

Tutorial: Beyond Prompt Engineering Speeding Up Prototyping and Development

Albrecht Schmidt, LMU Munich

Inspired by hcilab-Winterschool Tutorial from Steeven Villa and Meagan Loerakker as well as by <https://www.promptingguide.ai/techniques/> and https://en.wikipedia.org/wiki/Prompt_engineering

Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

Recommendations

Zero Shot Prompting

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

1 Translate English to French:

← *task description*

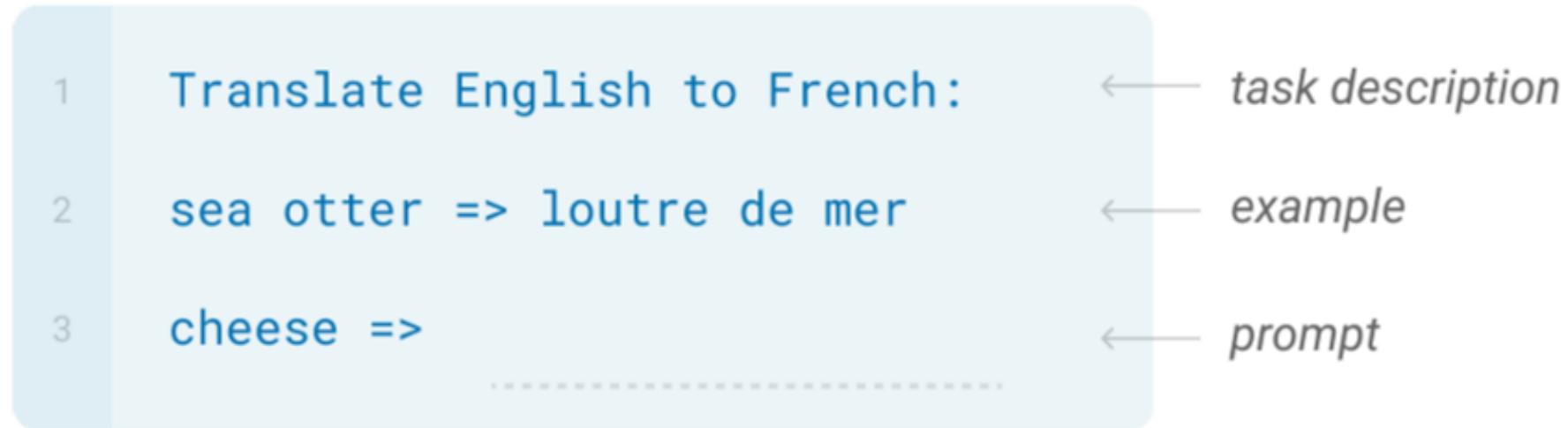
2 cheese =>

← *prompt*

Brown, Tom, et al. "Language models are few-shot learners." *Advances in neural information processing systems* 33 (2020): 1877-1901.
https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

One-Shot Prompting

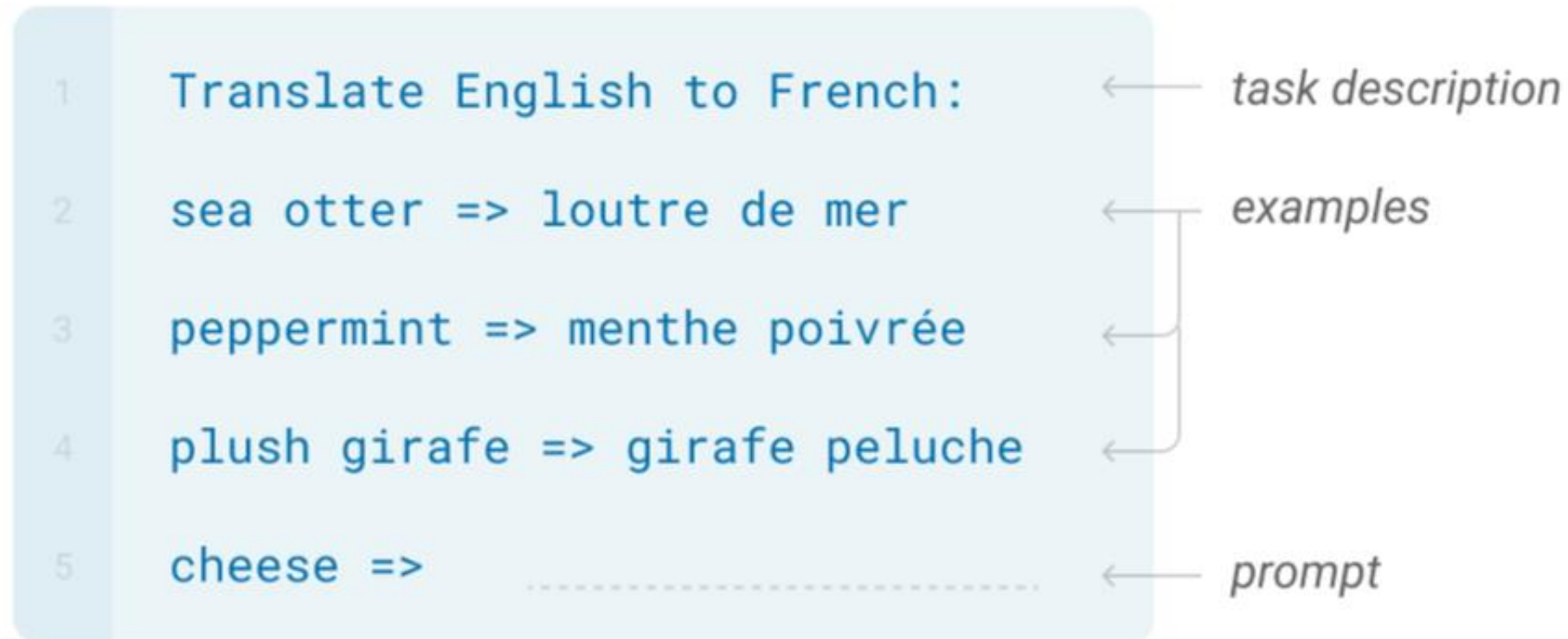
In addition to the task description, the model sees a single example of the task. No gradient updates are performed.



Brown, Tom, et al. "Language models are few-shot learners." *Advances in neural information processing systems* 33 (2020): 1877-1901.
https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf

Few Shot Prompting

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.



Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

Recommendations

Prompt Instruction Cheat Sheet

Content of the Instruction	Task of this Component	Dos (exemplary)	Don'ts (exemplary)
Bot Persona	Define a behavior with many individual facets in just a few words or for a story driven interaction with an NPC or avatar, define a bot persona with a backstory that makes the bot's behavior more exciting and interactive. Provide names, dates, places in the story.	Act as a sensitive elderly psychotherapist ... as a patient support staff ... as a professional journalist ... as a car in love with its driver ... you are Selma Brand, a 28-year old gunslinger who lost her family in a bandit raid on their farm. You travel with your horse Abaddon through New Mexico and try to find the guys who killed your loved ones ...	Act as a helpful AI
User persona	Define the level of detail, complexity, required knowledge of the recipient in a few words	Explain it like to someone with an MSc in software engineering ... like to a 5-year-old child	Answer the user ...
Verb	Define the task. To "answer" is not the task, because the model will always respond.	Summarize, list, ask, classify, explain, format, extract, translate to Spanish ...	Answer the question, give me, ...
Output format	Define the exact format of the output	A list of steps, a formula, python code, a JSON, a number, a recipe. [you can define the output format in a separate section as well]	Answer, text
Length, level of detail	Describe the output format and number of elements	Up to 3 sentences ... 10 examples with product name and price	Short, few, detailed
Tonality	Define the style, the way the model should convey the message	Empathetic, confident, aggressive, moaning, sarcastic, witty, stuttering	Friendly, neutral, smart, intelligent
Edge cases	Define responses for special situations: Model has no information ... does not understand the question ... if the user gets aggressive, toxic ...	If you can't answer the question based on the provided context, tell the user you can't ...	Answer only on the basis of your knowledge ...
Whitelist conversation topics	Define all the topics the model can talk about, not the other way round (list forbidden topics)	Discuss only the operation of the coffee machines of the manufacturer XYZ and talk about coffee culture. Friendly refuse to chat on something else.	Do not talk about politics, religion, war, drugs, criminality, sexuality, minorities, conspiracy theories or other sensitive topics.

Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

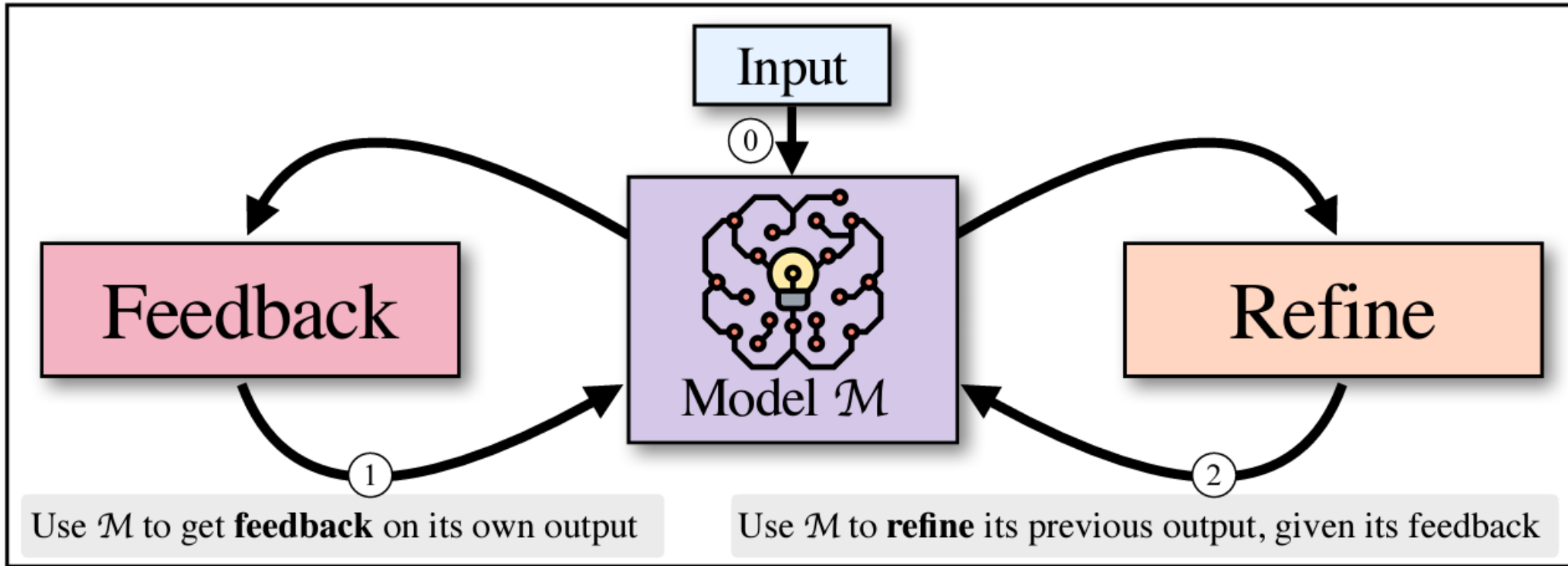
Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

Recommendations

Self-Refine: Iterative Refinement with Self-Feedback



Madaan, Aman, et al. "Self-refine: Iterative refinement with self-feedback." *Advances in Neural Information Processing Systems* 36 (2024).
https://proceedings.neurips.cc/paper_files/paper/2023/file/91edff07232fb1b55a505a9e9f6c0ff3-Paper-Conference.pdf

Self-Refine: Iterative Refinement with Self-Feedback

Task (e.g., improve some source code)

Prompt: Make suggestions to improve the following python code.

<Code is here in the prompt>

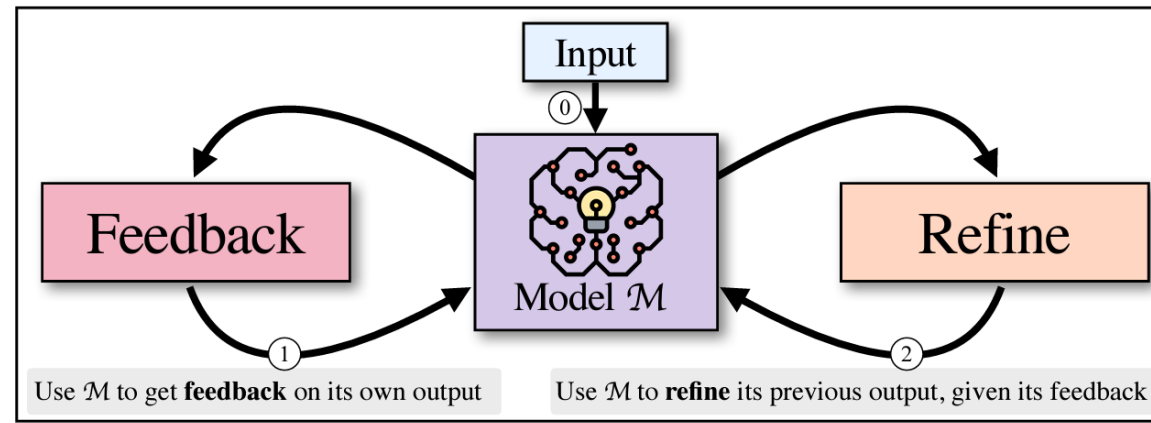
Just provide suggestions on how to make it faster and safer, don't suggest specific changes, give me general suggestions.

-> Response making suggestions

Prompt: improve the code with these suggestions

-> Revised code

- Applying the idea



Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

Recommendations

Divide and Conquer Iterative Creation

- Prompt the LLM to divide the task/problem
- Prompt to divide the subtasks further
- ...
- Prompt to get answers to the sub tasks
- Fuse the solutions

Task (e.g., write an essay on X)

Prompt: I want to write an essay on X.
What would relevant aspects I need to take into account?

-> List of aspects

Prompt: What is a good structure for an essay on X? What sections should I have?

-> structure, section headings

Prompt: Write some keywords for the section 1?

2 3
...

-> Keywords sections 2

-> Keywords sections 2

-> Keywords sections 1

Prompt: Suggest Text for section 1?

2 3
...

