Learn Programming and Data Analysis with Prompt Engineering

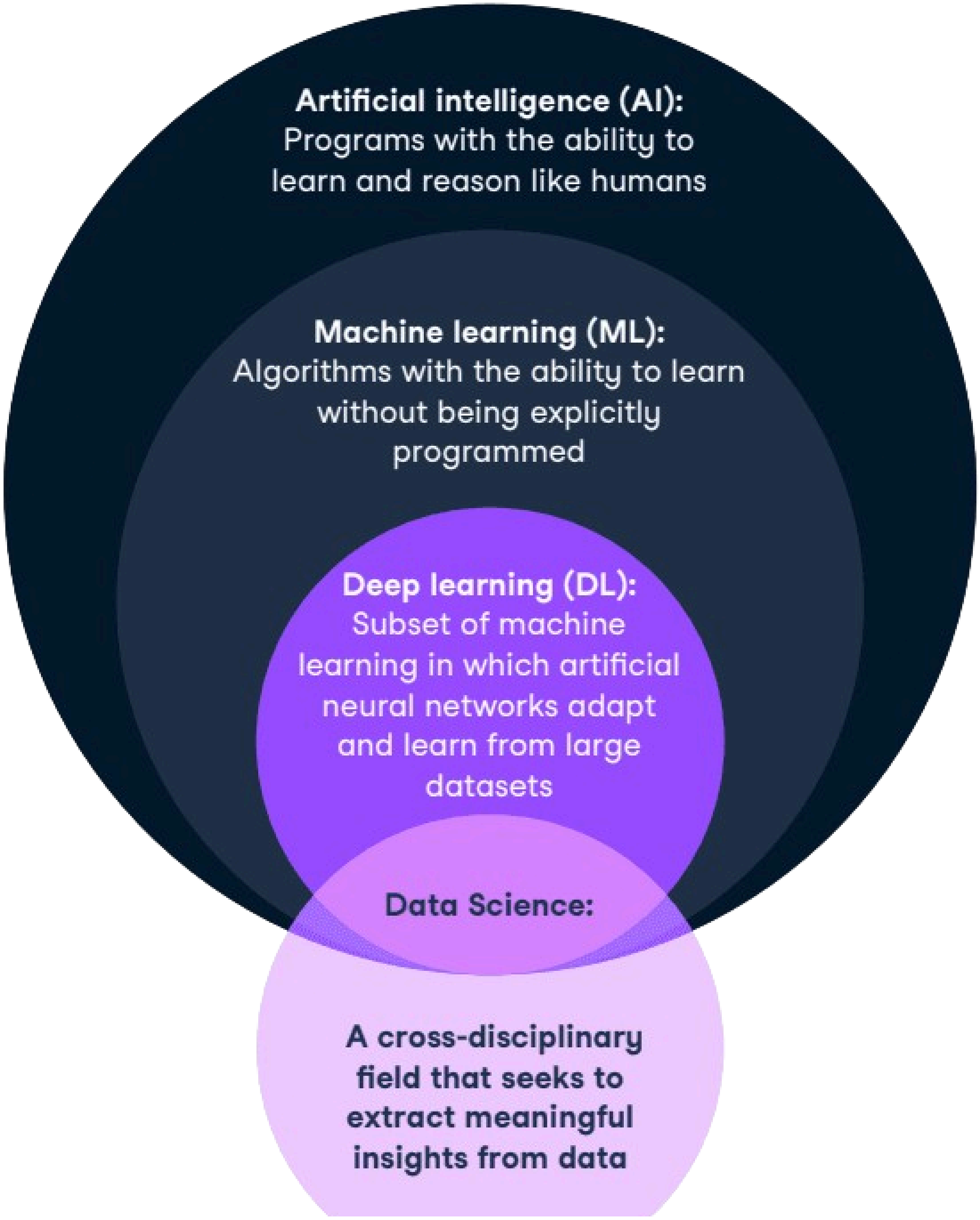
數聚點 | 郭耀仁 yaojenkuo@ntu.edu.tw | TDCC

Prompt Engineering

What is Prompt Engineering

Prompt engineering is the practice of designing and refining questions or instructions to receive specific responses from AI models, especially Large Language Models.

Relationship between data science vs. artificial intelligence vs. machine learning vs. deep learning can be illustrated with set operations



Artificial intelligence (AI):
Programs with the ability to learn and reason like humans

Machine learning (ML):
Algorithms with the ability to learn without being explicitly programmed

Deep learning (DL):
Subset of machine learning in which artificial neural networks adapt and learn from large datasets

Data Science:

