07.11.23

OPENCAMPUS.sh

Practical Engineering with LLMs

PROMPT ENGINEERING

QUIZ ON PROMPT ENGINEERING

https://forms.office.com/r/qEaVF35nDK

TODAY'S SCHEDULE

- Quiz
- Homework presentation
- Short recap on intro to LLMs & prompt engineering
- Breakout session on prompt hubs
- Anatomy of an app
- Breakout session on project ideas
- Presentation of project ideas and default projects
- Homework for next week

HOMEWORK PRESENTATION

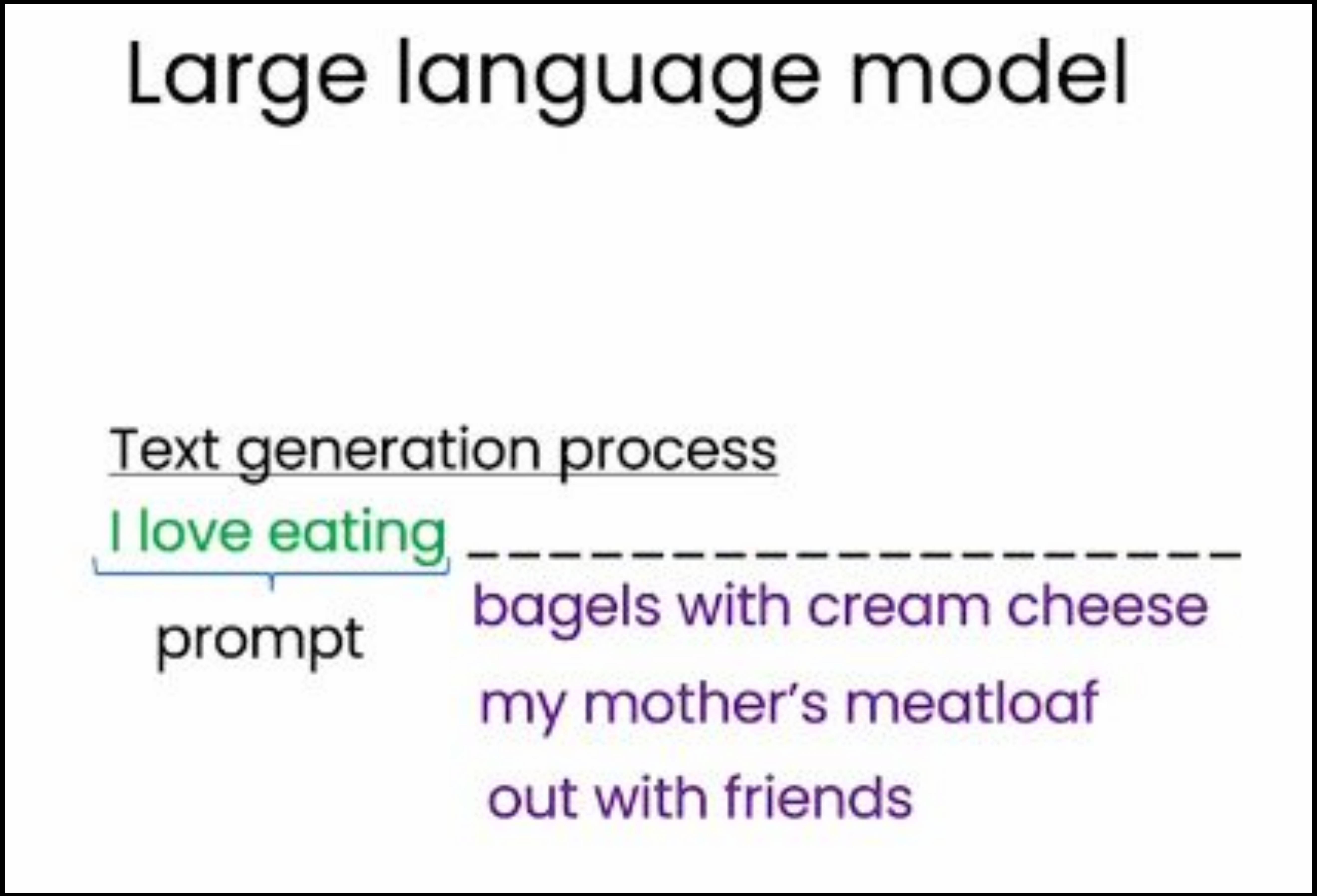
1. Watch the provided videos and review the text to identify various tactics and techniques used in Prompt Engineering. List these tactics and techniques with clear examples.

2. Handling Ambiguity: The word “Java” has at least three meanings (programming language, island in Indonesia, or coffee).
Task: Try out the following prompt: Can you tell me about Java?
Task: Suppose you are developing an app and do not know beforehand what your app's user means. Can you improve the prompt to help the model to clarify what the user represents and generate an answer?

3. Generate three potential course project ideas that you could work on. Be prepared to discuss these ideas in the next session. Feel free to consider alternative projects if needed.

4. Select one of the potential project ideas from Task 3 and create prompts for the language model to solve the problem. Use different prompting tactics and techniques to see which one produces the best results in addressing the chosen project idea.

QUICK RECAP- what are LLMs doing?



