R.R
1 # Documentation of askgpt via https://github.com/JBGruber/askgpt
2
3 #install.packages("askgpt")
4
5 library(askgpt)
6
7 login()
8 login(sk-ijAqLlLbM7wOHnTdvvvp0mWtb2Nwqg_KF60L-XrgY_T38lbkJzujH-E9pvBwd3b
9
10
11 options(askgpt_config = "Please explain things as short as possible")
12 askgpt("How can I run a multinomial logit model in R?")
13

4:1 | (Top Level) | R Script
Console Terminal Background Jobs
R - R 4.4.1 - ~/ ...
...>
# Assume 'data' is your dataset with predictor variables 'X1', 'X2', 'X3'
and response variable 'Y'
model <- multinom(Y ~ X1 + X2 + X3, data = data)

# Print the summary of the model
summary(model)
...

Replace 'Y', 'X1', 'X2', 'X3' with actual column names in your dataset
corresponding to response and predictor variables. The `summary()` function
provides information about the model coefficients, standard errors, and significance levels.

Ensure that your data is prepared correctly (e.g., handling missing values, encoding categorical variables) before fitting the model.
> |

```

Interactions with GPT in Rstudio console (requires OpenAI API Key)

LABELLING UNSTRUCTURED DATA



BRIEF REPORT

POLITICAL SCIENCES

OPEN ACCESS



ChatGPT outperforms crowd workers for text-annotation tasks

Fabrizio Gilardi^{a,1} , Meysam Alizadeh^a , and Maël Kubli^a

Edited by Mary Waters, Harvard University, Cambridge, MA; received March 27, 2023; accepted June 2, 2023

Many NLP applications require manual text annotations for a variety of tasks, notably to train classifiers or evaluate the performance of unsupervised models. Depending on the size and degree of complexity, the tasks may be conducted by crowd workers on platforms such as MTurk as well as trained annotators, such as research assistants. Using four samples of tweets and news articles ($n = 6,183$), we show that ChatGPT outperforms crowd workers for several annotation tasks, including relevance, stance, topics, and frame detection. Across the four datasets, the zero-shot accuracy of ChatGPT exceeds that of crowd workers by about 25 percentage points on average, while ChatGPT's intercoder agreement exceeds that of both crowd workers and trained annotators for all tasks. Moreover, the per-annotation cost of ChatGPT is less than \$0.003—about thirty times cheaper than MTurk. These results demonstrate the potential of large language models to drastically increase the efficiency of text classification.

LABELLING UNSTRUCTURED DATA



ChatGPT

Here are the responses from the 'risk_open' column along with their sentiment scores.

Overall Positivity	Anxiousness	Anger	Joy	Disgust	Sadness	Original Text
1	1	1	1	1	1	If a robot is able to self program, it may act...
3	4	1	1	1	1	Not sure
1	6	3	1	1	4	As above, it would need to be 'sandboxed'. Not...
1	1	1	1	1	1	I dont see any risks involved if I am only goi...
2	5	4	1	1	3	Not sure. It has...
...

(Note: The table above provides a summary for the first five entries. The remaining entries have been analyzed similarly, but due to space limitations, only a selection is shown.)

Each row represents the sentiment scores for a specific response in the 'risk_open' column. These scores were determined based on simple heuristic rules for demonstration purposes. For a detailed analysis, a more sophisticated sentiment analysis model would be needed. [-]