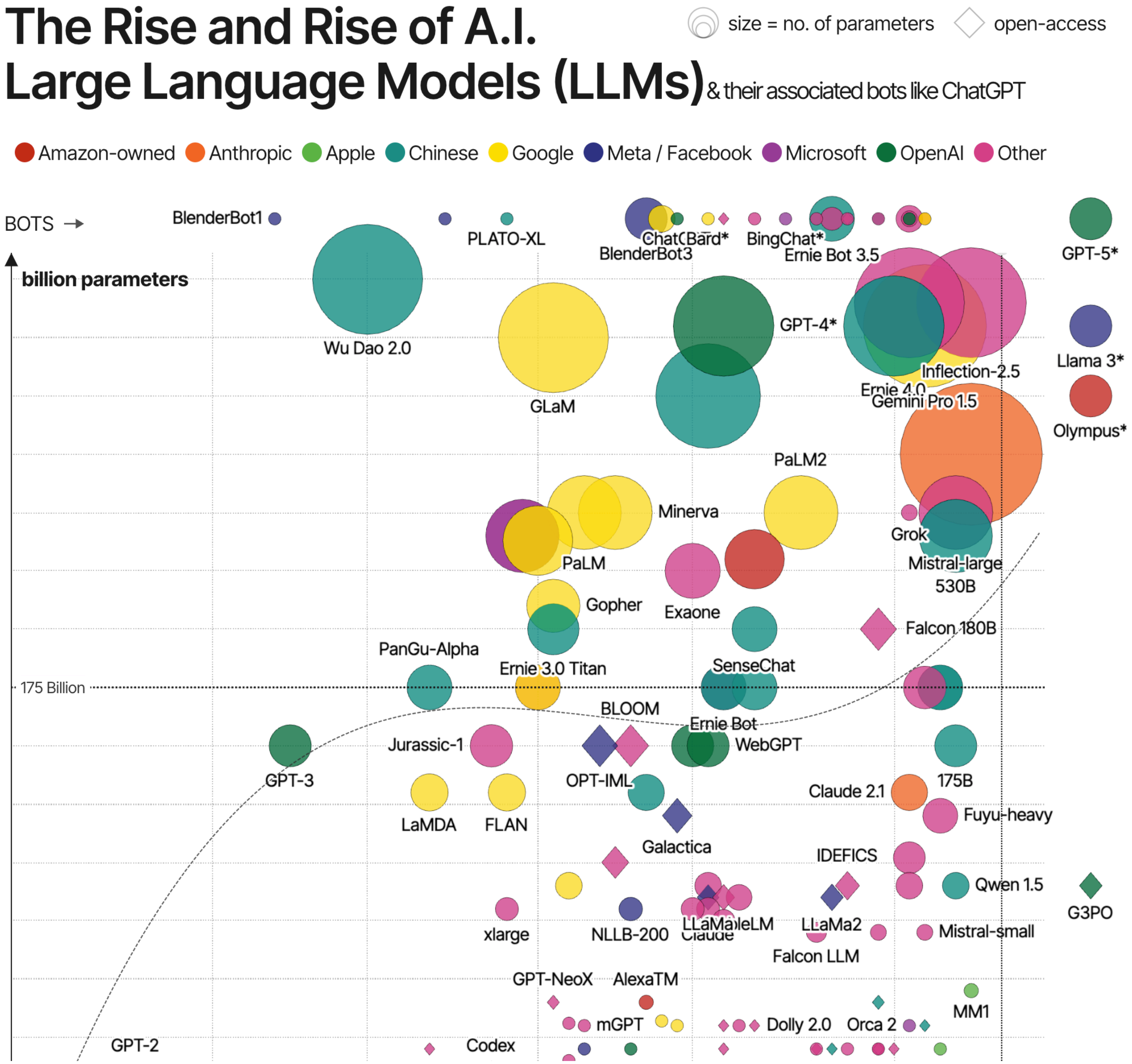
A cross-disciplinary field that seeks to extract meaningful insights from data

About Large Language Models

- LLMs are models behind popular tools including ChatGPT, Google Bard, Claude and DALL-E...etc.
- They are called "large" because these types of models are normally made millions or billions of parameters.
- The underlying technology of LLMs is called transformer, which is a type of deep learning architecture. Transformer is the T in GPT(Generative Pretrained Transformer).

Billions of parameters

The Rise and Rise of A.I. Large Language Models (LLMs) & their associated bots like ChatGPT



Source: https://informationisbeautiful.net/visualizations/the-rise-of-generative-ai-large-language-models-llms-like-chatgpt/

Generative Pre-trained Transformers

- Generative Pre-trained Transformers (GPT) are a type of deep learning model used to generate human-like text.
- Before Transformers, there are fully-connected NN, Convolutional NN, RNN, LSTM and each of them specializes a certain domain.
- Common uses include answering questions, summarizing text, translating text to other languages, generating code, stories, conversations, and other content types.

Before GPT

- Text generation was performed with other deep learning models, such as recursive neural networks (RNNs) and long short-term memory neural networks (LSTMs).
- The RNNs and LSTMs performed well for outputting single words or short phrases but could not generate realistic longer content.
- The current AI revolution for natural language only became possible with the invention of transformer models, starting with Google's BERT in 2017.

GPT-1

In 2018, OpenAI published a paper (Improving Language Understanding by Generative Pre-Training) about using natural language understanding using their GPT-1 language model. This model was a proof-of-concept and was not released publicly.

