Perfecting the Prompt

Intro to Prompt Engineering

Simon Stone

Research Software Engineer for HPC & AI

May 27, 2025

Introducing Research Software Engineering

Collaborative expertise in software engineering, designed to bridge the gap between innovative ideas and impactful outcomes. Our services include:

- 🤝 **Grant Proposal Consulting** to ensure accurate resource estimations and project feasibility.
- 🚀 **Rapid Prototyping** to refine concepts and explore solutions.
- ⛑️ **Ongoing Application Support** and **Application Rehabilitation** for existing applications.
- 🌎 **Open-Source Releases** to share knowledge and contribute to the wider community.

Contact us today to discuss your project and discover how Research Software Engineering can be your trusted partner in innovation.

What you will learn in this session

💬 How Large Language Models construct language

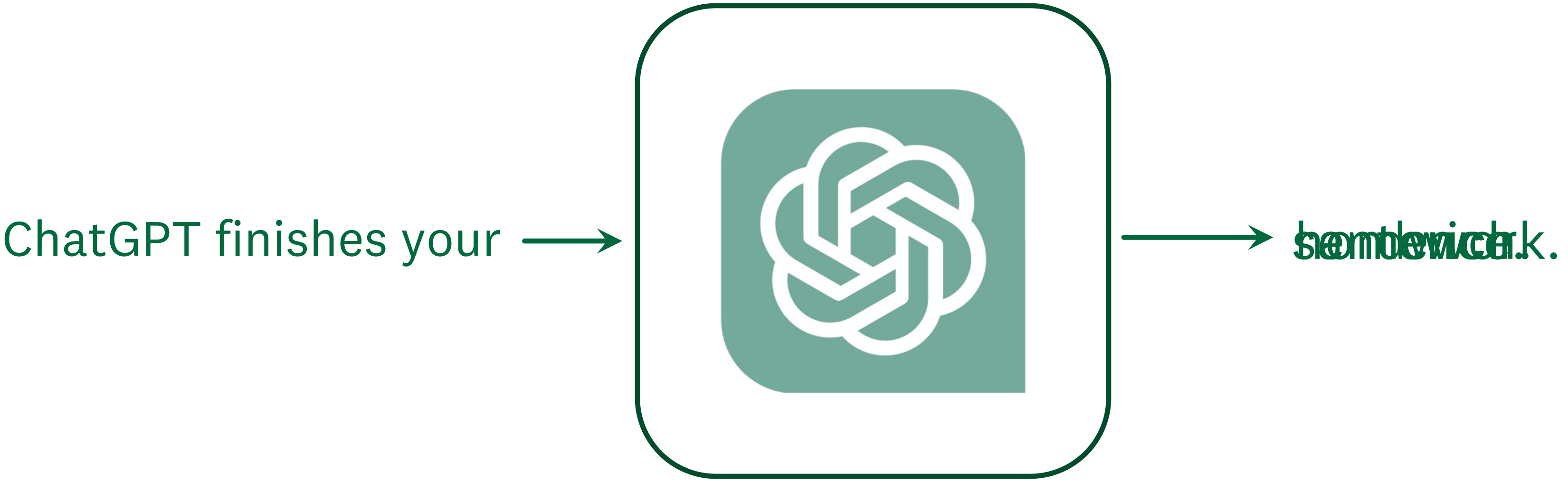
🎛️ The three main ways to customize an LLM’s output