-> Text section 2

-> Text section 2

-> Text section 1

Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

Recommendations

Chain-of-Thought Prompting

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

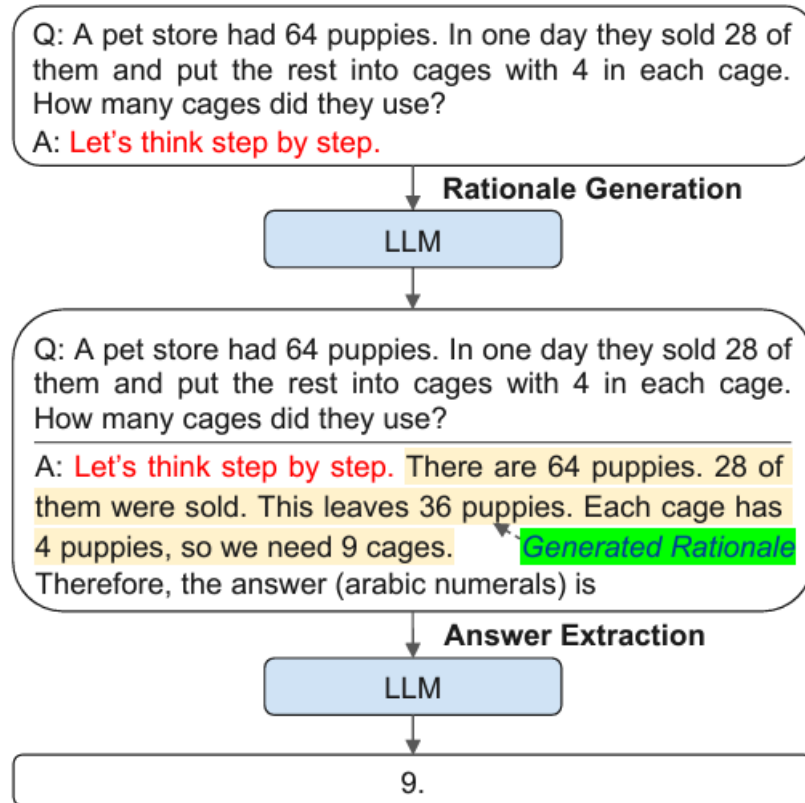
A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

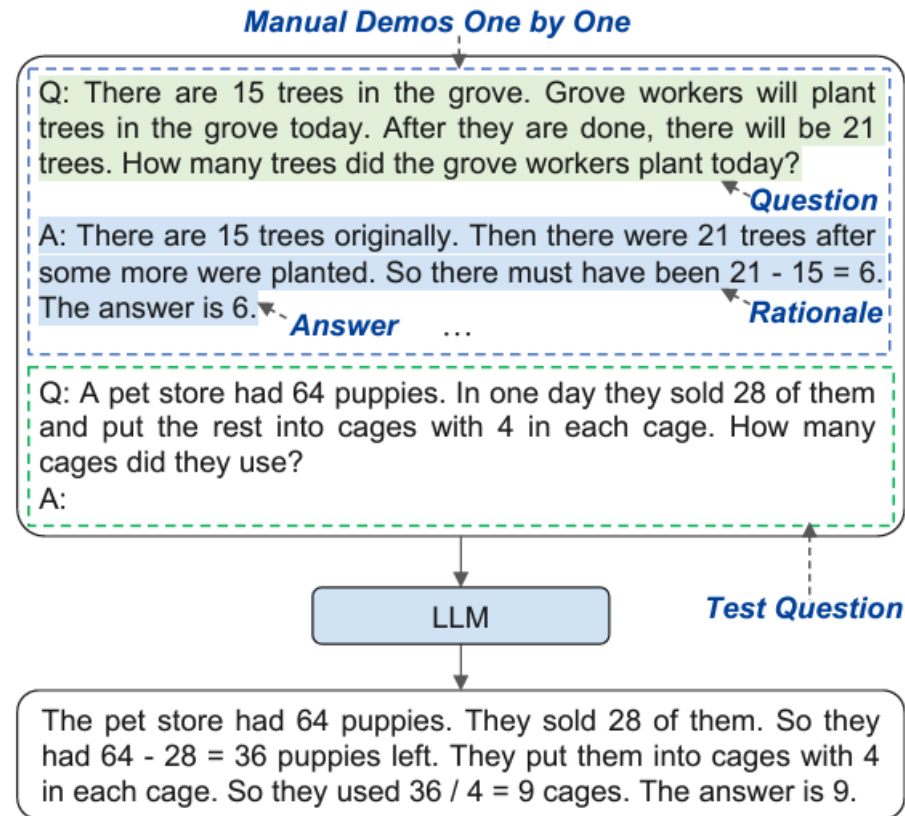
Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✅

Automatic Chain-of-Thought Prompting



(a) Zero-Shot-CoT



(b) Manual-CoT

Zero-shot CoT Prompting

(a) Few-shot

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The answer is 8. ✗

(b) Few-shot-CoT

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are $16 / 2 = 8$ golf balls. Half of the golf balls are blue. So there are $8 / 2 = 4$ blue golf balls. The answer is 4. ✓

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: The answer (arabic numerals) is

(Output) 8 ✗

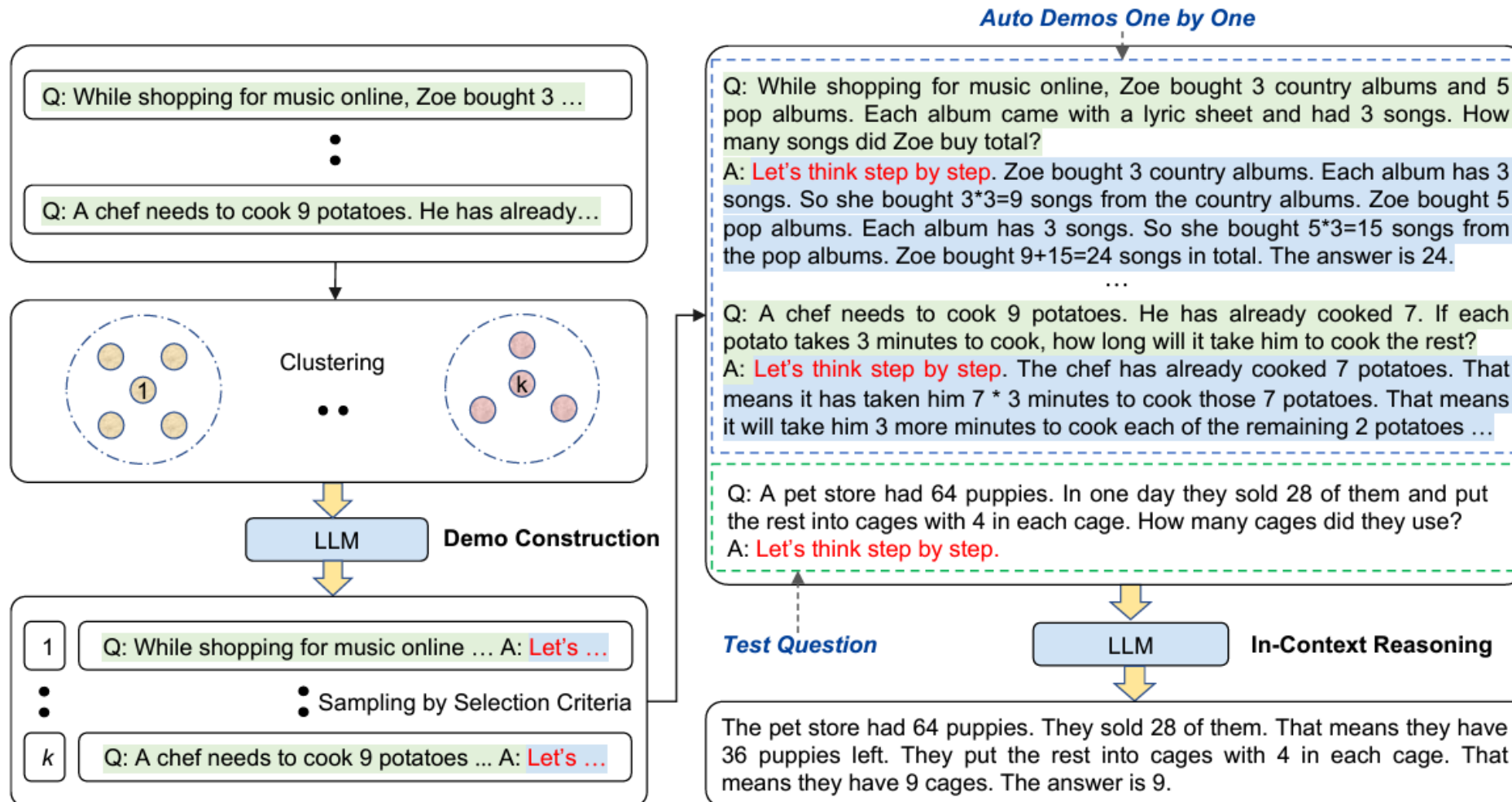
(d) Zero-shot-CoT (Ours)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: **Let's think step by step.**