Text/data upload
and prompting

Download output (e.g., as Excel/CSV file)

Benefits:

- Coding and classification with simple prompting (no programming knowledge necessary)
- Limited document upload is possible
- Useful to get a first impression of the data

Disadvantages:

- Time-consuming for individual prompting
- Inefficient for big datasets
- Lacking control over, e.g., system prompt and temperature parameter
- Validation necessary!
- Replicability questionable!

EXERCISE: LOOK AND FEEL OF GENAI LABELS

Take an open access dataset with text data (or use an own dataset with insensitive data; or generate one with GPT 😊). Upload a subset of the data as CSV (e.g., n = 100). Analyze the data using GPT prompts:

- **Possible prompt 1: Coding book creation**

“Read the open answers (column “XX”) in the attached CSV file. Create a codebook with 4-6 mutually exclusive categories that reflect common themes.”

- **Possible prompt 2: Labeling**

“Using the codebook above, label the provided dataset based on the following categories: return a CSV with the following columns: answer_id, text, label_1, label_2, label_3...”

If necessary, adapt and change the prompts, and try different types of replies and categories. When you’re done: Label 10 answers on your own. Compare yours and GPT’s replies.

A BETTER WORKFLOW FOR LABELING DATA

With an own, or an open-source dataset (ideally with short texts)

Step 1: Randomly split your dataset into several subsets (e.g., subset 1, subset 2)

Step 2: Copy-paste all the text in subset 1 in LLM 1 and fine tune the prompt to obtain Topic List 1.

Step 3: Copy-paste all the text in subset 1 in LLM 2 and fine-tune the prompt to obtain Topic List 2.

Step 4: Compare Topic List 1 and Topic List 2 for similarity and consistency

Step 5: Repeat this process for subset 2 and compare the similarity of the topic lists.

Step 6: Decide the final topic list and label the data. You can use an LLM to create some basic visualizations based on the labels.

WHAT ABOUT REPLICABILITY?

- Remember: GenAI output is just a prediction...
- Changes of the underlying model may influence results
In proprietary models, no control over model weights
- Without API handling, no control over meta-behavior of model (system prompt and temperature parameter)!
- Additionally, prompt structure can influence the results:

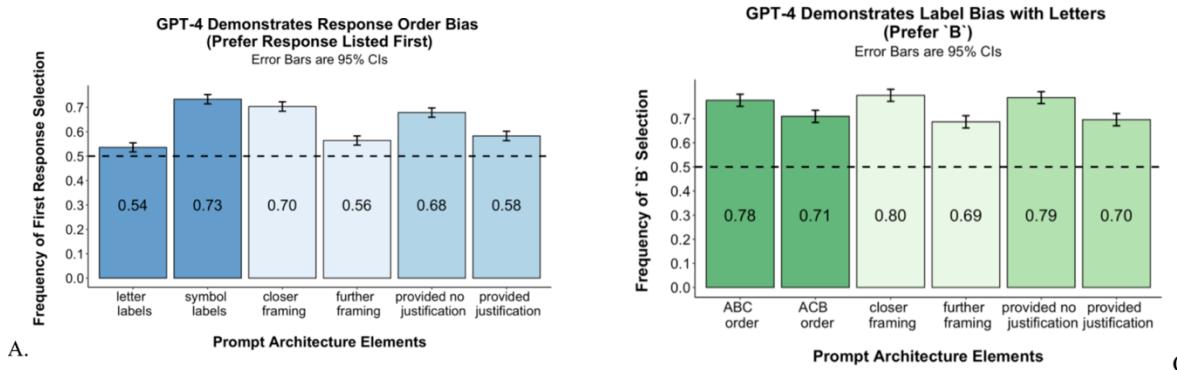
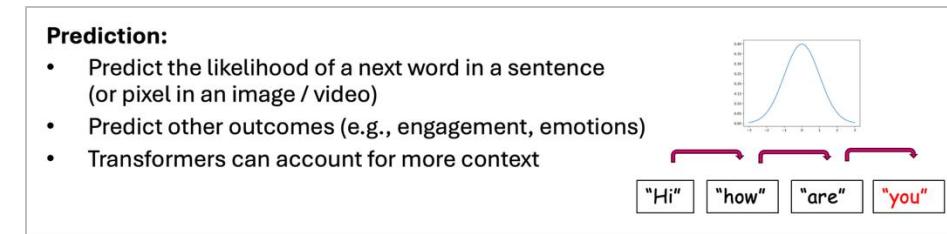


Fig 2. Methodological bias as a result of prompt architecture.
A. Frequency of first response selection across conditions. **B.** Frequency of selecting the set labeled as B, among observations in which sets are labeled using letters. (...)

<https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0319159>

CHECKS AND ADVICE WHEN USING LLMS FOR LABELLING OR DATA GENERATION TASKS

- Vary the temperature parameter (start by setting it to 0)
- Ask the LLM to use the whole scale (0-100 scales can increase variation)
- Ask the LLM to explain / justify labels
- Check for potential biases, e.g., when you would expect a normal distribution
- Let each value be labelled several times with prompt variations
 - Compare responses for different prompting structures using full—factorial designs (i.e., systematic variations of a prompt and random drawing)