🫀 The anatomy of a prompt

😎 Techniques to improve your prompts and get better results

Background

How Large Language Models construct language



ChatGPT finishes your

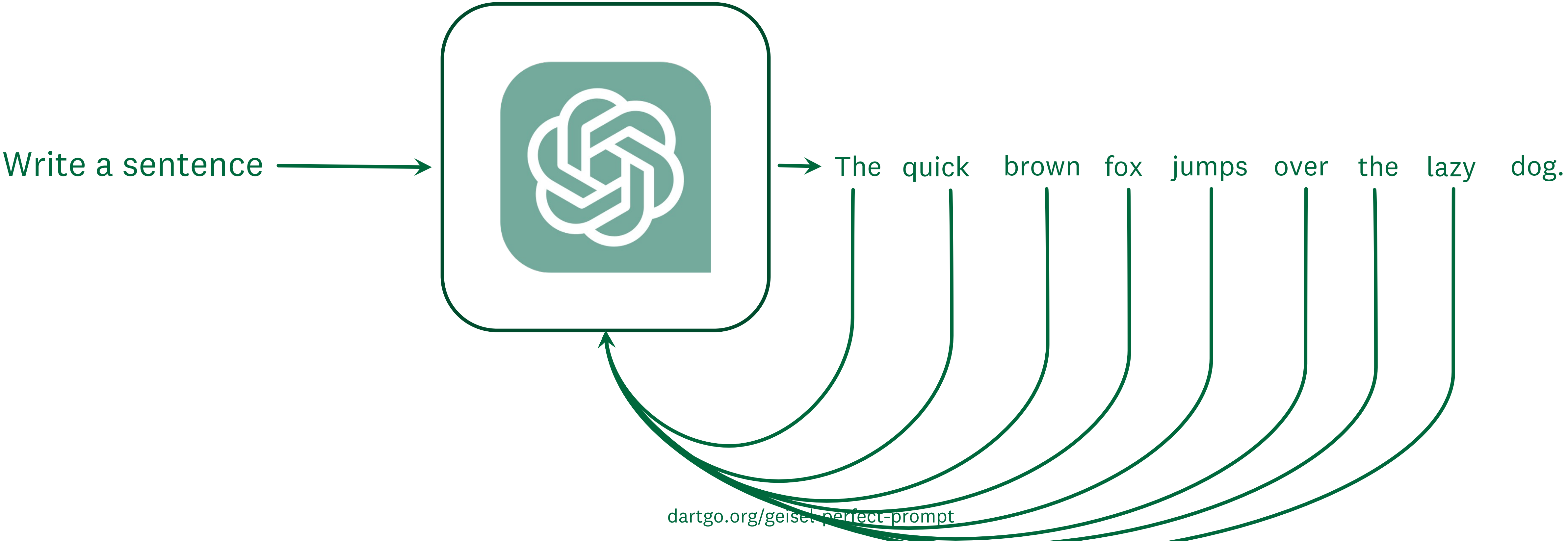
homework.

sandwich.

sentence.

Background

How Large Language Models construct language



Background

How Large Language Models construct language

🗣 How many words are in your response?

There are 12 words in my response.

🗣 List them out in a numbered list.

Sure, here are the words in my previous response listed in a numbered list:

1. There
2. are
3. 12
4. words
5. in
6. my
7. response.

Background

How Large Language Models construct language

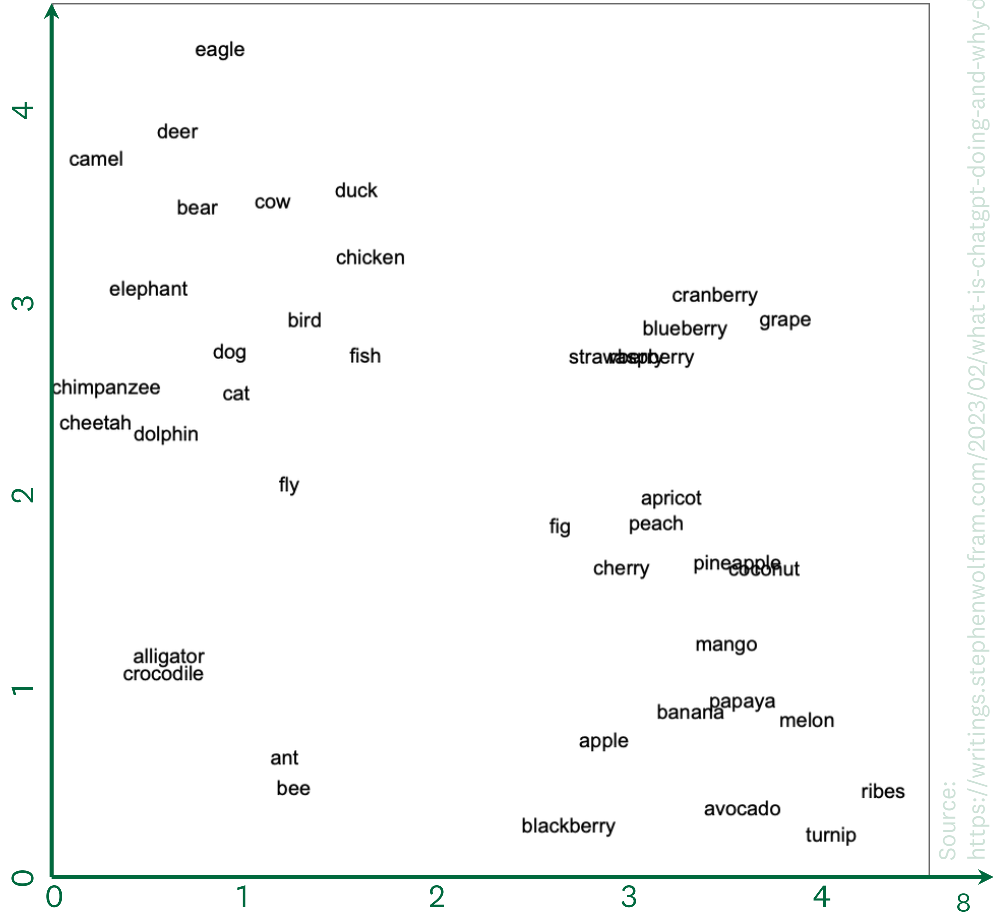
- The patterns in written language are learned from text data in a training phase
- GPT 3.5, for example, has been trained on about 570 GB of texts or 300 billion words (English Wikipedia: 4.2 billion words)
- This would still not be enough data if we were looking for examples of longer specific word sequences!

Background

How Large Language Models construct language

Two important “tricks” solve this problem:

- The model learns to map every word into a numerical “meaning space”, where words with similar meaning are represented by similar numbers
- That way the model can base the probability of the next word on sequences of similar words
- GPT 3.5 uses 2048 dimensional embeddings (probably)



Source: https://writings.stephenwolfram.com/2023/02/what-is-chatgpt-doing-and-why-does-it-work/

How Large Language Models construct language
Implications for prompt design

🤷‍♂️ There is no notion of truth or fact built into an LLM

How Large Language Models construct language
Impli

🤷 The

“All responses are hallucinations, but some hallucinations are helpful.”

How Large Language Models construct language
Implications for prompt design

- 🤷‍♂️ There is no notion of truth or fact built into an LLM
- 🔙 LLMs can only reference what came before the current token (no planning ahead)
- ⛓️ LLMs rely on structure that is consistent with its training material
  - Demonstration: Who is Timothée Chalamet’s mother?
- 🫛 LLMs have a notion of similarity of words

Ways to customize an LLM’s output

💬 Prompting

- You guide the model through instructions, context, and a few examples

📚 Retrieval-Augmented Generation

- You have a specific knowledge source (e.g., a collection of documents) you want the model to reference when responding to your prompt

🎛️ Fine-tuning the model

- You repeatedly intend to use the model in a particular domain and/or want to specialize it to a specific task