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓

Automatic Chain-of-Thought Prompting



Zhang, Zhuosheng, et al. "Automatic chain of thought prompting in large language models." *arXiv preprint arXiv:2210.03493* (2022).

Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

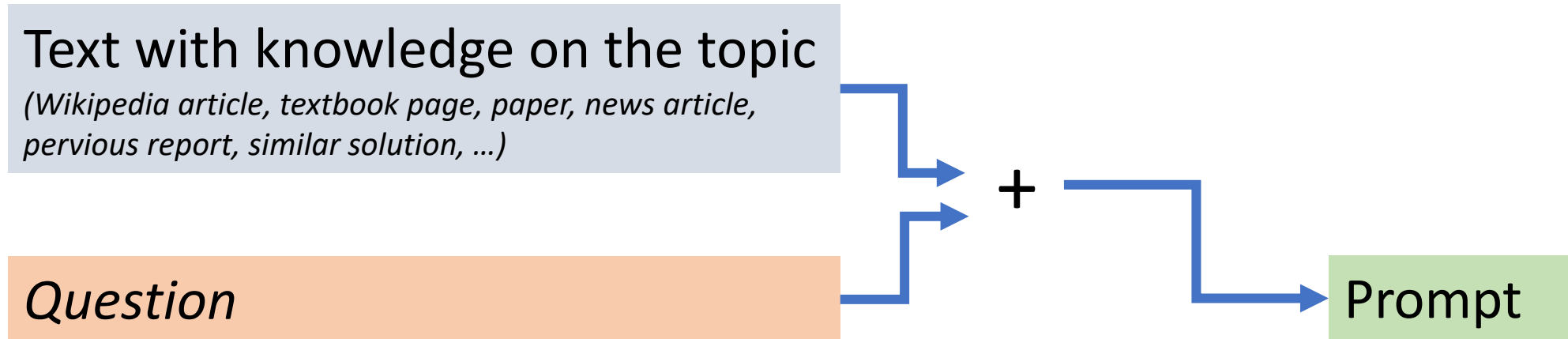
Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

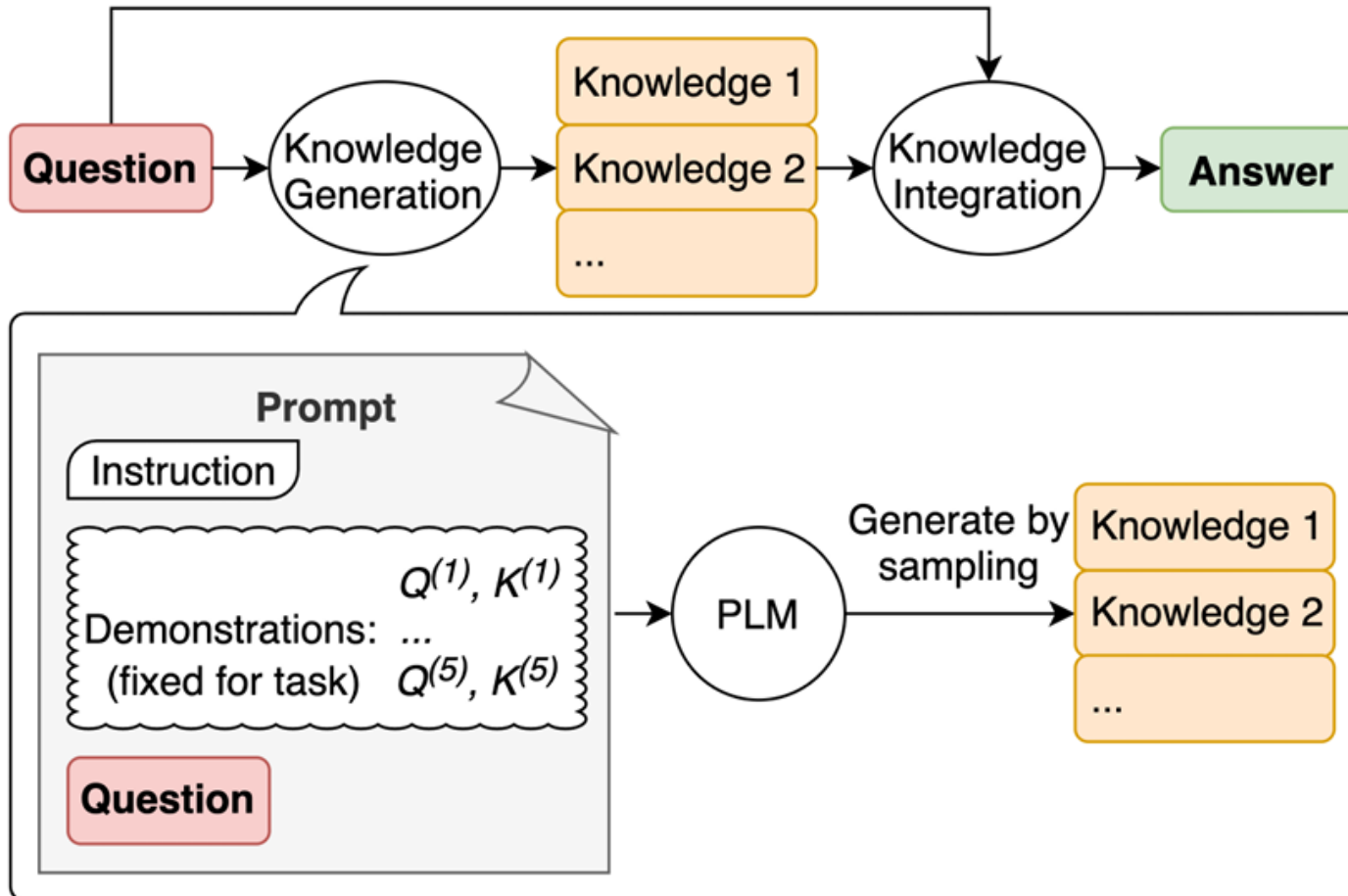
Recommendations

Prompting with Knowledge



- The knowledge included in the prompt, does not require information to answer the question
- The knowledge “steers” the answer into the right direction