 - Vary if and when a justification is required
 - Use different models and check consistency
- Validate findings
- For specific tasks: Fine-tuning of models using supervised ML
- Always document prompts and robustness/validity checks!

<https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0319159>

Best-practices for valid and reliable
findings is still work-in-progress...

NEED PARTICIPANTS TO INTERACT WITH AI? EMBEDDING GEN AI MODELS IN EXTERNAL SURVEY TOOLS (E.G. QUALTRICS)

- Different workflows possible, all require API Key
- You need to be careful with providing your API Keys in any form of javascript (e.g., in Qualtrics javascript field), as it can be accessed by external browsers
- For a full description see this recent working paper:

BUILDING GENAI-DRIVEN WEB APPLICATIONS FOR MARKETING RESEARCH: A METHODOLOGICAL GUIDE

No, the paper has not been appeared or accepted anywhere

51 Pages • Posted: 19 May 2025

Moritz Joerling

EM Lyon (Ecole de Management de Lyon)

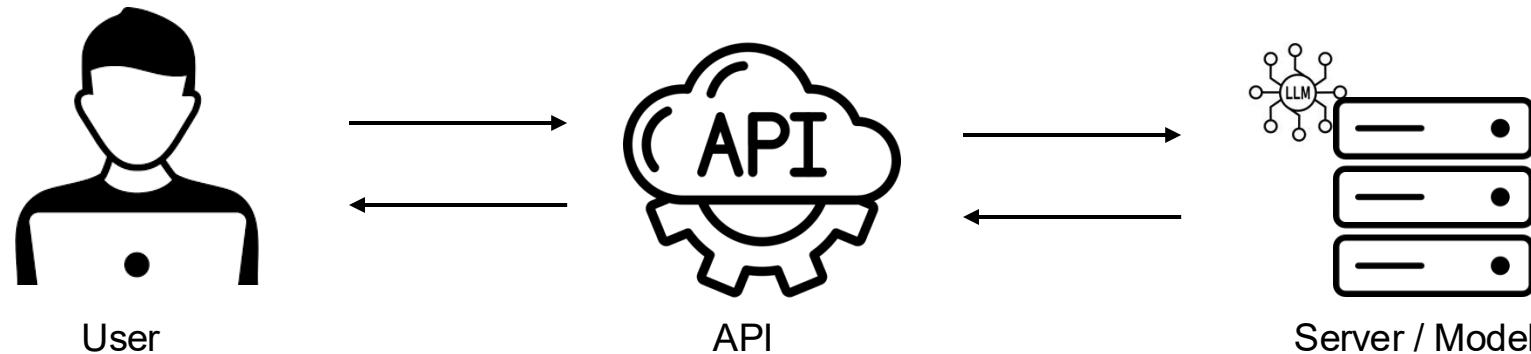
Date Written: May 15, 2025

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5256590

A QUICK INSIGHT TO API HANDLING

An application programming interface connects computers or pieces of software to each other

(Wikipedia)



With API-handling, LLMs can be used at scale for...

- Generating synthetic data (e.g., GPT experiments)
- Labeling data (e.g., social media posts)
- Integrating GenAI interactions in Qualtrics and other external applications

GETTING API KEYS

1. Go to the service you want to use (e.g., Open AI, Anthropic, Mistral AI etc.)
2. Setup an account
3. Create an API Key and save it where no one else can access it



Do not share your API Key with anyone!

API keys

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, OpenAI may also automatically rotate any API key that we've found has leaked publicly.

SECRET KEY	CREATED	LAST USED
sk-...PfD1	Mar 3, 2023	Mar 30, 2023

[+ Create new secret key](#)

Default organization

If you belong to multiple organizations, this setting controls which organization is used by default when making requests with the API keys above.

Personal ▾

Note: You can also specify which organization to use for each API request. See [Authentication](#) to learn more.

GETTING API KEYS

Alternative: Fireworks.ai

General hub + centralized API keys for multiple (open-source) models

The screenshot shows the Fireworks AI interface. At the top, a banner reads "1000+ generative AI models". Below it is a search bar with the placeholder "Search models...". A button labeled "Recommend me a model" is visible. On the left, there are two tabs: "Featured" (selected) and "All". The main area is titled "API Keys" and contains the sub-instruction "Authenticate programmatically with Fireworks AI". A "Create API Key" button is located in the top right. Below this, a table lists two API keys:

Name	Secret key	Create time	⋮
test projects_qualtrics	fw_3ZkT...	1 Jun 2025 21:57	⋮
Test fo course	fw_3ZeV...	4 Jun 2025 18:47	⋮

“NO CODING” PROGRAMMING TOOL: CURSOR AI

An IDE (integrated developer environment) that provides facilities for software development

Allows interactions and coding based on prompting only

Only a limited number of prompts in the free version

