IOWA STATE UNIVERSITY
Apparel, Events, and Hospitality Management

INNOVATE at Iowa State

Leveraging Large Language Models for Business Analytics

Chunsheng (Jerry) Jin

Outline of this class

- Review of previous class
- Understanding Large Language Models (LLMs)
- Introduction to GPT API
- Hands-on GPT API practice

Review of previous class

Naïve Bayes

- Spam email detection
  - Good performance
  - 95% accuracy

- Sentiment analysis on Twitter data
  - Poor performance
  - 76% accuracy

Review of previous class

Limitations of Naïve Bayes

- Strong assumption of independence between words
  - Ignore grammar and order
  - Do not distinguish same words with different meanings (free beer vs. free speech)

Review of previous class

Limitations of Naïve Bayes

- Strong assumption of independence between words
  - Ignore grammar and order
  - Do not distinguish same words with different meanings (free beer vs. free speech)
- **How to solve this issue?**
  - **Large Language Models (LLMs)**

Class activity 1

https://app.sli.do/event/2wy9SoazGcMhzSCuTyzxpe

Understanding LLMs

Machine Learning (ML)

- Naïve Bayes
- Linear regression
- Logistic regression
- Clustering analysis

Deep Learning (DL)

- Artificial neural networks (ANNs)

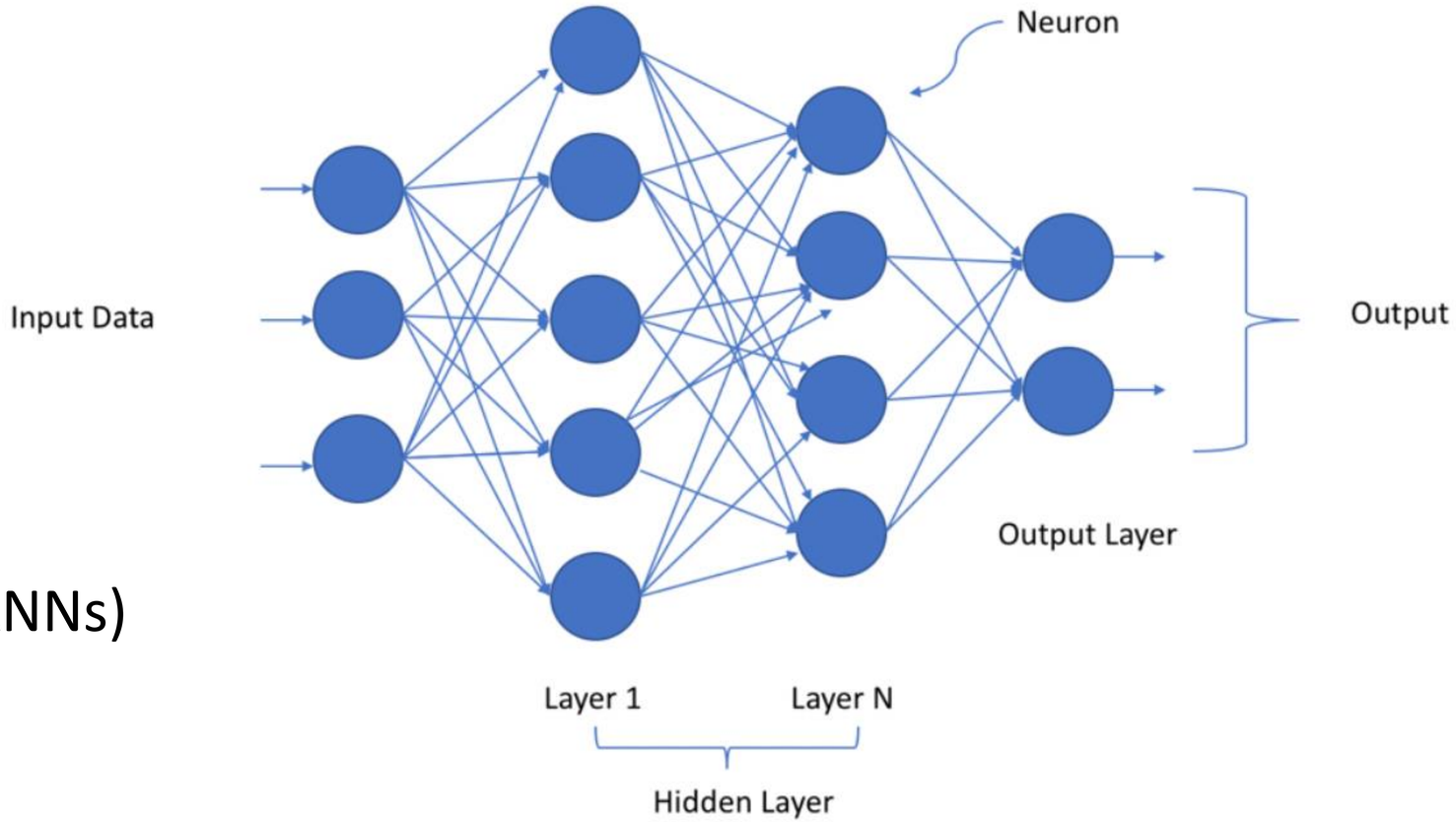


Image source: https://towardsdatascience.com/a-laymans-guide-to-deep-neural-networks-ddcea24847fb

Understanding LLMs

Machine Learning (ML)

- Naïve Bayes
- Linear regression
- Logistic regression
- Clustering analysis

Deep Learning (DL)

- Artificial neural networks (ANNs)
  - Deep neural networks (DNNs)
    - Convolutional neural networks (CNNs)
    - Recurrent neural networks (RNNs)
    - Transformer
      - LLMs (e.g., GPT, Claude)