Knowledge Generation Prompting



Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

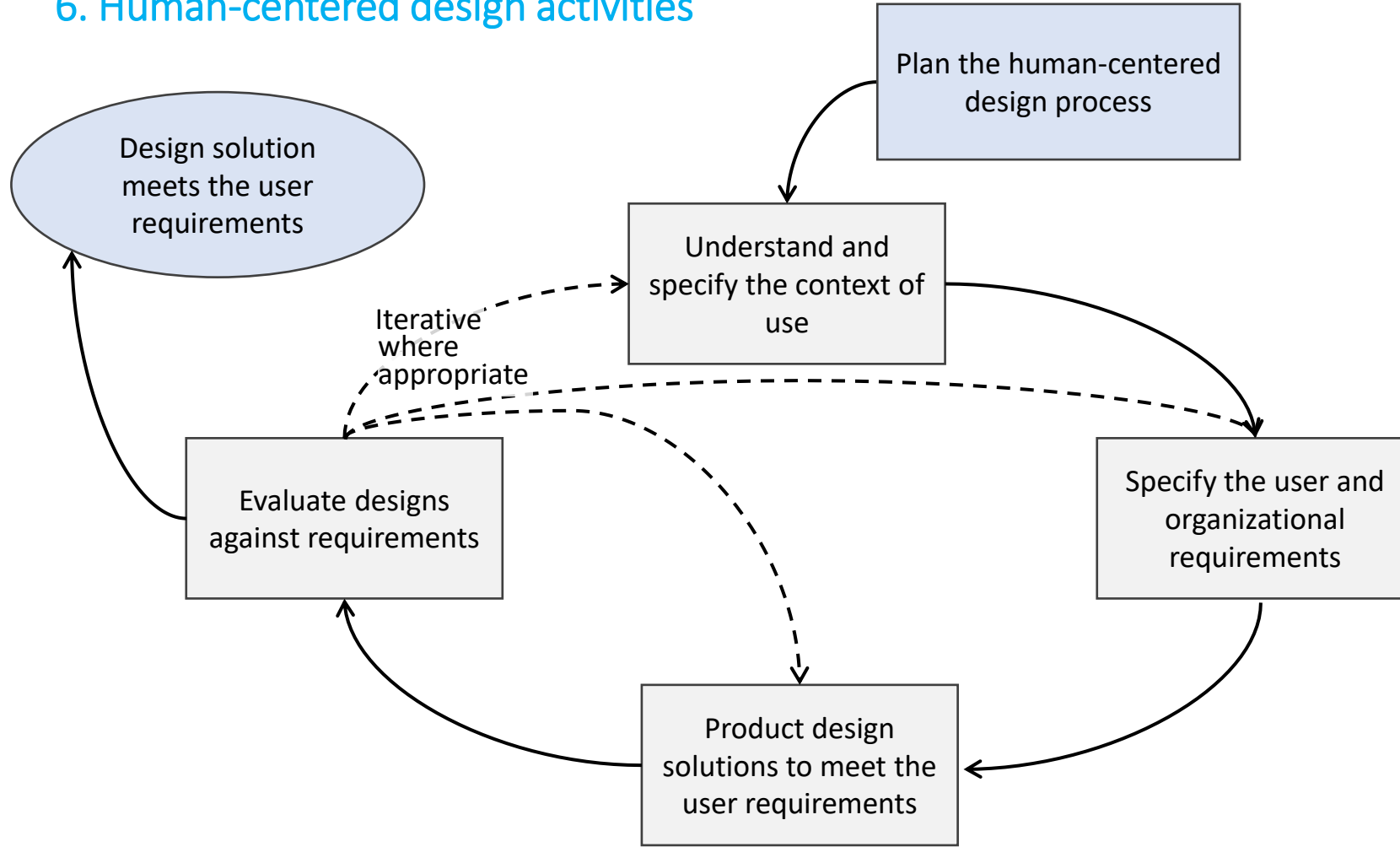
Replacing developer?