Large language model

Text generation process

I love eating ____________________

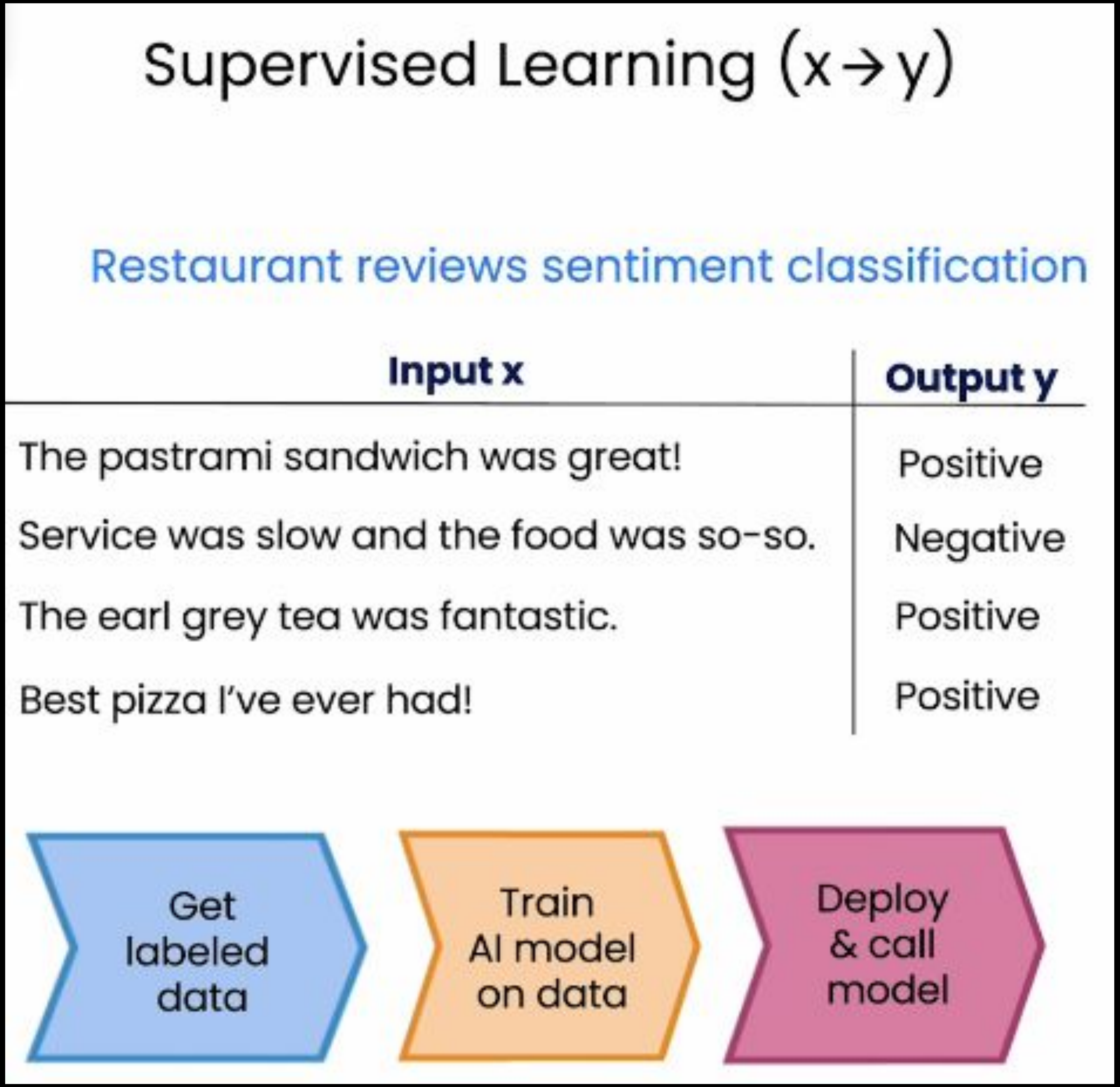
prompt

bagels with cream cheese

my mother's meatloaf

out with friends

QUICK RECAP- how are they trained?



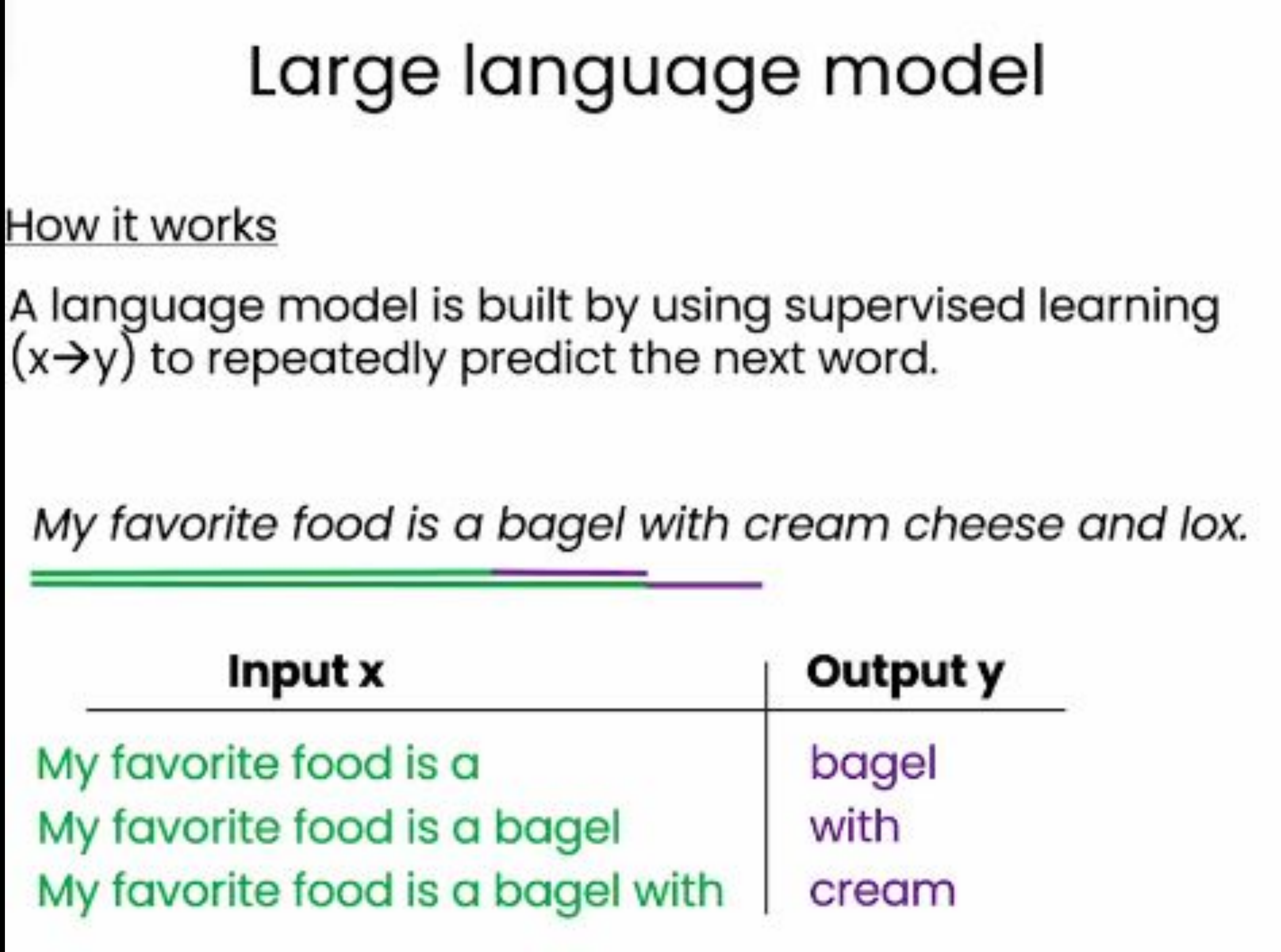
Supervised Learning (x→y)

Restaurant reviews sentiment classification

Input x	Output y
The pastrami sandwich was great!	Positive
Service was slow and the food was so-so.	Negative
The earl grey tea was fantastic.	Positive
Best pizza I've ever had!	Positive

Get labeled data
Train AI model on data
Deploy & call model

QUICK RECAP- how are they trained?