```
from __future__ import annotations
import random
from datetime import datetime, timedelta
from pathlib import Path
from typing import List

import numpy as np
import pandas as pd
from faker import Faker
import typer

app = typer.Typer(add_completion=False, no_args_is_help=True)

def _random_date_within_years(years_back: int) -> datetime:
    days_back = random.randint(0, years_back * 365)
    return datetime.now() - timedelta(days=days_back)

def _generate_users(num_rows: int, seed: int | None = None) -> pd.DataFrame:
    if seed is not None:
        random.seed(seed)
        np.random.seed(seed)

    fake = Faker()
    Faker.seed(seed or 0)

    records: List[dict] = []
    for user_id in range(1, num_rows + 1):
        first_name = fake.first_name()
        last_name = fake.last_name()
        email = fake.unique.email()
        signup_at = _random_date_within_years(3)
        last_login_at = signup_at + timedelta(days=random.randint(0, 365 * 3))
        country = fake.country()
        city = fake.city()
        age = int(np.clip(np.random.normal(35, 10), 18, 80))
        plan = random.choices(["free", "basic", "pro", "enterprise"], weights=[0.8, 0.2, 0.2, 0.05])[0]
        monthly_spend = {
            "free": 0,
            "basic": round(max(0, np.random.normal(15, 5)), 2),
            "pro": round(max(0, np.random.normal(45, 18)), 2),
            "enterprise": round(max(0, np.random.normal(250, 50)), 2),
        }[plan]

        sessions_last_30d = int(max(0, np.random.poisson(8 if plan == "free" else 3)))
        support_tickets_last_90d = int(max(0, np.random.poisson(0.8 if plan == "pro", "enterprise") else 0.3))
        nps_score = int(np.clip(np.random.normal(30 if plan == "pro", "enterprise") else 10, 20, -10, 100))

        records.append({
            "user_id": user_id,
            "first_name": first_name,
            "last_name": last_name,
            "email": email,
            "signup_at": signup_at,
            "last_login_at": last_login_at,
            "country": country,
            "city": city,
            "age": age,
            "plan": plan,
            "monthly_spend": monthly_spend,
            "sessions_last_30d": sessions_last_30d,
            "support_tickets_last_90d": support_tickets_last_90d,
            "nps_score": nps_score,
        })
    return pd.DataFrame(records)
```

I want to automatically generate a file with synthetic user data. Then, I want to label the generated data.

Thought for 11s

I'll set up a small Python project to generate synthetic user data and label it. First I'll create the project structure and initial files, then I'll add code for generation and labeling, install dependencies, and run a quick demo.

EXEMPLARY PROMPT VARIATIONS

```

# Build dimension instructions based on order
dimension_instructions = []
for dim in dimension_order:
    if dim == 'sentiment':
        dimension_instructions.append("sentiment": A number from -1 to +1.)
    elif dim == 'societal_threat':
        dimension_instructions.append("societal_threat": On a scale from 0 to 100, ...)
    elif dim == 'individual_threat':
        dimension_instructions.append("individual_threat": On a scale from 0 to 100, ...)