Recommendations

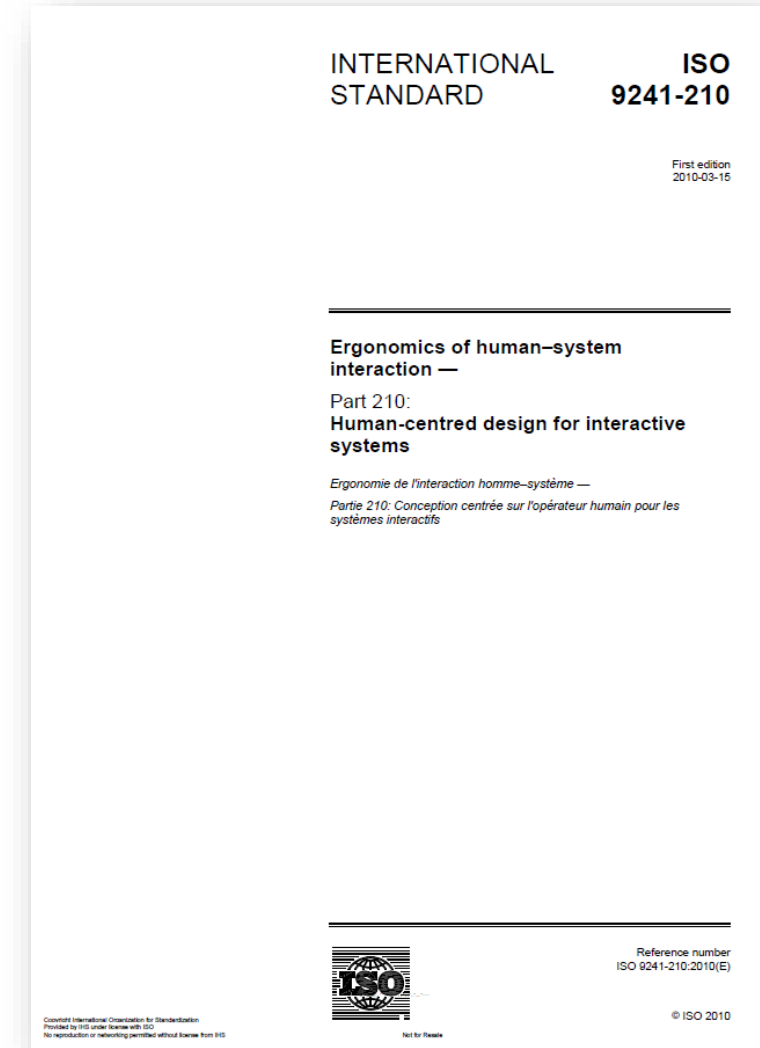
ISO 9241-210

Human-centered design for interactive systems

6. Human-centered design activities



ISO 9241-210:2019(EN) Human-centered design for interactive systems



Understand User Needs and Context

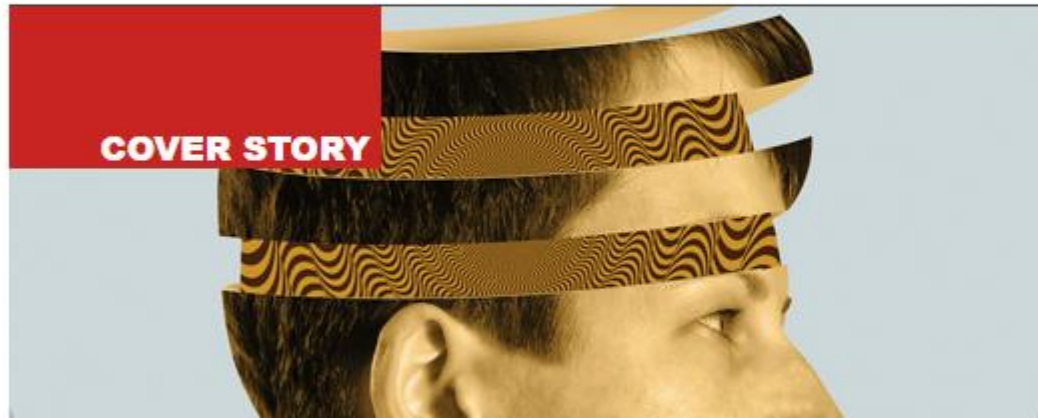
Specifying Requirements

- Stakeholders
- Personas
- Focus groups
- Interview questions
- Survey questions and surveys
- Transcription and summarization

“brainstorming”?

Generating “fictitious”
answers to questionnaires?

Generating “fictitious”
discussions in focus groups?



COVER STORY

Simulating the Human in HCD with ChatGPT: Redesigning Interaction Design with AI

Albrecht Schmidt, Passant Elagroudy, Fiona Draxler, Frauke Kreuter, Robin Welsch

News: [Google DeepMind used a large language model to solve an unsolved math problem](#) January 29, 2024

IN THIS ISSUE

- Enter
- WELCOME
- Columns
- Forums
- Features
- Calendar
- Exit
- Conversations in Sketch

- ▶ TABLE OF CONTENTS
- ▶ VIEW IN DIGITAL LIBRARY
- ▶ DIGITAL EDITION FORMAT
- ▶ VIEW IN PDF FORMAT



SIGN IN

[Forgot Password?](#)

[Create Web Account](#)

[Join ACM](#)

[Join SIGCHI](#)

[Subscribe to Interactions](#)

Albrecht Schmidt, Passant Elagroudy, Fiona Draxler, Frauke Kreuter, and Robin Welsch. 2024. Simulating the Human in HCD with ChatGPT: Redesigning Interaction Design with AI. interactions 31, 1 (January - February 2024), 24–31. <https://doi.org/10.1145/3637436>



Simulating the Human in HCD with ChatGPT Redesigning Interaction Design with AI

Albrecht Schmidt, LMU Munich,
Passant Elagroudy, German Research Center for Artificial Intelligence,
Fiona Draxler, LMU Munich, Frauke Kreuter, LMU Munich,
Robin Welsch, Aalto University

Insights

- Generative AI can enhance the human-centered design process.
- LLMs encode human experiences and can be used to emulate users at a large scale.
- Use of generative AI must be made transparent in human-centered design.
- Generative AI will not replace user studies but rather will enhance the toolkit of HCI researchers and practitioners.

IMAGE BY BRADLEY / SHUTTERSTOCK.COM

INTERACTIONS.ACM.ORG

Human-centered design (HCD) puts the human at the center of interactive system design. Can we do that without actively including the human user in the process? Is that still HCD? We believe that large language models (LLMs) and generative AI will fundamentally change the way we design and implement interactive systems. Models are not new to HCI, but the scale at which LLMs can support the design process is changing their value and applicability. The use of models in design decisions for interactive systems has a long tradition in HCI. Task modeling is commonly used in menu design.

There are attempts to model human physiology as well as cognitive processes, and there is an entire journal (<https://www.springer.com/journal/11257>) and conference (UMAP; <https://www.um.org/umap2023/>) devoted to user modeling. Models are helpful since they do not require the user during the design process and thus potentially speed up that process. Creating models, however, has been difficult and cumbersome, which is why human-centered design, which involves people in the process, has been the most common approach to creating usable systems. Putting the

JANUARY-FEBRUARY 2024 INTERACTIONS 25

Albrecht Schmidt, Passant Elagroudy, Fiona Draxler, Frauke Kreuter, and Robin Welsch. 2024. Simulating the Human in HCD with ChatGPT: Redesigning Interaction Design with AI. interactions 31, 1 (January - February 2024), 24–31. <https://doi.org/10.1145/3637436>

Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

Recommendations

AL

What interface elements should a time tracking application for the workplace have?



A time tracking application for the workplace should have a user-friendly and efficient interface that caters to the needs of both employees and managers. Here are some key interface elements to consider:

1. Login/Registration: A secure login/registration system to authenticate users and manage user profiles.
2. Dashboard: A central dashboard that provides a quick overview of the user's tracked time, ongoing tasks, project progress, and any relevant notifications.
3. Timer: A simple and easily accessible timer to start, pause, and stop time tracking for tasks or projects.
4. Task/Project Management: A section to create, edit, and manage tasks or projects, including assigning team members, setting deadlines, and tracking progress.
5. Timesheets: A detailed timesheet view that allows users to input and review their time entries, broken down by task, project, or date.

Midjourney Bot a 1x1 inch OLED display on top of a finger ring, th



Midjourney Bot heute um 10:41 Uhr

a 1x1 inch OLED display on top of a finger ring, the display shows a news article with text and image - Variations by @albrecht.schmidt (fast)



Midjourney Bot gestern um 12:05 Uhr

a finger ring with , worn on a finger, with a small LED screen on top to replace the stone - @albrecht.schmidt (fast)