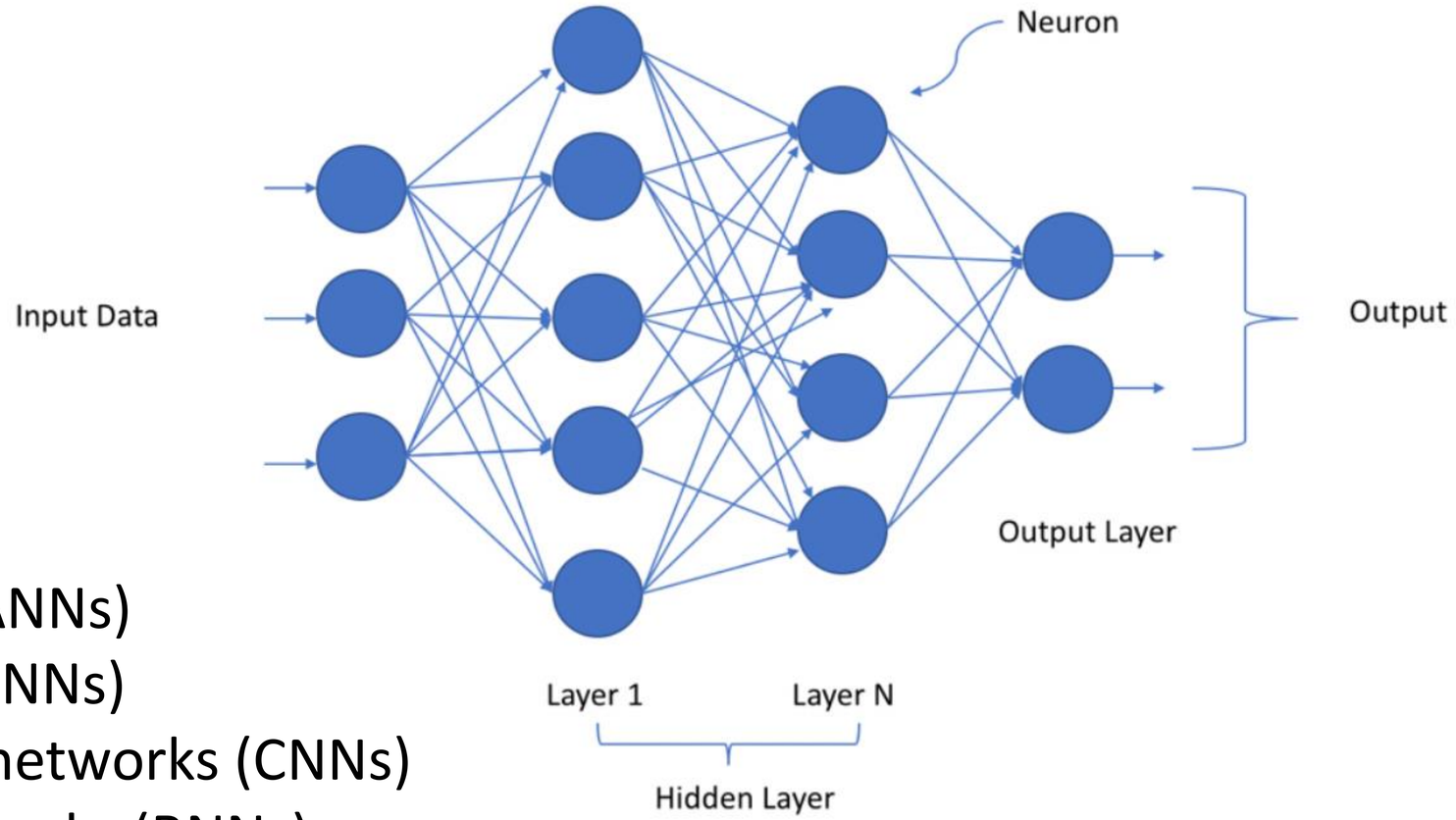
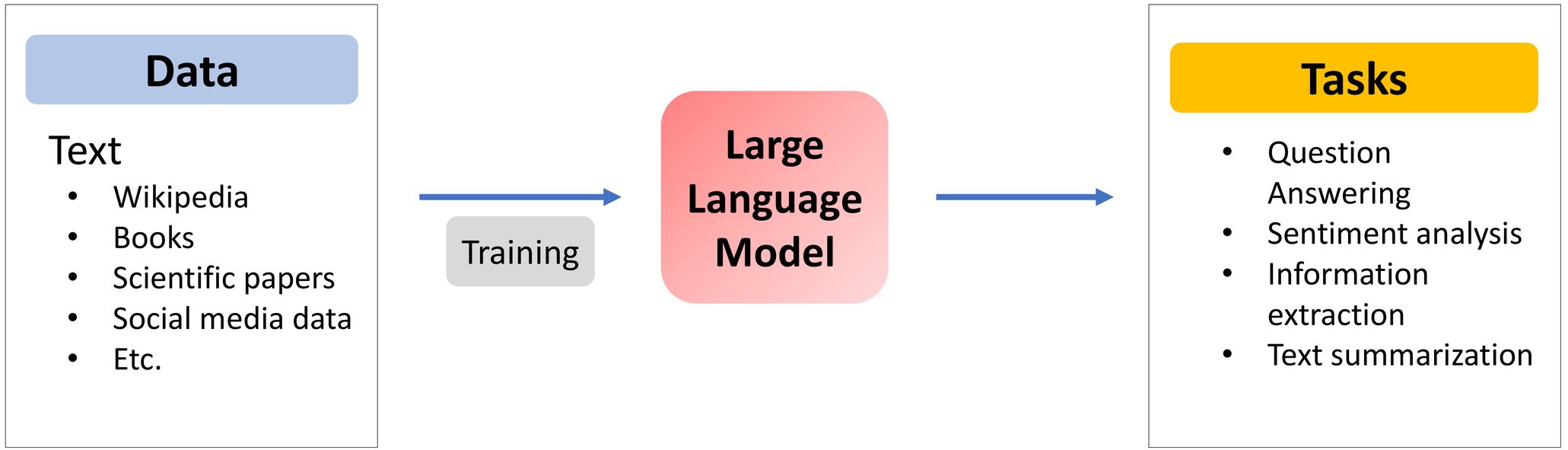