Large language model

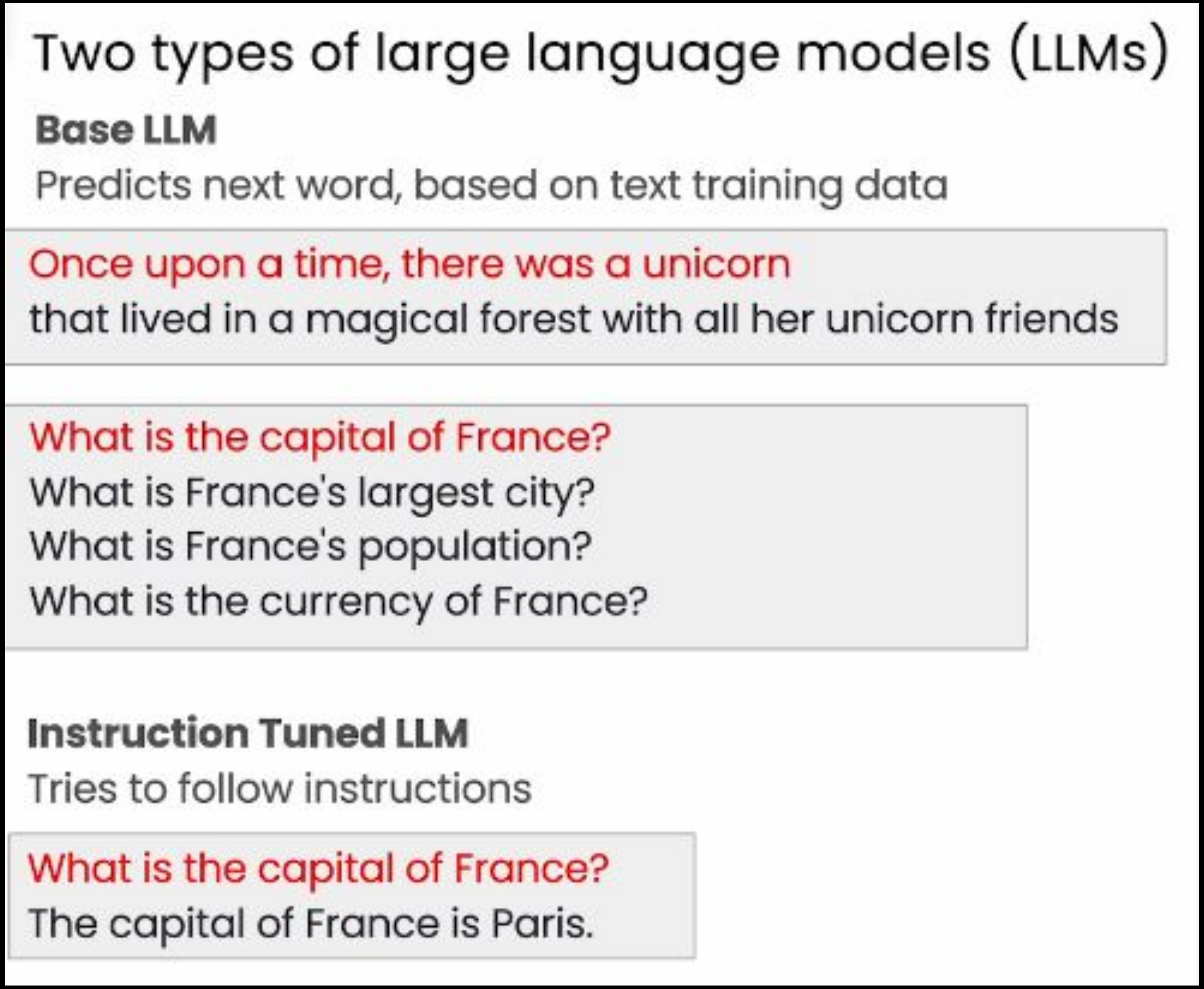
How it works

A language model is built by using supervised learning (x→y) to repeatedly predict the next word.

My favorite food is a bagel with cream cheese and lox.

Input x	Output y
My favorite food is a	bagel
My favorite food is a bagel	with
My favorite food is a bagel with	cream

QUICK RECAP- how are they trained?



Two types of large language models (LLMs)

Base LLM

Predicts next word, based on text training data

Once upon a time, there was a unicorn
that lived in a magical forest with all her unicorn friends

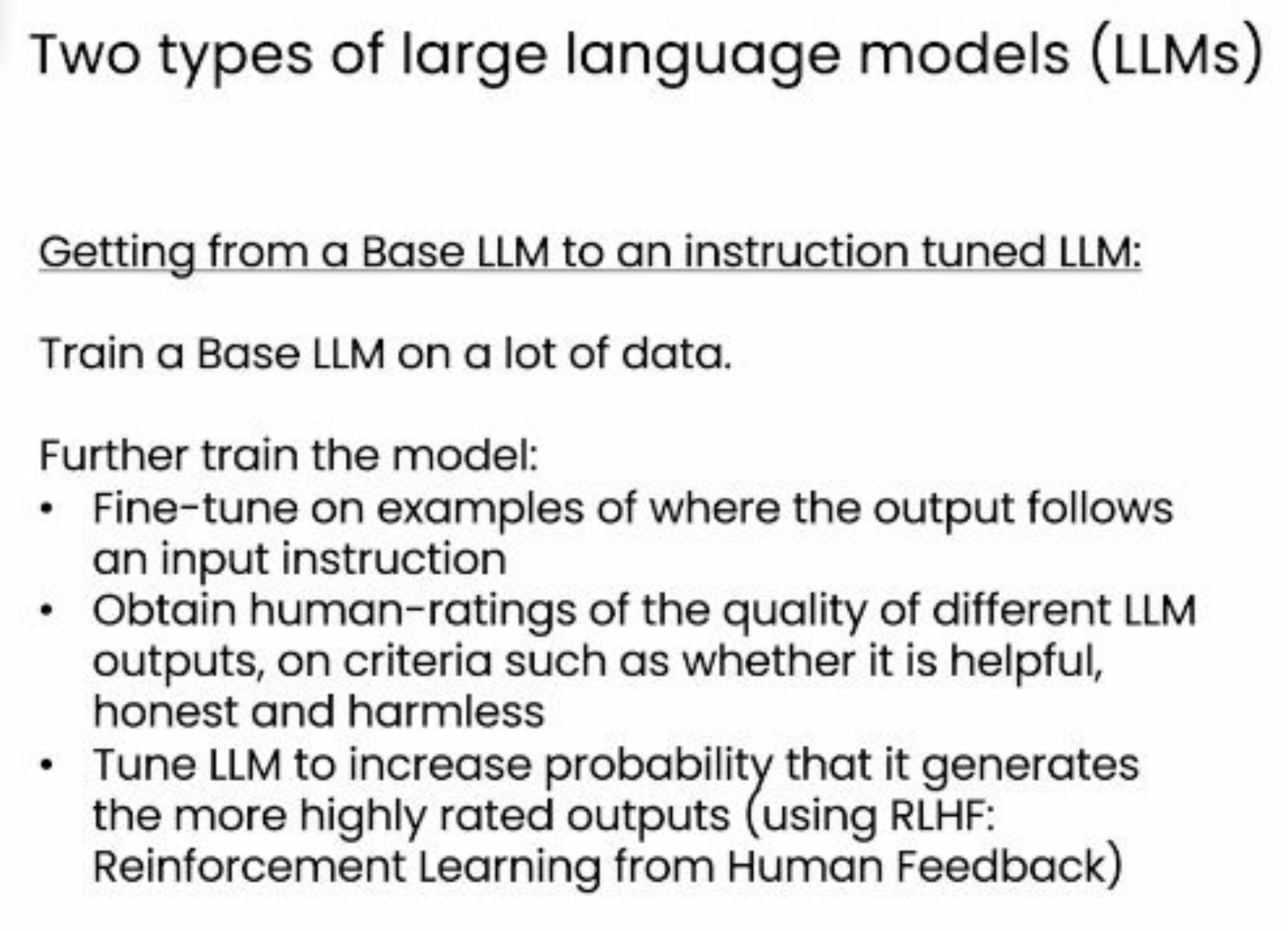
What is the capital of France?
What is France's largest city?
What is France's population?
What is the currency of France?

Instruction Tuned LLM

Tries to follow instructions

What is the capital of France?
The capital of France is Paris.

QUICK RECAP- how are they trained?