<https://www.midjourney.com>

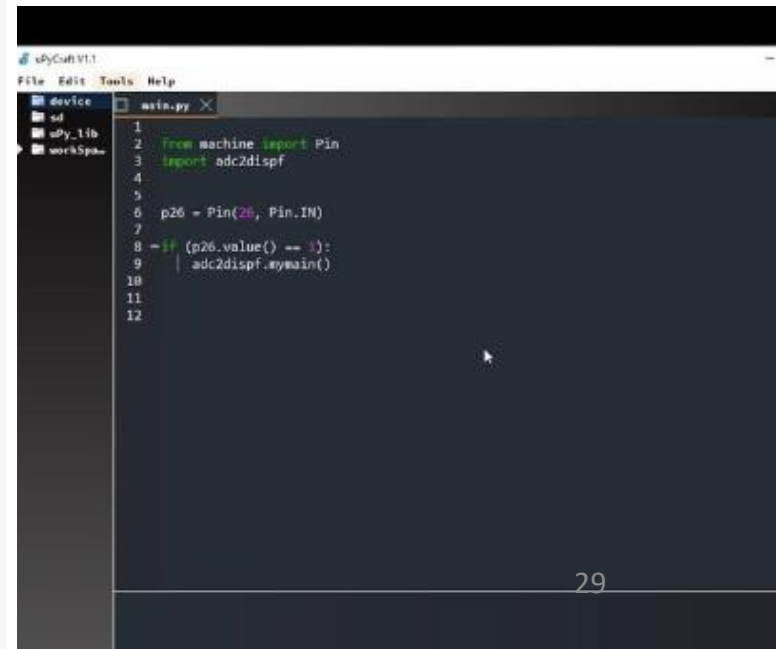
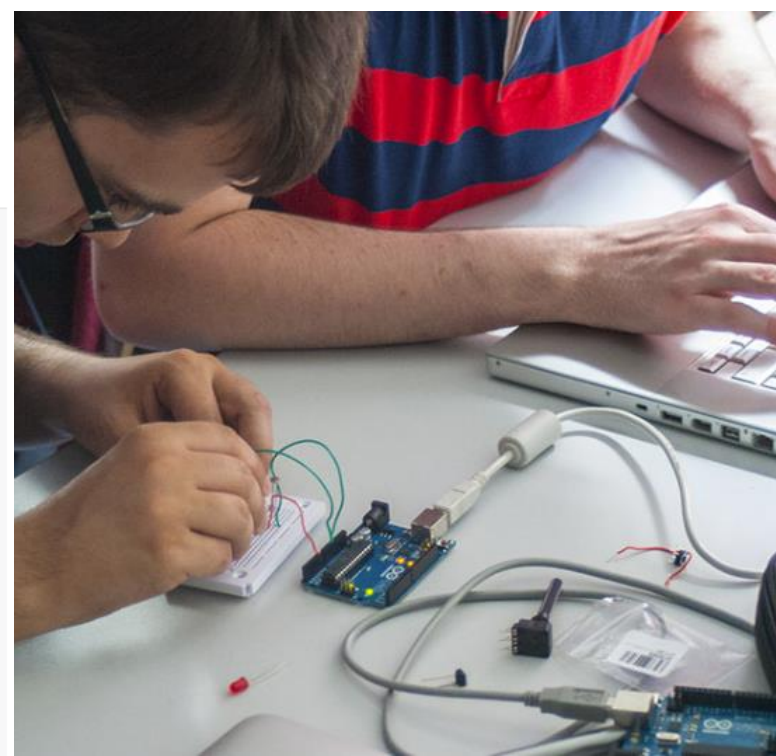
What sensors should I use for a plant monitoring systems? Include suggestions for specific sensors that work with Arduino.



A plant monitoring system can involve several parameters depending on the level of sophistication you're aiming for. Here are some basic parameters you might want to monitor, and sensors suitable for each:

1. **Soil Moisture:** This is one of the most basic parameters to monitor in a plant. A simple capacitive soil moisture sensor, like the DFRobot Capacitive Soil Moisture Sensor, can do a great job. Capacitive sensors are generally better than resistive sensors because they're less corrosive.
2. **Temperature and Humidity:** DHT11 or DHT22 sensors are commonly used with Arduino to measure air temperature and humidity. The DHT22 is a bit more accurate.
3. **Light Intensity:** A photodiode or phototransistor, such as the BH1750 light intensity sensor, can measure the amount of light the plant is receiving.
4. **Soil pH:** The pH of the soil can affect the plant's ability to absorb nutrients. A pH sensor like the DFRobot Gravity Analog pH Sensor can provide this information.
5. **Soil Temperature:** Different plants prefer different soil temperatures. A DS18B20 temperature sensor can be used for this purpose.
6. **Carbon Dioxide (CO2) Levels:** High levels of CO2 can promote plant growth. The MH-Z19 infrared CO2 sensor can be used to measure CO2 levels in the environment.
7. **Nutrient Levels:** This is a more advanced topic, but there are sensors available to monitor

DFKI 2024, Humane AI Net, Albrecht Schmidt



AL

I want to use an ESP32 to read an analog sensor (moisture in the soil), a DHT11 humidity and temperature sensor, and a digital output. Which pins should I use?



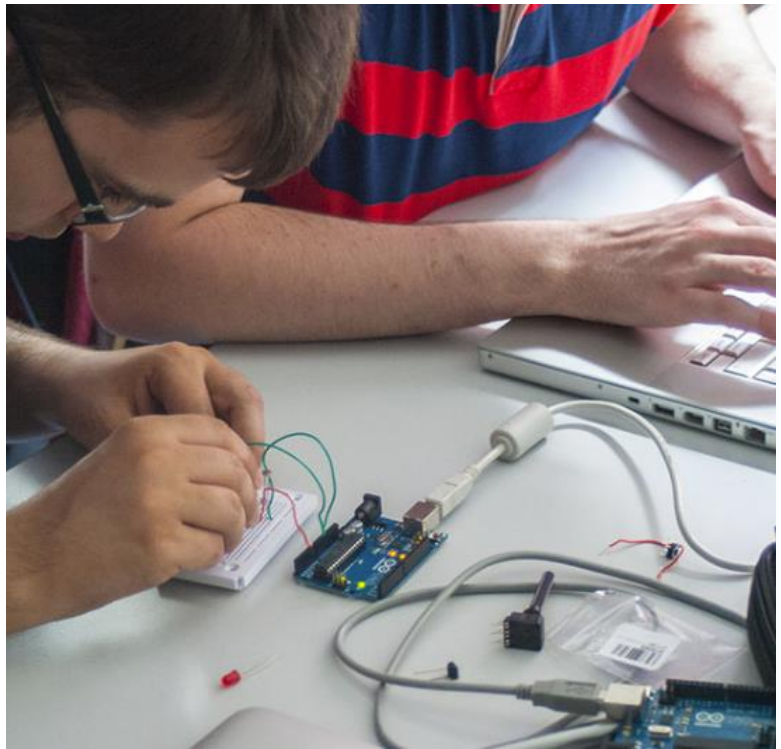
The ESP32 is a great board for this purpose as it has more than enough GPIO pins, many of which can function as analog inputs.

1. **Analog sensor (Soil Moisture):** The ESP32 board has multiple analog pins. You can use any of the ADC1 channels, which include GPIO32 to GPIO39. Note that ADC2 channels are used by the WiFi module, which can cause conflicts. For instance, you could use GPIO32 for your soil moisture sensor.
2. **DHT11 Humidity and Temperature sensor:** This is a digital sensor, so you could use any available digital pin. For instance, you could use GPIO25.
3. **Digital Output:** Again, this could be any unused digital pin. For instance, you could use GPIO26.

So, to summarize:

- Soil Moisture Sensor --> GPIO32
- DHT11 Sensor --> GPIO25
- Digital Output --> GPIO26

This is just an example. You can assign the pins as needed based on your specific requirements and the layout of your board.



```
spyCub V1.1
File Edit Tools Help
device main.py x
ed
oPy_1.1b
workSpa-
1
2 from machine import Pin
3 import adc2dispf
4
5
6 p26 = Pin(26, Pin.IN)
7
8 -if (p26.value() == 1):
9     | adc2dispf.py.main()
10
11
12
```

AL

Can you write a simple program in MicroPython using these pins that provides a wifi access point and allows http calls to read the sensors and switch the digital output?



Sure, here's a basic example of how you might set up such a program using MicroPython on the ESP32. This is a very basic example and doesn't include error handling or other elements you might want in a full program. Also, note that you need to install the `dht` module to read data from the DHT11 sensor.