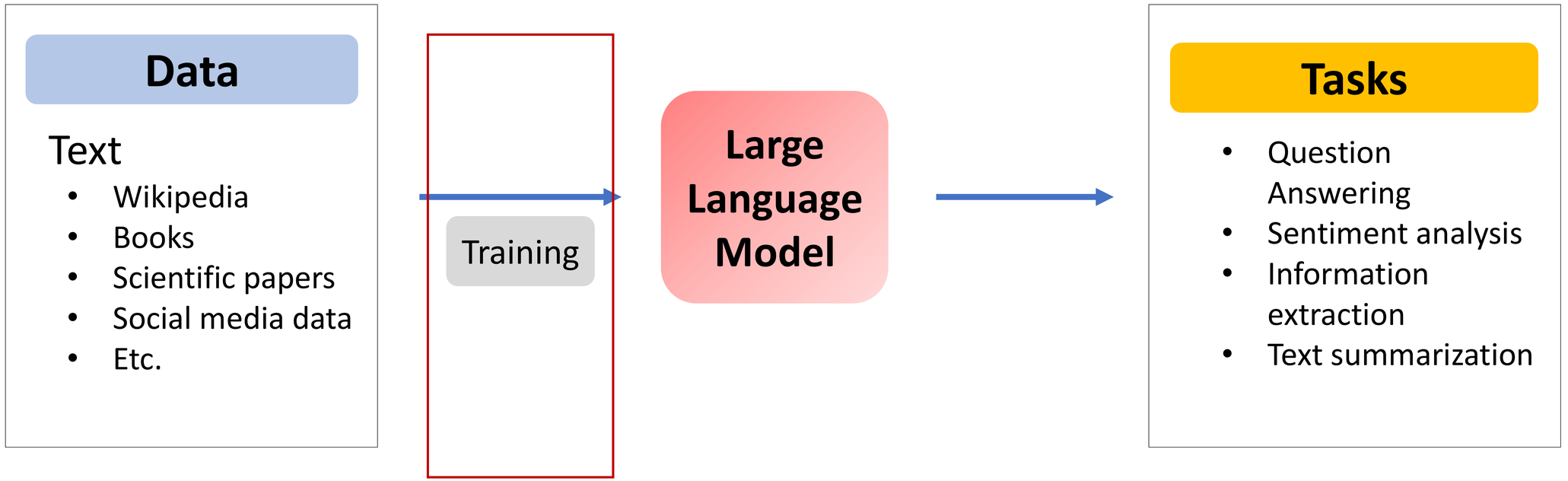