Ways to customize an LLM’s output

💬 Prompting

- You guide the model through instructions, context, and a few examples

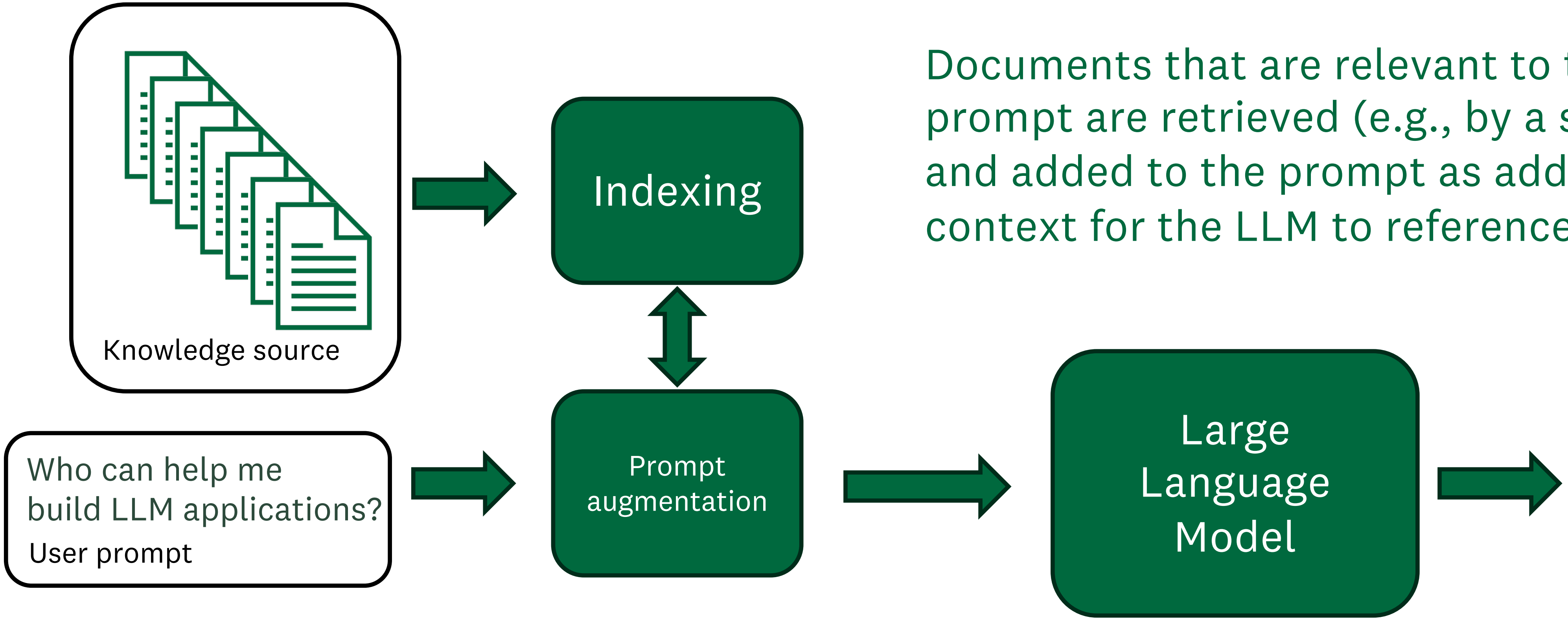
📚 Retrieval-Augmented Generation

- You have a specific knowledge source (e.g., a collection of documents) you want the model to reference when responding to your prompt

🎛️ Fine-tuning the model

- You repeatedly intend to use the model in a particular domain and/or want to specialize it to a specific task

Ways to customize an LLM's output
Retrieval-Augmented Generation (RAG)



You can contact Research Software Engineering.

The email address is research.computing@ dartmouth.edu.

Documents that are relevant to the user's prompt are retrieved (e.g., by a search engine) and added to the prompt as additional context for the LLM to reference.

Ways to customize an LLM’s output
Retrieval-Augmented Generation (RAG)

Demo: elicit.com

Ways to customize an LLM’s output

Fine-tuning a model

- 🎛️ If you cannot achieve the desired model behavior by prompting, you may be able to fine-tune a model
- 🏋️ Fine-tuning means continuing to train the model by changing its parameters
- ⁉️ Fine-tuning requires at least 50 to 100 examples of prompts and the corresponding desired responses

Prompt engineering

Anatomy of a prompt

A good prompt contains some or all of the following components:

☝️ Instruction

- “Summarize the following article.”

👉 Context

- “The summary will appear on social media and should be engaging and energetic.”

📃 Input Data

- the full text of the article

🖨️ Output indicator

- “Format your response using markdown and emojis.”

Prompt engineering

Anatomy of a prompt

Demo: The History of Sourdough Bread

Prompt engineering
Context stuffing and Grounding

📄 Adding additional text to use as a reference to your prompt is called context stuffing

👉 Using context information to control what the LLM’s responses are based on is called grounding

Prompt engineering
Context stuffing and Grounding

Demo: RAG at chat.dartmouth.edu

Try it yourself!