Source: https://paperswithcode.com/paper/improving-language-understanding-by

GPT-2

In 2019, OpenAI published another paper (Language Models are Unsupervised Multitask Learners) about their GPT-2 model. This time, the model was made available to the machine learning community and found some adoption for text generation tasks. GPT-2 could often generate a couple of sentences before breaking down. This was state-of-the-art in 2019.

Source: https://paperswithcode.com/paper/language-models-are-unsupervised-multitask

GPT-3

- In 2020, OpenAI published another paper (Language Models are Few-Shot Learners) about their GPT-3 model. The model had 100 times more parameters than GPT-2 and was trained on an even larger text dataset, resulting in better model performance. The model continued to be improved with various iterations known as the GPT-3.5 series, including the conversation-focused ChatGPT.
- This version took the world by storm after surprising the world with its ability to generate pages of human-like text. ChatGPT became the fastest-growing web application ever, reaching 100 million users in just two months.

Source: https://arxiv.org/abs/2005.14165v4

GPT-4

In 2023, OpenAI published GPT-4 improves on GPT-3.5 models GPT-4 scoring 40% higher than GPT-3.5 on OpenAI's internal factual performance benchmark.

Why prompt engineering matters

- OpenAI's GPT series, especially GPT-2 and GPT-3, took transformers to the next level.
- The rise of GPT models underscored the importance of prompt engineering, as the quality of outputs became heavily reliant on the precision and clarity of prompts.

Latest prompt engineering developments

- Adaptive prompting techniques: to adjust their responses based on the user's input style and preferences.
- Multimodal prompt engineering: to process and respond to prompts that include a mix of text, images, and sometimes even audio inputs.
- Integration with domain-specific models.

Key elements of a prompt

- Instruction: {summarize...}, {explain in detail...}, {elaborate with example...}.
- Context.
- Input data: given the following {text, paragraph, numbers...}.
- Output indicator: return the result in {json, csv, ...} format.

Context format

- Plain text.
- Plain text with Markdown(`.md`) format.

Source: https://markdown.tw/

The ”Learn to code” 1.0

- Official documentations.
- Books.
- Bootcamps.

The ”Learn to code” 2.0

- StackOverflow.
- Blog posts.
- Online courses.
- YouTube videos.
- Online judgers.

The ”Learn to code” 3.0

- Prompt engineering.
- GitHub Copilot.
- Cursor.

ChatGPT is so good at coding

- Addressing 728 coding problems from the LeetCode testing platform in five programming languages: C, C++, Java, JavaScript, and Python.
- It was able to produce functional code for easy, medium, and hard problems on LeetCode with success rates of about 89, 71, and 40 percent, respectively.

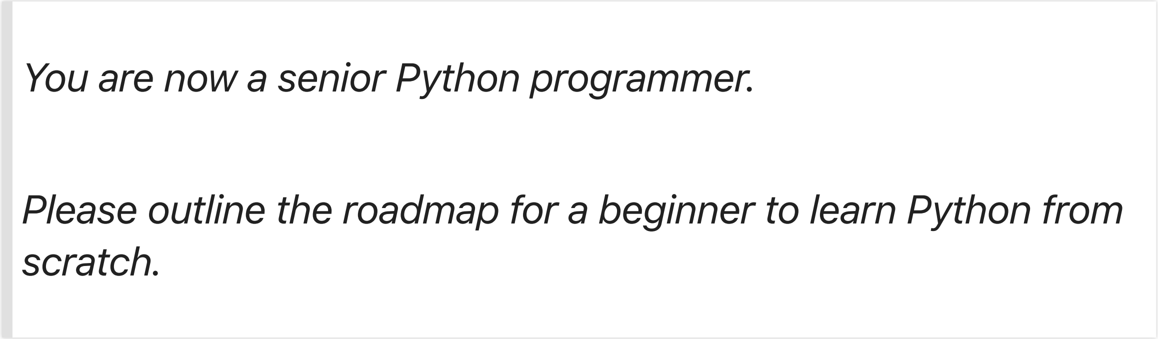
Source: https://ieeexplore.ieee.org/document/10507163

Why is ChatGPT so good at coding

- Coding is a matter of input/output mapping, which shares the same spirit with modeling.
- The Q&A community posts(e.g. StackOverflow) highlight with code chunks.
- GitHub has a wide variety of public repositories for different languages.
- LeetCode has a discussion thread for each coding challenge.