Two types of large language models (LLMs)

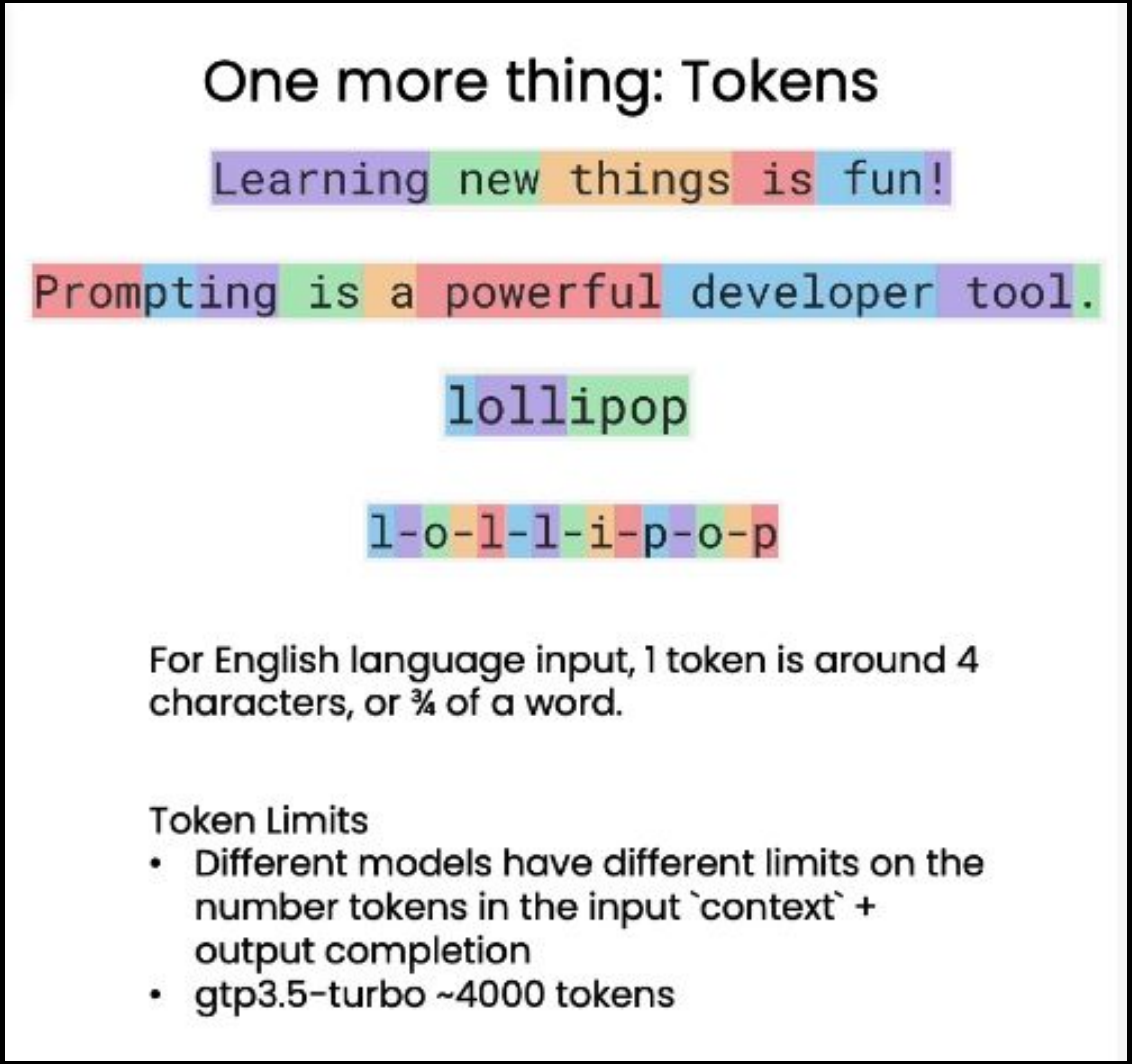
Getting from a Base LLM to an instruction tuned LLM:

Train a Base LLM on a lot of data.

Further train the model:

- Fine-tune on examples of where the output follows an input instruction
- Obtain human-ratings of the quality of different LLM outputs, on criteria such as whether it is helpful, honest and harmless
- Tune LLM to increase probability that it generates the more highly rated outputs (using RLHF: Reinforcement Learning from Human Feedback)

QUICK RECAP - Tokens



One more thing: Tokens

Learning new things is fun!

Prompting is a powerful developer tool.

lollipop

l-o-l-l-i-p-o-p

For English language input, 1 token is around 4 characters, or ¾ of a word.

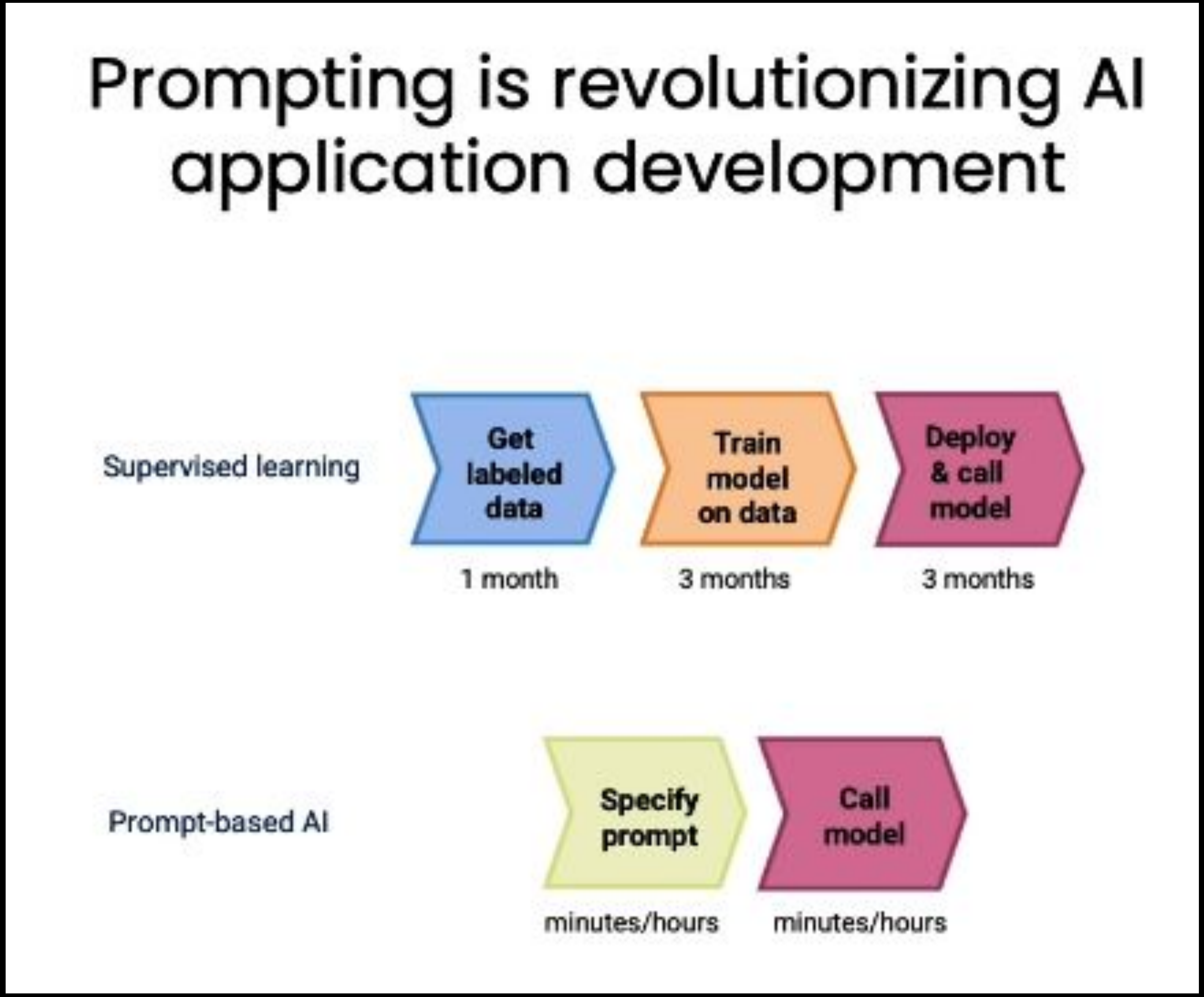
Token Limits

- Different models have different limits on the number tokens in the input `context` + output completion
- gtp3.5-turbo ~4000 tokens

QUICK RECAP- what are LLMs able to do?...