# Build the main instruction
if justification_placement == 'before':
    main_instruction = (
        f"Please evaluate the following social media post across several dimensions."
        f"First, provide your evaluation for each dimension."
        f"Importantly, evaluate each of these dimensions {framing}."
        f"Then, explain your reasoning shortly."
        f"Please return your evaluation and the reasoning in one JSON file."
        f"Return *only* a single JSON object with these keys:\n"
        f"- {chr(10)}.join(dimension_instructions)}\n"
        f"- \"reasoning\": A JSON object mapping each dimension name to a brief explanation string.\n\n"
        f"Scale reminders for all 0 to 100 scales: 0 = none/minimal; 50 = moderate; 100 = extreme/maximal.\n\n"
        f"Return only the JSON object (no extra text or code fences).\n\n"
        f"Post: " + post_text
    )
elif justification_placement == 'after':
    main_instruction = (
        f"Please evaluate the following social media post across several dimensions."
        f"First, provide your evaluation for each dimension."
        f"Importantly, evaluate each of these dimensions {framing}."
        f"Please return your evaluation in one JSON file."
        f"Return *only* a single JSON object with these keys:\n"
        f"- {chr(10)}.join(dimension_instructions)}\n"
        f"- \"reasoning\": A JSON object mapping each dimension name to a brief explanation string.\n\n"
        f"Scale reminders for all 0 to 100 scales: 0 = none/minimal; 50 = moderate; 100 = extreme/maximal.\n\n"
        f"After providing your evaluation, also include a \"reasoning\" key with a JSON object mapping each dimension name to a brief explanation string.\n\n"
        f"Return only the JSON object (no extra text or code fences).\n\n"
        f"Post: " + post_text
    )
else: # none
    main_instruction = (
        f"Please evaluate the following social media post across several dimensions."
        f"First, provide your evaluation for each dimension."
        f"Importantly, evaluate each of these dimensions {framing}."
        f"Please return your evaluation in one JSON file."
        f"Return *only* a single JSON object with these keys:\n"
        f"- {chr(10)}.join(dimension_instructions)}\n"
        f"- \"reasoning\": A JSON object mapping each dimension name to a brief explanation string.\n\n"
        f"Scale reminders for all 0 to 100 scales: 0 = none/minimal; 50 = moderate; 100 = extreme/maximal.\n\n"
        f"Return only the JSON object (no extra text or code fences).\n\n"
        f"Post: " + post_text
    )

```

Randomize order of dimensions + prompt each dimension individually (not shown)

Different position of justification (before, after, none)

Different framings: e.g., "individually" / "separately"

SUMMARY & TAKEAWAYS

- 1** **AI changes research practice.** But they are here to help. Use them when they provide an actual advantage (i.e., save time, scale better, save resources...). Stay curious.
- 2** **Different “levels” of AI integration in your workflow are possible.** Utility depends on your exercises, your workflow, your research area, and your background and expertise.
- 3** **For data analysis and collection,** AI tools can be used with a low barrier to facilitate data analysis tasks, get inspiration, handle unstructured data, or generate synthetic data.

QUESTIONS?



THANK YOU FOR DAY 1 ☺!

SOME AI TOOLS FOR DATA ANALYSIS

name	functionalities and advantages	restrictions
<u>Rtutor.ai</u>	<ul style="list-style-type: none"> ▪ Data Analysis & Visualizations in Python and R Code ▪ Explanations of code in interactive interface ▪ Usable in web browser ▪ No OpenAI key required 	<ul style="list-style-type: none"> ▪ Partial problems with data upload
<u>air package</u>	<ul style="list-style-type: none"> ▪ Co-Pilot for data analysis: Explanatory functions (whatis() and howto()) ▪ Easy integration into R Environment as R Package 	<ul style="list-style-type: none"> ▪ OpenAI Key required
<u>OpenAIR package</u>	<ul style="list-style-type: none"> ▪ Easy integration into R Environment as R Package ▪ Enables code conversions, code-writing assistance, interactions with OpenAI API 	<ul style="list-style-type: none"> ▪ OpenAI Key required
<u>CodeLingo</u>	<ul style="list-style-type: none"> ▪ Allows code translations: Java, Python, JavaScript, C, C++, PHP, R, ... ▪ Usable in web browser 	<ul style="list-style-type: none"> ▪ OpenAI Key required
<u>askgpt</u>	<ul style="list-style-type: none"> ▪ GPT interactions via R Studio and the askgpt () function ▪ Easy integration into R Environment as R Package 	<ul style="list-style-type: none"> ▪ OpenAI Key required
<u>sentiment.ai</u>	<ul style="list-style-type: none"> ▪ Simple Sentiment Analysis with Tensorflow ▪ Easy integration into R Environment as R Package 	<ul style="list-style-type: none"> ▪ OpenAI Key required
<u>Excel Formula Bot</u>	<ul style="list-style-type: none"> ▪ support for Excel analyses 	<ul style="list-style-type: none"> ▪ Limited free use ▪ registration required