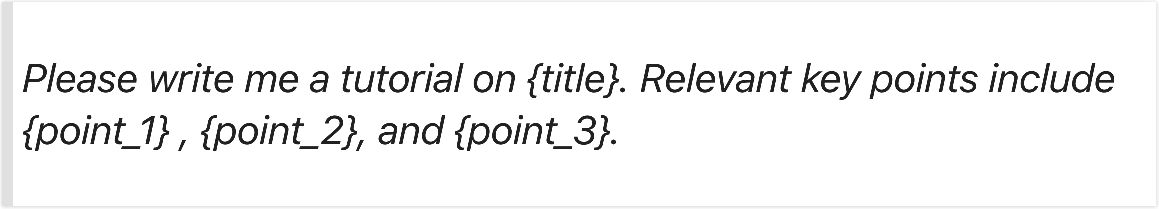
Learn Programming with Prompt Engineering

Conceptual prompts



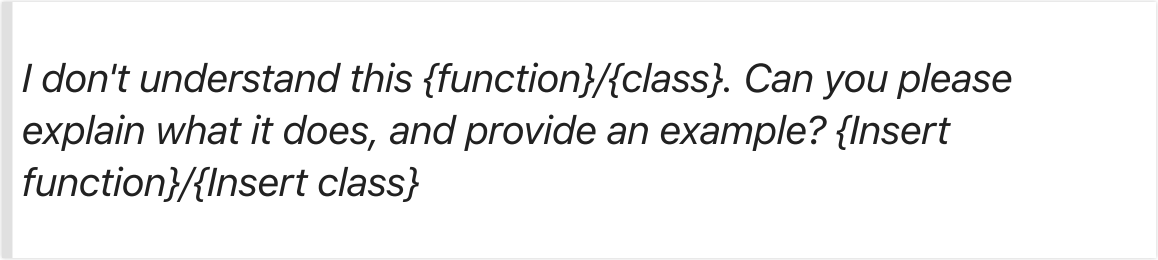
> *You are now a senior Python programmer.*
>
> *Please outline the roadmap for a beginner to learn Python from scratch.*

Conceptual prompts(cont’d)



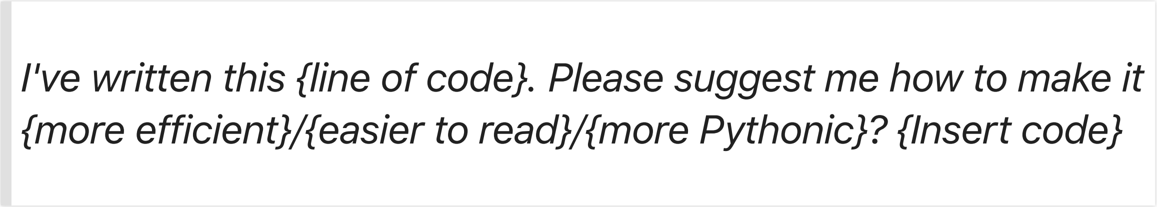
> *Please write me a tutorial on {title}. Relevant key points include {point_1} , {point_2}, and {point_3}.*

Code explanation



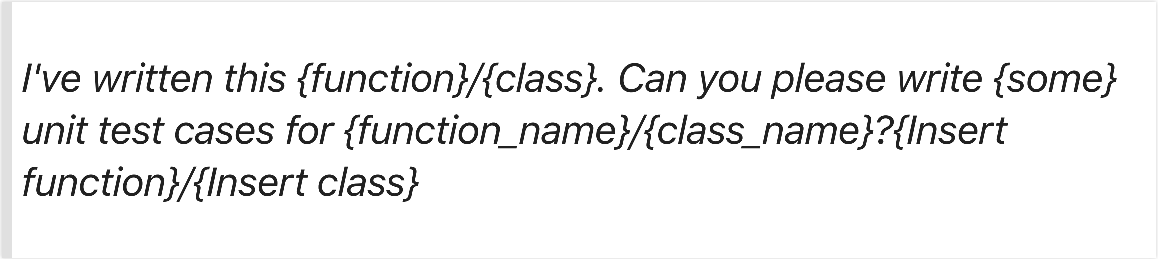
> *I don't understand this {function}/{class}. Can you please explain what it does, and provide an example? {Insert function}/{Insert class}*

Code suggestion



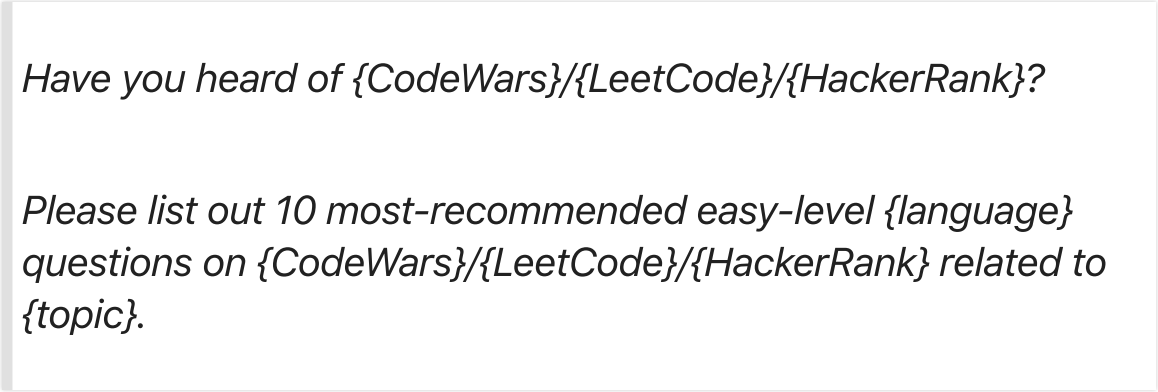
> *I've written this {line of code}. Please suggest me how to make it {more efficient}/{easier to read}/{more Pythonic}? {Insert code}*

Write unit tests



> *I've written this {function}/{class}. Can you please write {some} unit test cases for {function_name}/{class_name}?{Insert function}/{Insert class}*

Question recommendations



Have you heard of {CodeWars}/{LeetCode}/{HackerRank}?