Image source: https://towardsdatascience.com/a-laymans-guide-to-deep-neural-networks-ddcea24847fb

How do LLMs work?



How do LLMs work?



How do LLMs work?

Training : Self-Supervised Learning

- Causal language modeling
  - Automatically predict the next word based on the previous words
    - "I am happy because I am studying at CSU Chico."
      - Input: "I" → Output: "am"
      - Input: "I am" → Output: "happy"
      - Input: "I am happy" → Output: "because"
      - Input: "I am happy because" → Output: "I"
      - Input: "I am happy because I" → Output: "am"
      - Input: "I am happy because I am" → Output: "studying"
      - Input: "I am happy because I am studying" → Output: "at"
      - Input: "I am happy because I am studying at" → Output: "CSU"
      - Input: "I am happy because I am studying at CSU" → Output: "Chico"

Large Language Models (LLMs)

- Leading LLMs
  - OpenAI: GPT
  - Anthropic: Claude
  - Google : Gemini

- Perform NLP tasks
  - Question answering
  - Machine translation
  - Text summarization
  - Programming code generation

Current Capabilities of LLMs

- OpenAI introduced Operator on January 23, 2025

GPT API

GPT (Generative Pre-trained Transformer)

- **Generate** natural language
- **Pre-trained** on big data
- **Transformer** architecture
- GPT models: GPT4o and GPT4o mini
- ChatGPT is the Chatbot developed based on the GPT models.

GPT API

API (Application Programming Interface)