- Summarization
- Extraction
- Transforming
- Infering
- Classification
- Reasoning
- Writing
- Question Answering
- ...

QUICK RECAP - … an how? by prompting!



Prompting is revolutionizing AI application development

Supervised learning

Get labeled data — 1 month

Train model on data — 3 months

Deploy & call model — 3 months

Prompt-based AI

Specify prompt — minutes/hours

Call model — minutes/hours

TIPS / GUIDELINES FOR PROMPTING

1. **Be detailed and specific**
2. **Help the model to think though its answer**
3. **Experiment and iterate**

TIPS / GUIDELINES FOR PROMPTING

- **Be detailed and specific**
- **Help the model to think though its answer**
- **Experiment and iterate**

TIPS / GUIDELINES FOR PROMPTING

1. **Be detailed and specific**

- Give the model sufficient context (e.g. role, broader topic…)
- Set the style and tone of the answer
- Use system-message and user-message
- Use delimiters ($$$, §§§, ***, """", ''', <>)
- Ask for a structured output (e.g. json, html)
- Check user input for conditions
- Use examples (few-shot prompting)
- …

- Give sufficient context for LLM to complete the task

> Help me write an email asking to be assigned to the legal documents project.

- Give sufficient context for LLM to complete the task

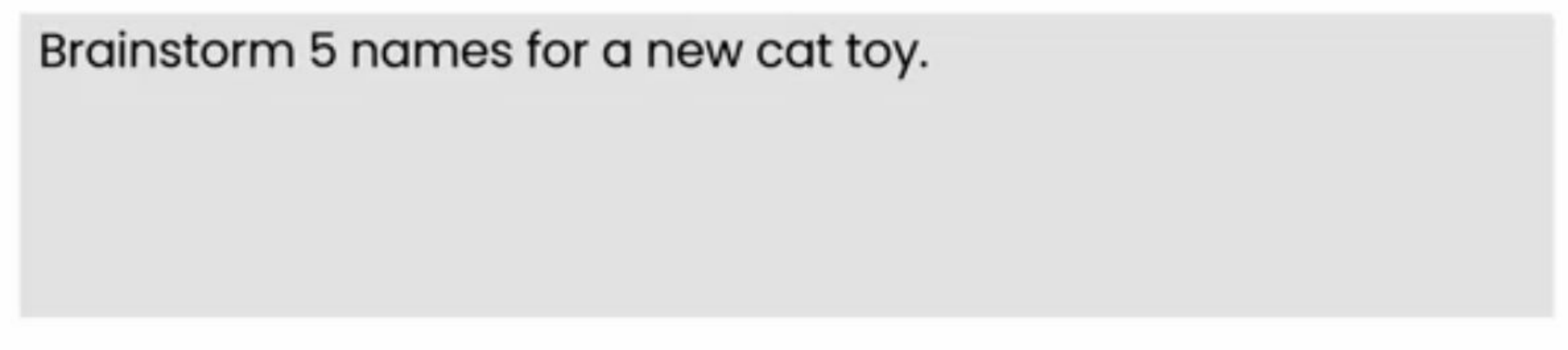
> Help me write an email asking to be assigned to the legal documents project.
>
> I’m applying for a job on the legal documents project, which will check legal documents using LLMs. I have ample experience prompting LLMs to generate accurate text in a professional tone.

TIPS / GUIDELINES FOR PROMPTING

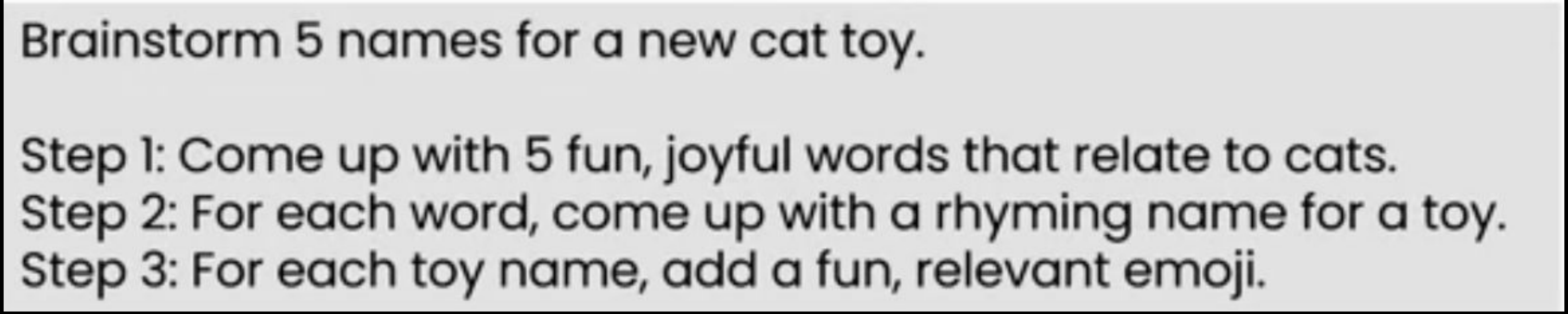
- Be detailed and specific
- Help the model to think though its answer
- Experiment and iterate

- Give sufficient context for LLM to complete the task
- Describe the desired task in detail

> Help me write an email asking to be assigned to the legal documents project.
>
> I’m applying for a job on the legal documents project, which will check legal documents using LLMs. I have ample experience prompting LLMs to generate accurate text in a professional tone.
>
> Write a paragraph of text explaining why my background makes me a strong candidate to this project and advocate for my candidacy.



Brainstorm 5 names for a new cat toy.



Brainstorm 5 names for a new cat toy.

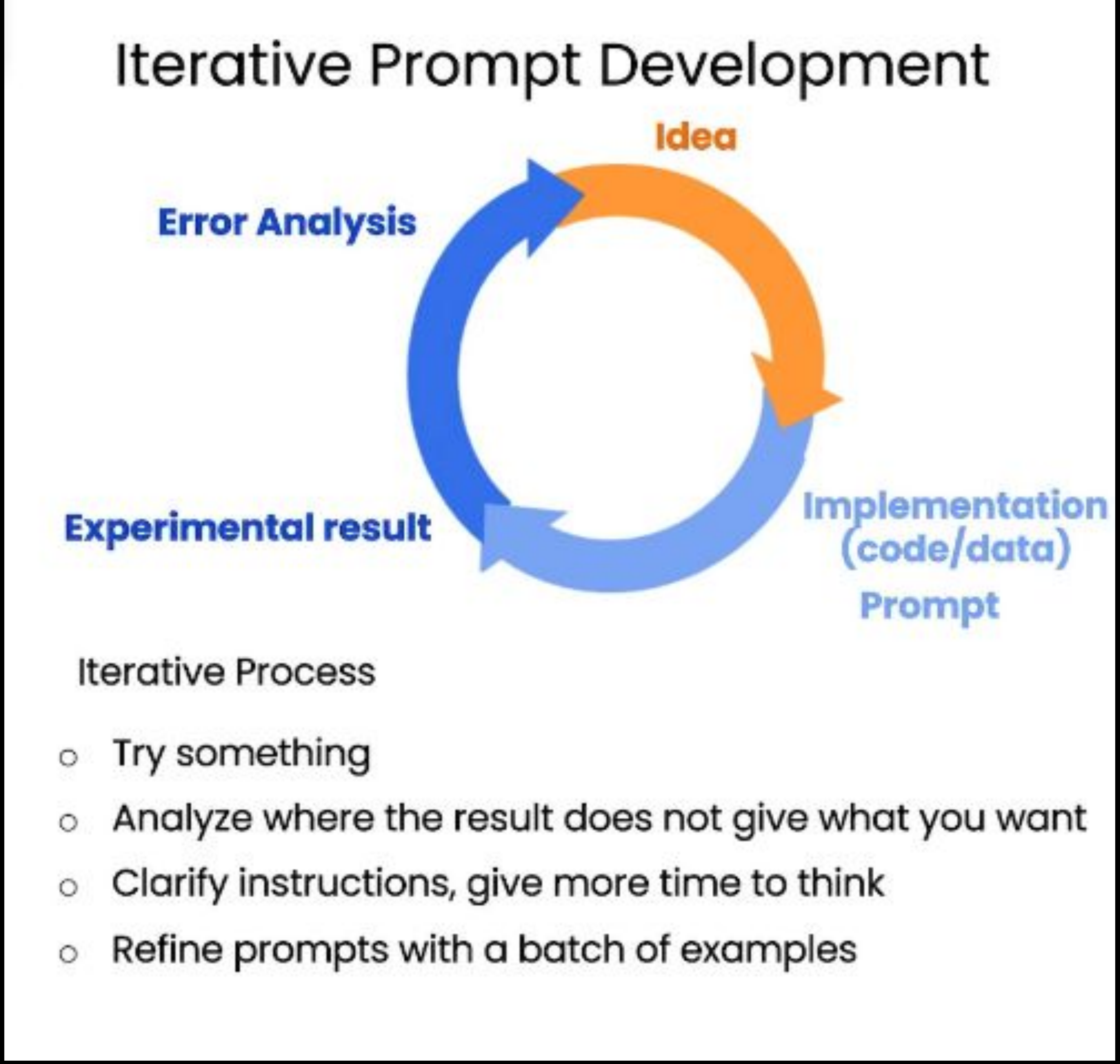
Step 1: Come up with 5 fun, joyful words that relate to cats.
Step 2: For each word, come up with a rhyming name for a toy.
Step 3: For each toy name, add a fun, relevant emoji.