Please list out 10 most-recommended easy-level {language} questions on {CodeWars}/{LeetCode}/{HackerRank} related to {topic}.

Use ChatGPT as lecturer's assistant

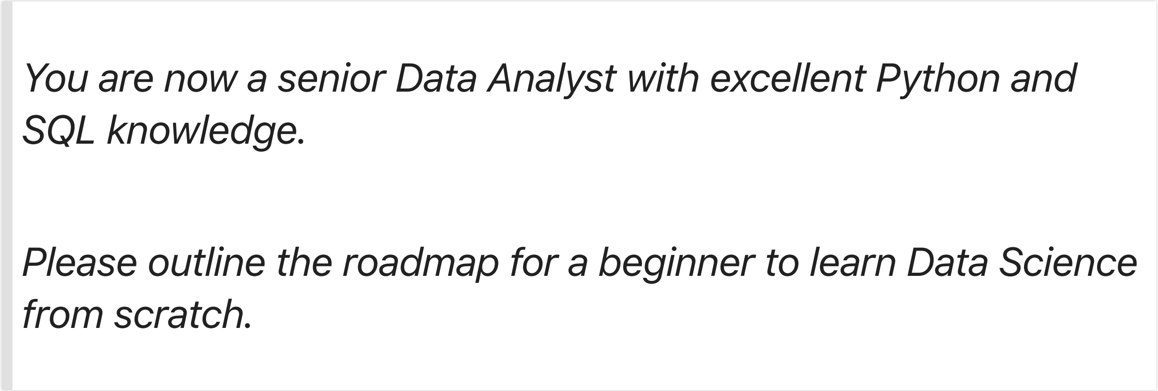
Take Python 的 50+ 練習：資料科學學習手冊 as an example.

Use ChatGPT as student’s assistant

Take Python 的 50+ 練習：資料科學學習手冊 as an example.

Learn Data Analysis with Prompt Engineering

Conceptual prompts



> *You are now a senior Data Analyst with excellent Python and SQL knowledge.*
>
> *Please outline the roadmap for a beginner to learn Data Science from scratch.*

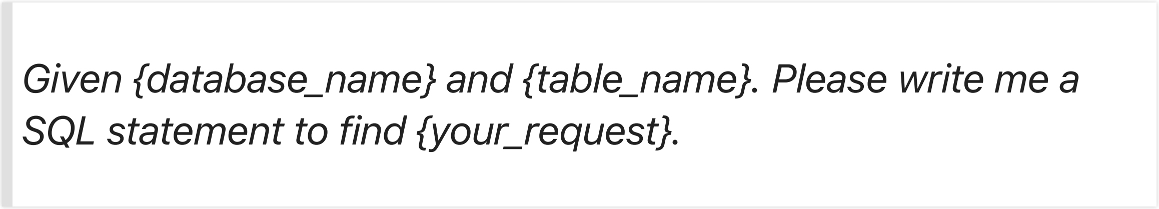
Conceptual prompts(cont'd)

> *Please write me a tutorial on {title}. Relevant key points include {point_1} , {point_2}, and {point_3}.*

Create a table with natural language

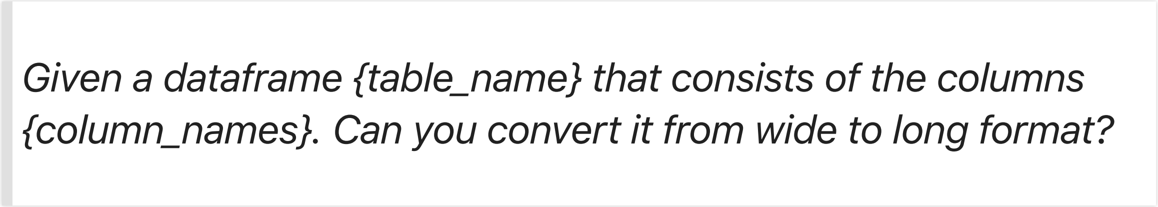
> *Please write me a SQL statement in {dbms_name} that create a table {table name} with the columns {column_name} and data types {data_type}. Include relevant constraints.*