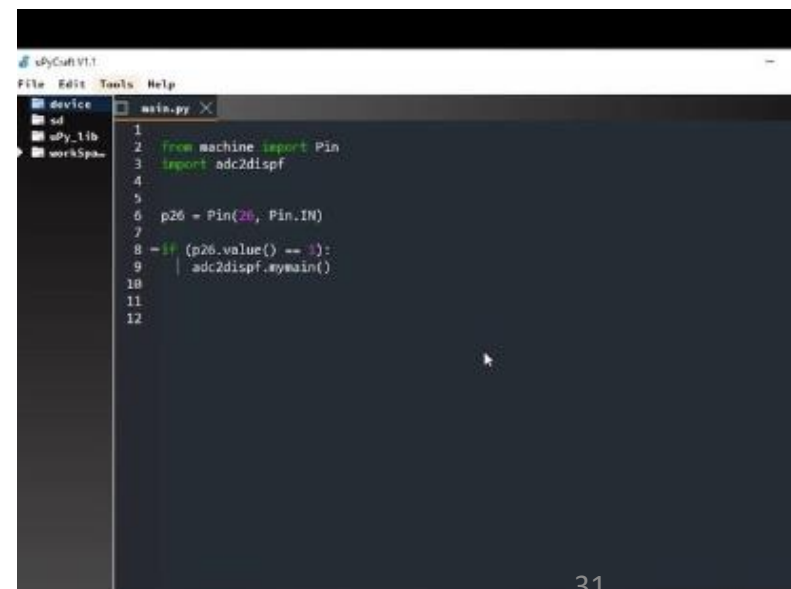
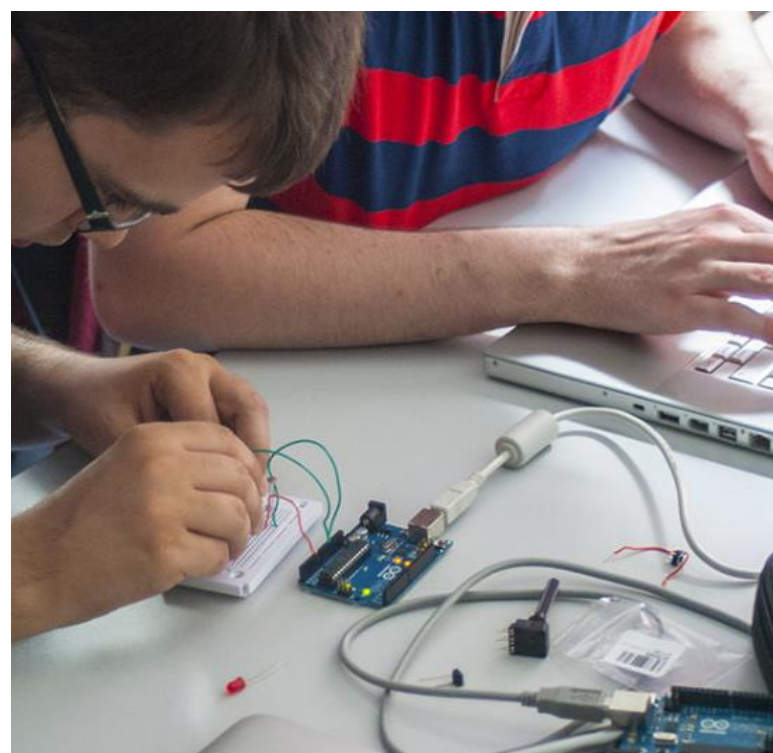
python

Copy code

```
import machine
import network
import dht
from machine import ADC, Pin
from uhttpd import HTTPServer

# Setup pins
adc = ADC(Pin(32))          # Create ADC object on ADC pin
d = dht.DHT11(Pin(25))     # Create DHT11 object on DHT pin
digital_out = Pin(26, Pin.OUT) # Create digital output on pin

# Connect to Wi-Fi
ap = network.WLAN(network.AP_IF)
ap.active(True)
ap.config(essid='MyESP32AP', password='testpassword')
```



Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

Recommendations

To replace Programmers
with Robots/LLMs Clients
will have to accurately
and completely describe
what they want!

We're Safe :-)

Outline

Prompting – Zero, One, Few Shot prompting

Prompt instruction cheat sheet

Self-Refine

Iterative creation

Chain of Thoughts

Prompting with knowledge

Why to do Human Centered Design with LLMs

Prototyping and Engineering with LLMs

Replacing developer?

Recommendations

Some Recommendation

- **It is only safe to use, where you can check the outcome!**
- Make use of GenAI and LLMs to improve HCD
- Understand, Utilize and Document the advantages
 - Do not waste the time of humans when you can do it with LLMs
- Understand and Document the shortcomings
- **Be transparent:** always make very clear where GenAI and LLMs are involved and where it is based on interaction with humans
 - do not “sell” the output of LLMs as human input
 - do not use it where you are required to give human input (e.g. prolific)

Activity Instructions

Use ChatGPT

randomly pick:

- 1 use case
- 1 output tone
- 1 output format

Your task is to generate a prompt that...

- LLM should **not** respond to questions outside scope of use case
- LLM should **not** break the tone
- LLM should only answer in specified output format
- LLM should be **resistant to prompt injection**

Activity

Use Case

- Language learning
- Sports training
- Psychological Support

Output Tone:

- Magician
- Football coach
- PhD Supervisor
- Kindergarden Teacher

Output Format: JSON document

Output Format: Tweet

Output Format: HTML document

Output Format: Markdown

Output Format: Latex

Discussion

Albrecht.Schmidt@ifi.lmu.de

<https://www.linkedin.com/in/albrechtschmidt/>

<https://twitter.com/albrechtschmidt>

References

- Albrecht Schmidt, Passant Elagroudy, Fiona Draxler, Frauke Kreuter, and Robin Welsch. 2024. *Simulating the Human in HCD with ChatGPT: Redesigning Interaction Design with AI*. *interactions* 31, 1 (January - February 2024), 24-31. <https://doi.org/10.1145/3637436>
- <https://medium.com/mlearning-ai/i-scanned-1000-prompts-so-you-dont-have-to-10-need-to-know-techniques-a77bcd074d97>
- Zhang, Zhuosheng, et al. "Automatic chain of thought prompting in large language models." *arXiv preprint arXiv:2210.03493* (2022).
- Brown, Tom, et al. "Language models are few-shot learners." *Advances in neural information processing systems* 33 (2020): 1877-1901. https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf
- Madaan, Aman, et al. "Self-refine: Iterative refinement with self-feedback." *Advances in Neural Information Processing Systems* 36 (2024). https://proceedings.neurips.cc/paper_files/paper/2023/file/91edff07232fb1b55a505a9e9f6c0ff3-Paper-Conference.pdf
- Wei, Jason, et al. "Chain-of-thought prompting elicits reasoning in large language models." *Advances in Neural Information Processing Systems* 35 (2022): 24824-24837.
- Kojima, Takeshi, et al. "Large language models are zero-shot reasoners." *Advances in neural information processing systems* 35 (2022): 22199-22213.
- Zhang, Zhuosheng, et al. "Automatic chain of thought prompting in large language models." *arXiv preprint arXiv:2210.03493* (2022).
- Liu, Jiacheng, et al. "Generated knowledge prompting for commonsense reasoning." *arXiv preprint arXiv:2110.08387* (2021). https://arxiv.org/pdf/2110.08387.pdf?trk=public_post_comment-text
- Albrecht Schmidt. 2023. *Speeding Up the Engineering of Interactive Systems with Generative AI*. In *Companion Proceedings of the 2023 ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS '23 Companion)*. Association for Computing Machinery, New York, NY, USA, 7-8. <https://doi.org/10.1145/3596454.3597176>