Model response:

Step 1: 5 fun, joyful words that relate to cats:	**Step 2:** Rhyming names for a toy based on those words:	**Step 3:** Add a fun, relevant emoji for each toy name:
Purr	Purr-Twirl	Purr-Twirl 🌀
Whisker	Whisker-Whisper	Whisker-Whisper 🫢
Feline	Feline-Beeline	Feline-Beeline 🐾
Pounce	Pounce-Bounce	Pounce-Bounce ⚽
Meow	Meow-Wow	Meow-Wow 😺

TIPS / GUIDELINES FOR PROMPTING

- Be detailed and specific
- Help the model to think though its answer
- Experiment and iterate



Iterative Prompt Development

Iterative Process

- Try something
- Analyze where the result does not give what you want
- Clarify instructions, give more time to think
- Refine prompts with a batch of examples

“The key to being an effective prompt engineer isn't so much about knowing the perfect prompt. It's about having a good process to develop prompts that are effective for your application.” - Andrew Ng

CHAIN-OF-THOUGHT PROMPTING

Chain of Thought (CoT) prompting encourages the LLM to explain its reasoning. Combine it with few-shot prompting to get better results on more complex tasks that require reasoning before a response.

BREAKOUT ROOMS: PROMPT-HUBS

In your breakout group visit the LangChain and Haystack prompt hubs and discuss the following points:

- How would you rate the overall usefulness (number of prompts, quality of prompts,etc.) of these prompt hubs?

- What are some examples of interesting prompts you can find on these prompt hubs? Collect 1-2 examples.

BREAKOUT ROOMS: PROMPT-HUBS

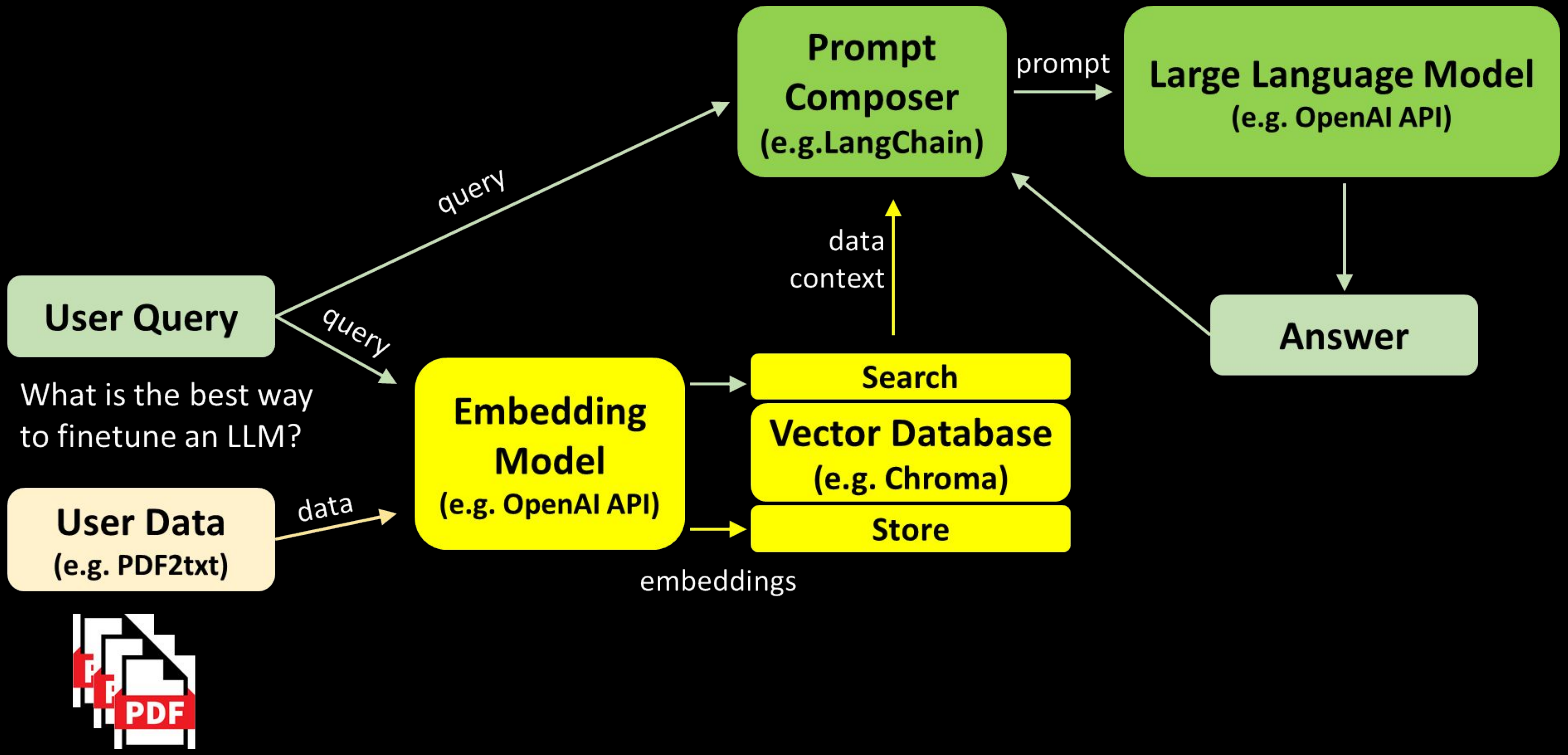
- LangChain prompt hub:
- https://smith.langchain.com/hub

- Haystack prompt hub:
- https://prompthub.deepset.ai/

BREAKOUT ROOMS: PROMPT-HUBS

- **In your group, think about at least one use case that could be tackled by good prompting and look for suggested solutions in both hubs.**
- **Discuss:**
    - What do you find helpful about the hubs?
    - What do you find critical?