Query with natural language



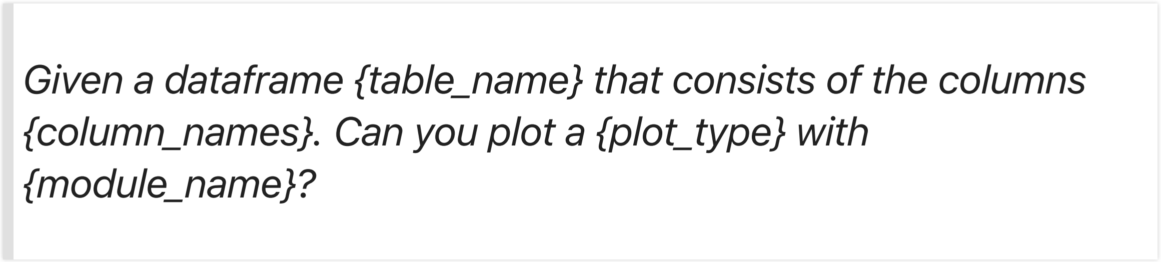
> *Given {database_name} and {table_name}. Please write me a SQL statement to find {your_request}.*

Data wrangling with Python



> *Given a dataframe {table_name} that consists of the columns {column_names}. Can you convert it from wide to long format?*

Data visualization with Python



Given a dataframe {table_name} that consists of the columns {column_names}. Can you plot a {plot_type} with {module_name}?

Use ChatGPT as lecturer’s assistant

- Take SQL的五十道練習：初學者友善的資料庫入門 as an example.
- Take Python 的 50+ 練習：資料科學學習手冊 as an example.

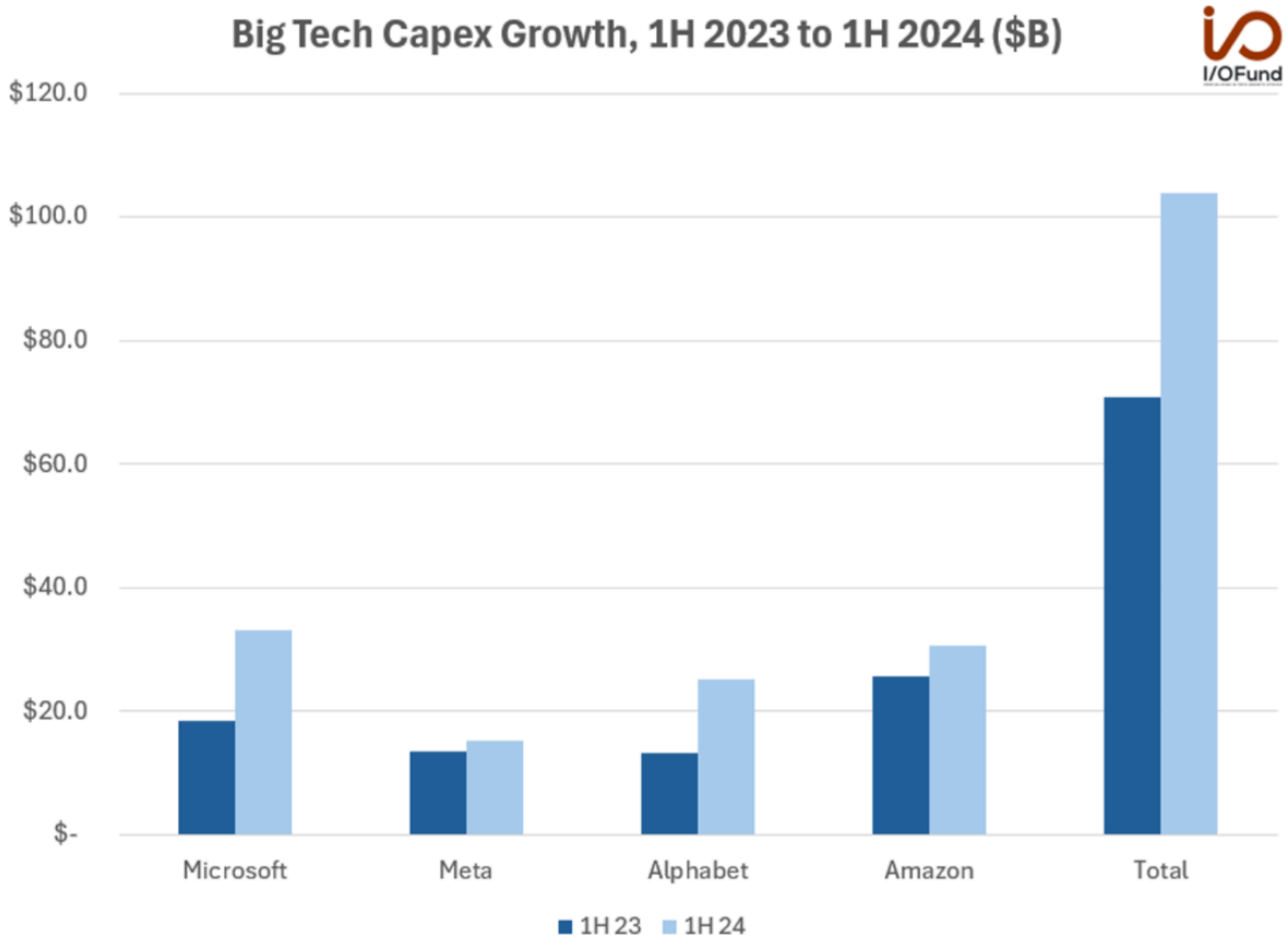
Use ChatGPT as student's assistant

- Take SQL的五十道練習：初學者友善的資料庫入門 as an example.
- Take Python 的 50+ 練習：資料科學學習手冊 as an example.