- Allow different software applications to communicate with each other

Source: https://www.geeksforgeeks.org/what-is-an-api/

GPT API

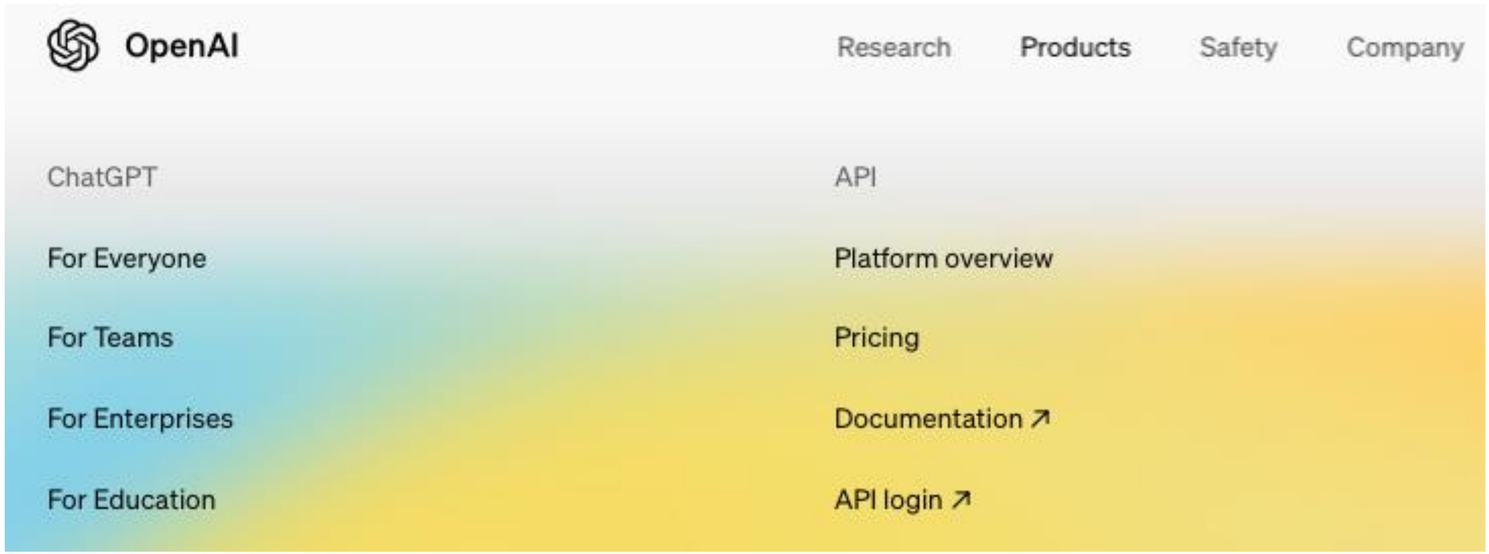
- Provide access to GPT models to perform various NLP tasks

Advantages of GPT API over ChatGPT

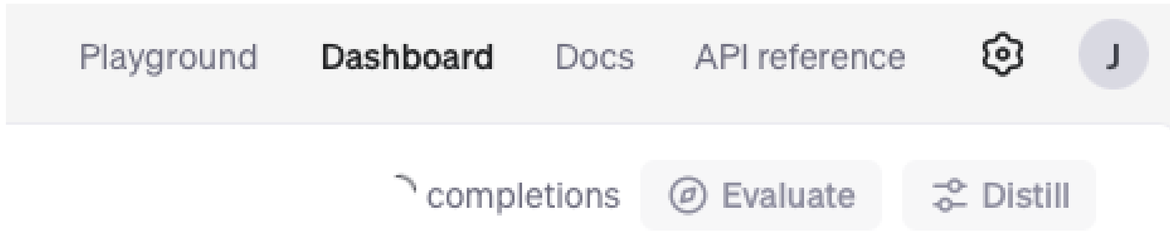
- Scalability: Request large volumes of requests
  - OpenAI o1: 50 messages a week (as of December 2024)
- Consistent Performance:
  - e.g., set parameter of temperature=0

Get GPT API Access

- Go to the OpenAI website and sign up for an account.

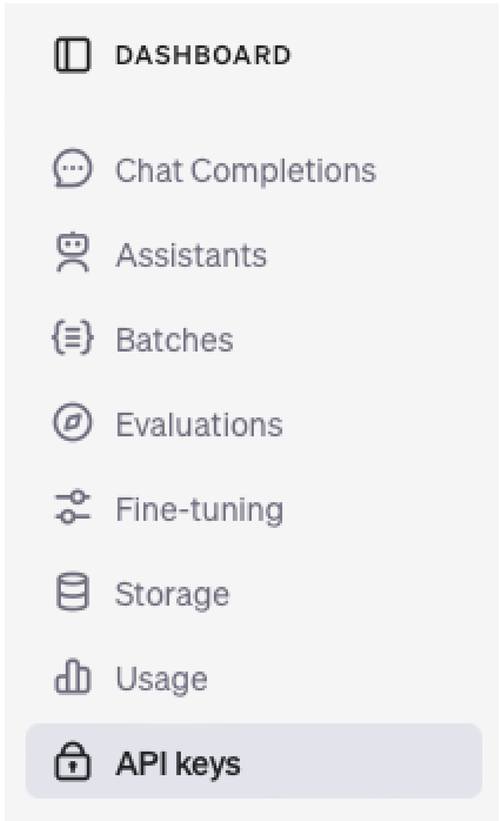


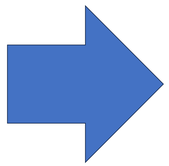
- Go to Dashboard on the top right corner.