ANATOMY OF AN APP



https://www.youtube.com/watch?v=dXxQ0LR-3Hg

ANATOMY OF AN APP

A simple recipe for an LLM app:

- Preparing your data: you already know how
- Writing professional prompts: Week 2
- Managing conversations, memory, and models with LangChain: Week 3
- Using your data for retrieval-enhanced output: Weeks 4 and 5
- Developing appealing GUIs with Gradio (or Streamlit): Week 6
- Getting high quality results through output evaluation: Week 7
- Making it safer and cheaper (?) with open-source models: Week 8

BREAKOUT ROOMS: PROJECT IDEAS

For each idea make a project card:

- Working Title: For example, "Chat with Excel"
- Data Sources: For example, Excel tables or text-based tables.
- Functions: For example, users can pose questions and receive graphical and statistical reports in response.
- Use Cases: For instance, a layperson can quickly analyze large datasets.
- Challenges: For example, generating correctly formatted Python code for graphics and statistical analysis.
- People interested: Horst & Elvira

BREAKOUT ROOMS: PROJECT IDEAS

You have 15 min in the breakout rooms.

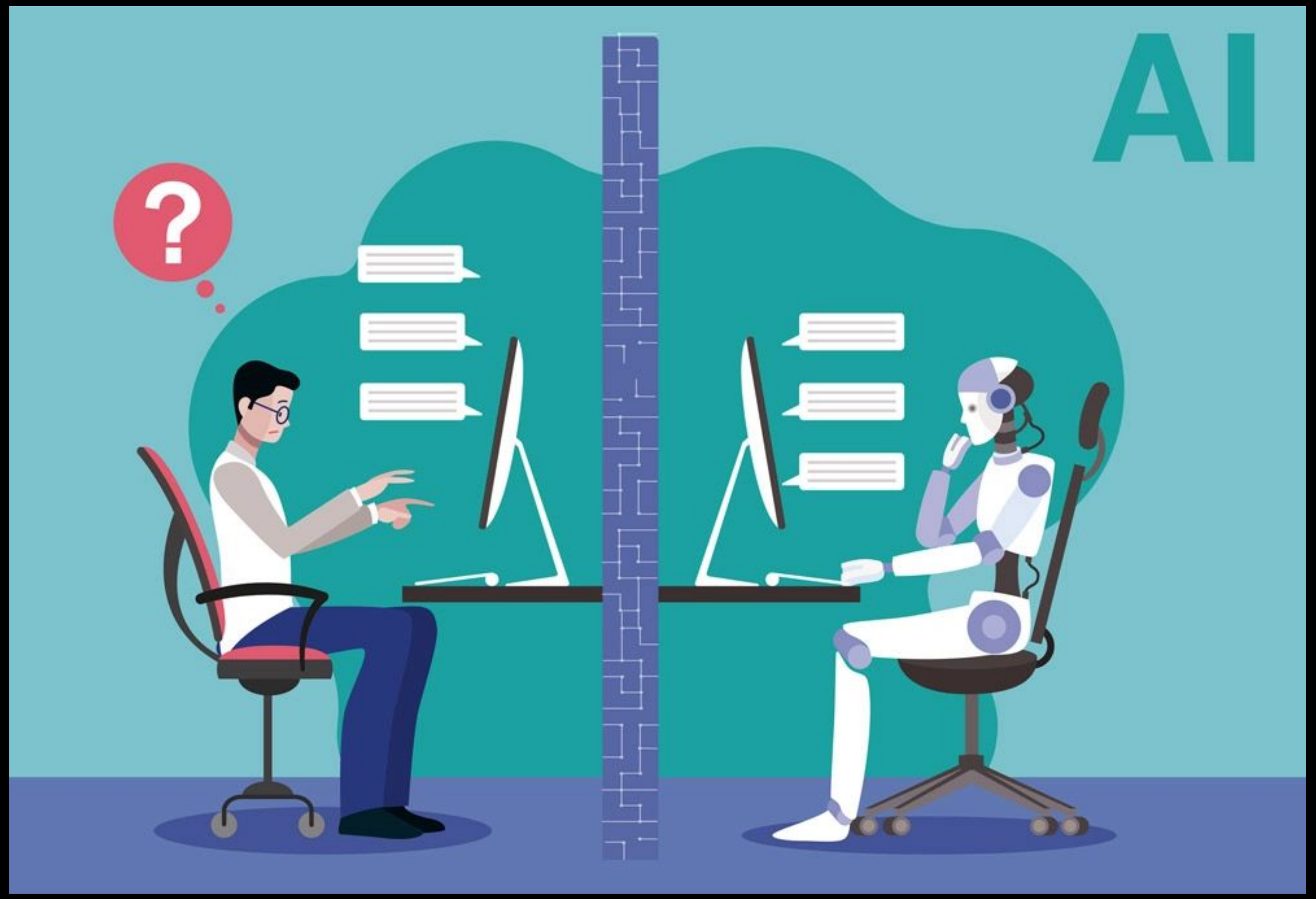
Pin your project cards here!

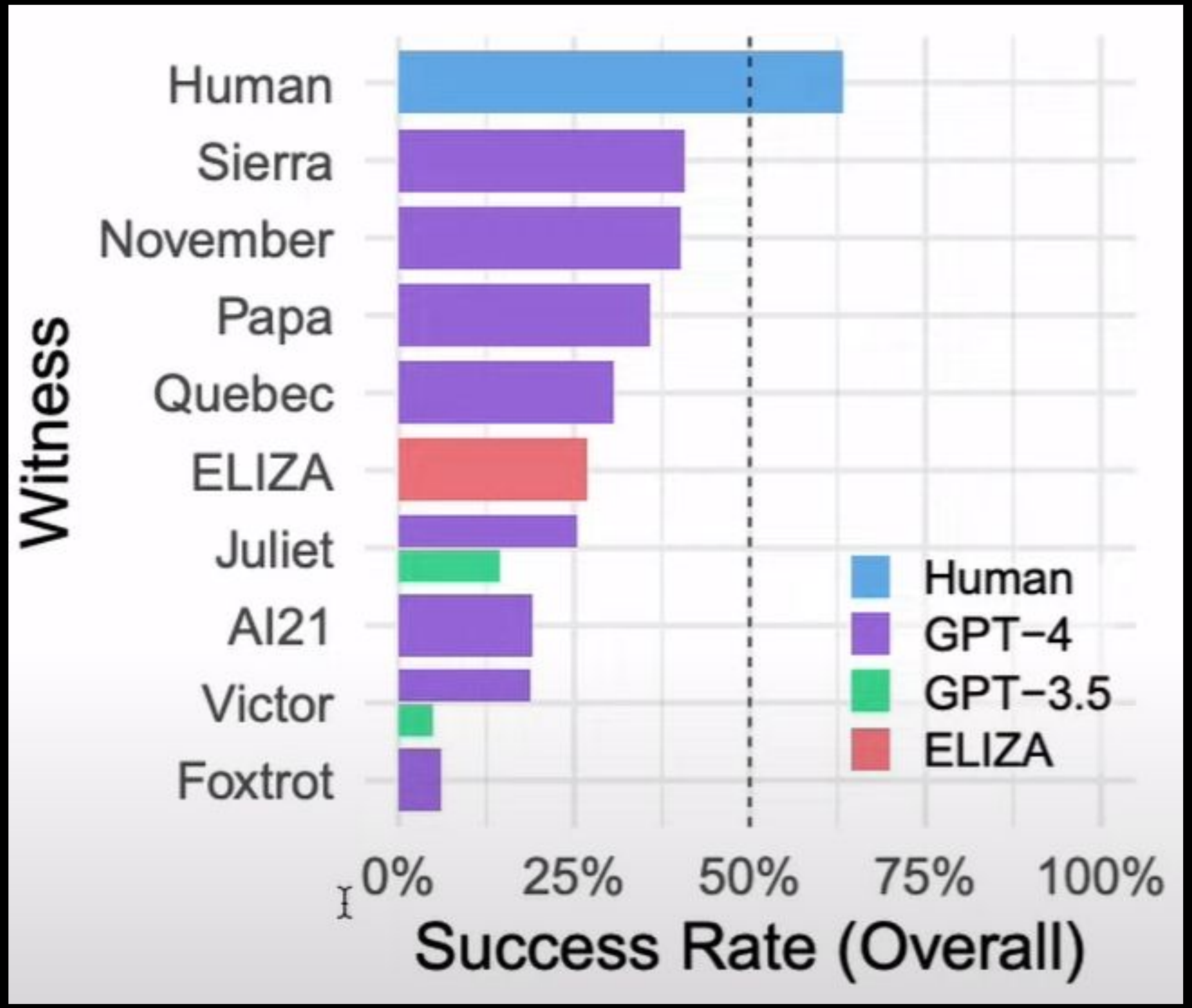
Present your ideas in the plenum!