Choose an article, blog entry, or any other website you want to ground the conversation in!

Examples:

- thelancet.com
- policies.dartmouth.edu
- grants.nih.gov/funding/activity-codes

Prompt engineering
Zero-shot prompting

🤔 Zero-shot prompting means asking a question without providing any examples

  - The model gets “zero shots” to learn the task

- Demo: “Write a lesson plan for a workshop on Citation Management”

Try it yourself!

Some examples (courtesy of GPT-4o Mini):

"Generate a concise summary of the impact of climate change on New England's ecosystems."

"List five innovative research questions related to artificial intelligence in education."

"Provide three strategies for effectively networking in a professional academic environment."

"Outline a framework for organizing a successful panel discussion on mental health in college students."

Prompt engineering
Few-shot prompting

⁉ Few-shot prompting means asking a question and providing one or more examples of desired responses

- The model gets a “few shots” to identify the pattern to reproduce

- Demo: ”Write a lesson plan for a workshop on Agentic AI.”

Try it yourself!

Example (courtesy of GPT-4o):

“Dartmouth's Fall Hackathon: Spend a weekend collaborating with fellow students, solving real-world problems using technology. Enjoy workshops, networking, and fun activities!

International Cultural Night: Celebrate the diversity of our campus with an evening of performances, food stalls, and cultural exhibits from around the world.

Describe the annual Dartmouth Winter Carnival.”

Prompt engineering
Personas

- 👤 You can describe a particular perspective or set of characteristics to a Large Language Model
- 🎭 The LLM will then “roleplay” this persona in its responses
- 👍 Through personas, you can avoid writing very explicit instructions relating to the style and manner of the responses
- Demo: ”You are a first-year student at Dartmouth”

Try it yourself!

Some examples (courtesy of GPT-4o):

- *“As Eleazar Wheelock, introduce yourself and explain your vision for establishing Dartmouth College.”*
- *“As Sarah, a sophomore majoring in biology, write an email to incoming first-year students about the best resources on campus to help with their studies.”*
- *“As Vince Lombardi, inspire the athletes at Dartmouth College with a speech about dedication and teamwork.”*

Prompt engineering

Chain-of-thought prompting or Reasoning

- LLMs depend on predictable, left-to-right sequences

  “Jumping” to a conclusion often causes errors

- Prompting the model to respond with a step-by-step explanation breaks a problem down into multiple shorter, more predictable sequences leading to fewer errors

- Demo: “The cafeteria had 23 apples. If they used 15 to make lunch, bought 6 more, but half of those were rotten - how many good apples do they have?”

Try it yourself!

Some examples (courtesy of GPT-4o):

- *“Consider the goal of enhancing Dartmouth College's green campus initiatives. Outline the steps and reasoning involved in developing a comprehensive plan.”*

- *“Outline the steps and considerations for developing an interdisciplinary program in 'Sustainability and Digital Innovation' at Dartmouth College.”*

Prompt engineering
Getting feedback from the LLM

🤝 Asking the LLM itself to produce or improve a prompt can be very beneficial

🔎 It helps to tease out elements that need to be more explicit or that may be ambiguous

Demo: Image generation

Perfecting the prompt
Summary

- LLMs construct responses from left to right with no planning ahead
- Good prompts give clear instructions, context, and examples
- Personas can help to avoid writing too many explicit instructions
- LLMs are more accurate when they process complex tasks in smaller steps (chain-of-thought prompting)

Next steps
I want to know more!

Fantastic top-to-bottom explanation of ChatGPT:

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

Prompting guides:

- https://platform.openai.com/docs/guides/prompt-engineering/strategy-write-clear-instructions
- https://www.promptingguide.ai/

Hub for specialized prompts:

- https://smith.langchain.com/hub

If you have questions, ideas, or just want to play around with Generative AI, come talk to us!

- Reach out: research.computing@dartmouth.edu
- Visit our workshops: www.dartgo.org/rradworkshops

Thank you!