Surviving in AI world

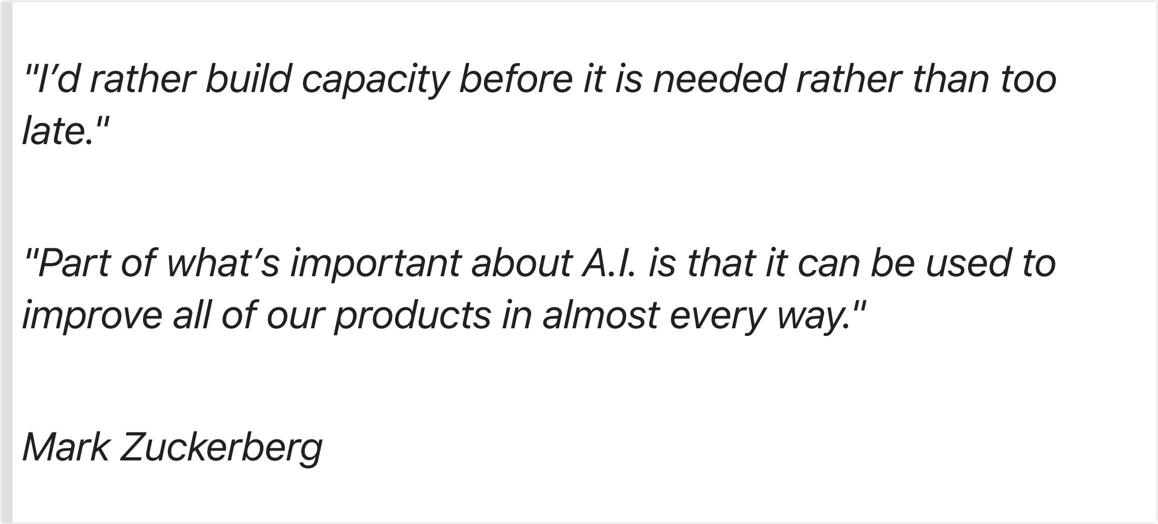
Big Tech Battles On AI

In the first half of 2024, Alphabet, Amazon, Meta and Microsoft spent nearly $104 billion in capex, up 47% YoY, with more than half of that total coming in Q2.



Source: https://www.forbes.com/sites/bethkindig/2024/08/08/microsoft-leads-big-tech-in-ai-monetization-amazon-a-close-second

Big Tech Battles On AI(Cont’d)



> *"I’d rather build capacity before it is needed rather than too late."*
>
> *"Part of what’s important about A.I. is that it can be used to improve all of our products in almost every way."*
>
> *Mark Zuckerberg*

Source: https://www.nytimes.com/2024/07/31/technology/meta-earnings-artificial-intelligence.html

55% of organizations are now experimenting generative AI technologies

Gartner, Inc. poll of more than 1,400 executive leaders, 45% reported that they are in piloting mode with generative AI, and another 10% have put generative AI solutions into production.

Source: https://www.gartner.com/en/newsroom/press-releases/2023-10-03-gartner-poll-finds-55-percent-of-organizations-are-in-piloting-or-production-mode-with-generative-ai

63% of data specialists are using generative AI technologies

The majority (63%) of data science practitioners said they're using generative AI the same amount or more now compared to 2022. Many feel their jobs are threatened by generative AI.

Source: https://www.anaconda.com/lp/state-of-data-science-report-2023

85% of enterprises will expand AI with open source models

Many are turning to open source models such as GPT-J, BERT, and FLAN-T5. The release of portable model families such as Llama 2, marketplaces such as Hugging Face, and corporate investments into these initiatives will further accelerate this trend.

Source: https://www.anaconda.com/lp/state-of-data-science-report-2023

”Boring AI” use cases generate the majority of the value

Knowledge workers utilizing GPT-4 experienced significant enhancements in efficiency and quality, particularly in domains like operations, customer service, legal and compliance, and technology.

Source: https://www.bcg.com/publications/2023/assessing-the-impact-of-generative-ai-on-workforce-productivity

The rise of small domain-specific LLMs

- ChatLaw(Law)
- Med-PaLM 2(Medicine, Google)
- BloombergGPT(Finance, Bloomberg)
- Replit Code(Coding, Replit)

Source: https://www.bcg.com/publications/2023/assessing-the-impact-of-generative-ai-on-workforce-productivity

The risks of AI bubble

Wall Street investment banks including Goldman Sachs and Barclays, as well as VCs such as Sequoia Capital, have issued reports raising concerns about the sustainability of the AI gold rush, arguing that the technology might not be able to make the kind of money to justify the billions being invested into it.