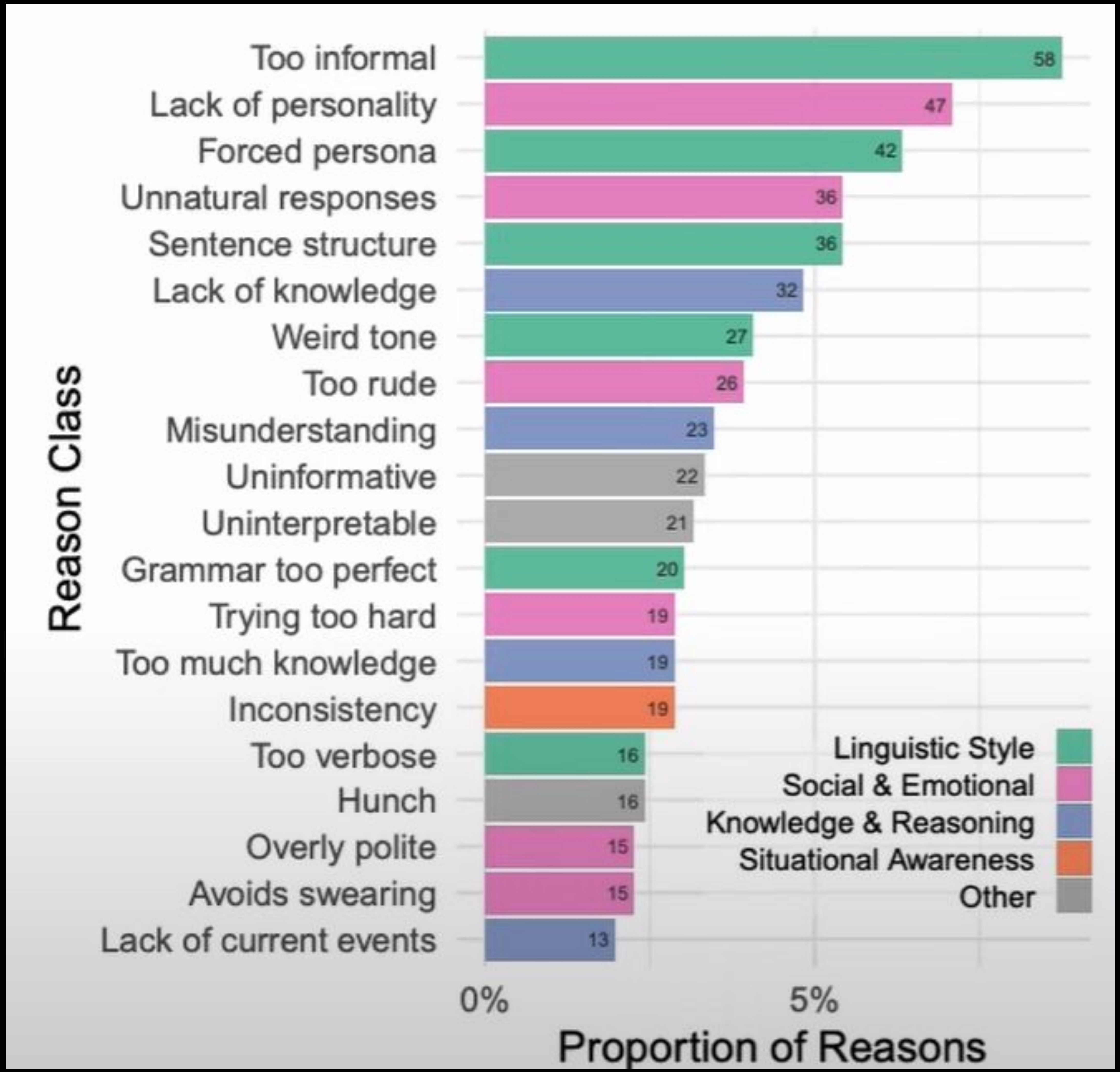
DEFAULT PROJECTS

- Turing-Test App
- Ted-Talk-Chat
- Chat with multiple documents
- Chat with podcasts
- Chat with structured data (CSV)
- Personal Assistant with access to Calendar and E-Mail
- Synthetic data generation for fine-tuning of NLP models

Default project: Turing Chat







https://arxiv.org/pdf/2310.20216.pdf
https://www.youtube.com/watch?v=_CmNm6qJ43c

Default project: TED Life & Business Coach

User question

e.g. "How can I overcome procrastination?"

System initiates a conversation to “understand” the problem

e.g. f “What specific challenges or obstacles are you currently facing in relation to {problem}?”

System engages in chain of thought reasoning

e.g identify domain of problem, summarize conversation, generate keywords for search

System searches a table of TED talk URLs and transcripts

https://www.kaggle.com/datasets/thedatabeast/ted-talk-transcripts-2006-2021

e.g. Tim Urban: Inside the mind of a master procrastinator

System generates output in JSON and displays results

Speaker name of relevant TED talk: ...
Title of relevant TED talk: ...
50-word summary of relevant TED talk: ...
5 actionable key messages of the TED talk:
URL of the TED talk: [Watch the TED talk here](https://www.example.com/ted-talk-url)

Default project: Chat with multiple documents

- Upload multiple documents to your application
- Store the documents in a vector store and add metadata
- Retrieve parts of the documents based on similarity measures from the vector store
- Generate an answer to a chat message that references the retrieved document by the stored metadata
- Variations: Instead of documents use transcribed podcasts or use structured data like CSV files as data

Default project: Personal Assistant

- A chat agent that has access to multiple tools that connect with the API of your calendar, e-mail, etc.
- Automatically creates, updates, deletes calendar entries based on the the chat messages
- Retrieves new e-mails, categorizes the e-mails and if necessary already creates an answer draft for you

Default project: Synthetic Data Generation

- Use Large Language Models to create synthetic Natural Language Processing datasets to fine-tune smaller NLP models

- Smaller NLP models can achieve high accuracies when fine-tuned for specific tasks and are (potentially) cheaper and more easy to host than LLMs

- Example: A dataset to train a text classifier to classify work sheets as part of one of the school subjects

TASKS UNTIL NEXT WEEK

- Watch the short course “LangChain for LLM Application Development” from Deeplearning.AI

- Work through the Jupyter notebook “HomeworkLangChain”