Source: https://www.washingtonpost.com/technology/2024/07/24/ai-bubble-big-tech-stocks-goldman-sachs

60% of employees will get prompt engineering training

Prompt engineering will become a valuable asset across many roles, but not a career path.

Source: https://www.forrester.com/blogs/predictions-2024-data-and-analytics

AI will focus data professions on value creation

- Problem solving.
- Project management.
- Autonomy and collaboration.
- Communication and storytelling.

Source: https://www.anaconda.com/lp/state-of-data-science-report-2023

How coders can survive

- Stick to basics and best practices.
- Find the tool that fits.
- Chain-of-thought prompting.
- Be critical.

Stick to basics and best practices

- A lot more to software engineering than just generating code, from eliciting user requirements to debugging, testing, and more.
- Analyzing a problem and finding elegant solutions.
- That is algorithms, data structures, design pattern, and domain knowhow regarding programming and data analysis.

Find the tool that fits

- To incorporate each tool into development workflow: the creation of unit tests, generating test data, or writing documentation...etc.
- An open mind for new tools, don't settle on just one tool.

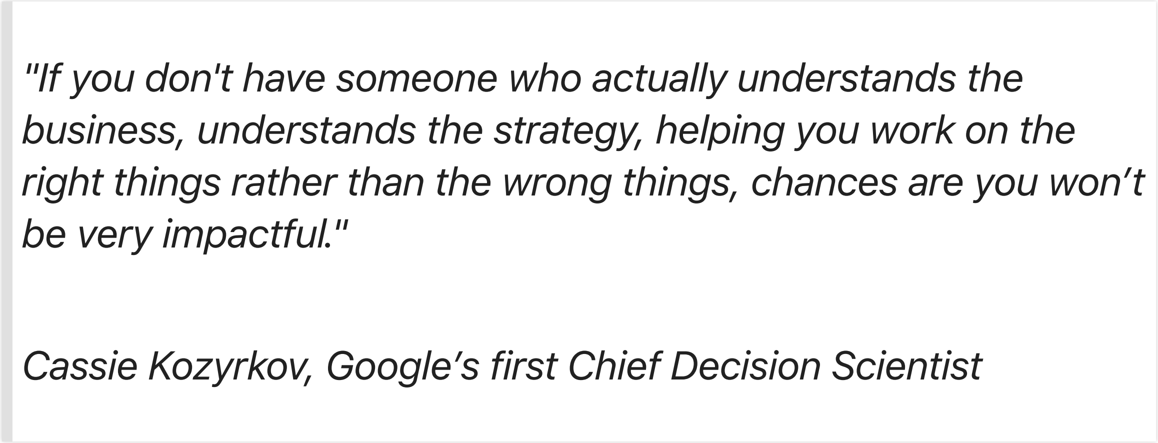
Chain-of-thought prompting

- To incorporate a divide-and-conquer strategy.
- That is to break down a problem into multiple steps and tackle each one to solve the entire problem.
- Instead of asking ChatGPT to write an entire program from scratch, divide those tasks and ask it to write specific functions for each.

Be critical

- Be critical of the outputs, as they tend to hallucinate and produce inaccurate or incorrect code.
- Privacy, copywright, and security are the underlining risks.

Developers will survive



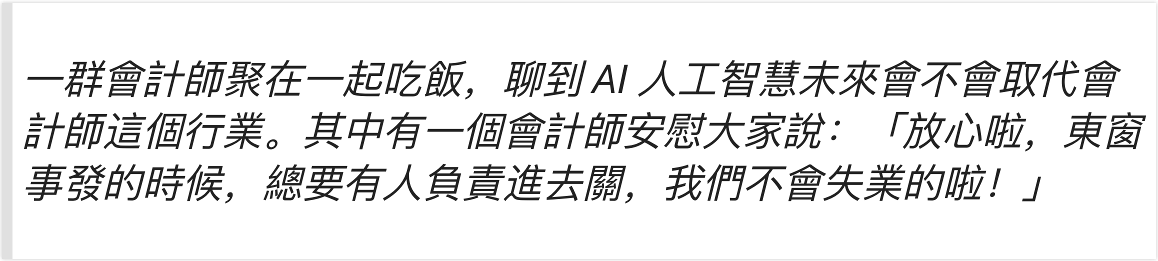
> *"If you don't have someone who actually understands the business, understands the strategy, helping you work on the right things rather than the wrong things, chances are you won’t be very impactful."*
>
> *Cassie Kozyrkov, Google’s first Chief Decision Scientist*

Coders will survive

> *我也很愛吃麵包，以前也犯了一樣的錯，可是我認識麵包店叔叔後，才瞭解自己做的麵包，遠比偷來的還要美味。*
>
> *麵包小偷2：誰偷了葡萄乾麵包*

Source: https://www.books.com.tw/products/0010917146

And so will others



> *一群會計師聚在一起吃飯，聊到 AI 人工智慧未來會不會取代會計師這個行業。其中有一個會計師安慰大家說：「放心啦，東窗事發的時候，總要有人負責進去關，我們不會失業的啦！」*

Source: Google Search {會計} {AI} {笑話}