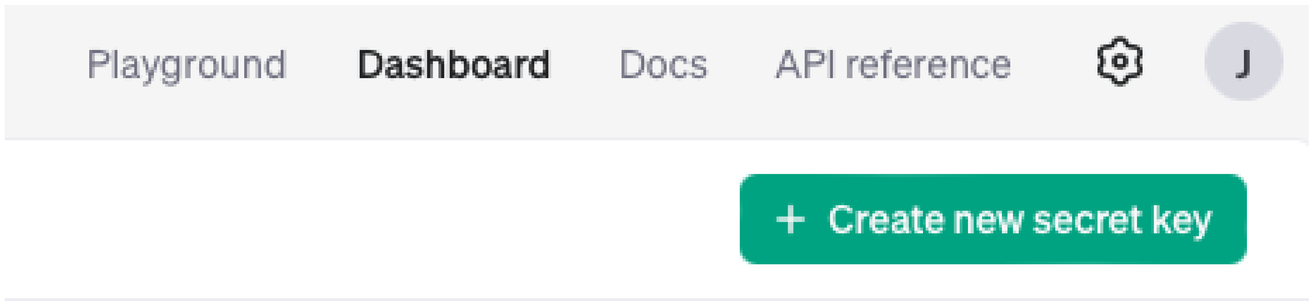


Get GPT API Access

- Click “API keys” on the left column => Click “Create new secret key”

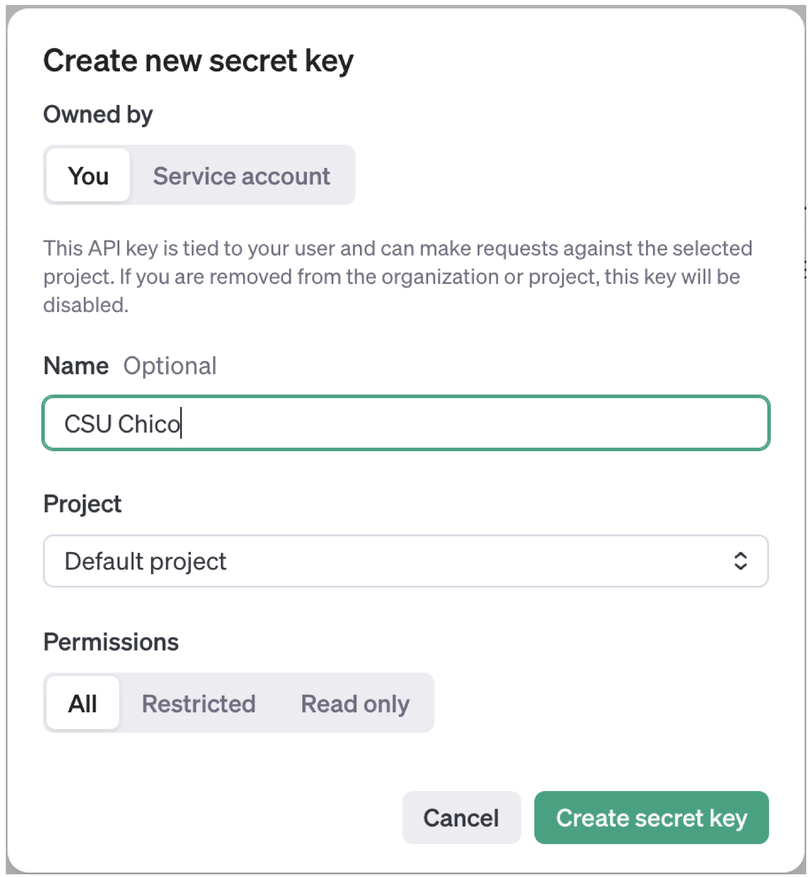


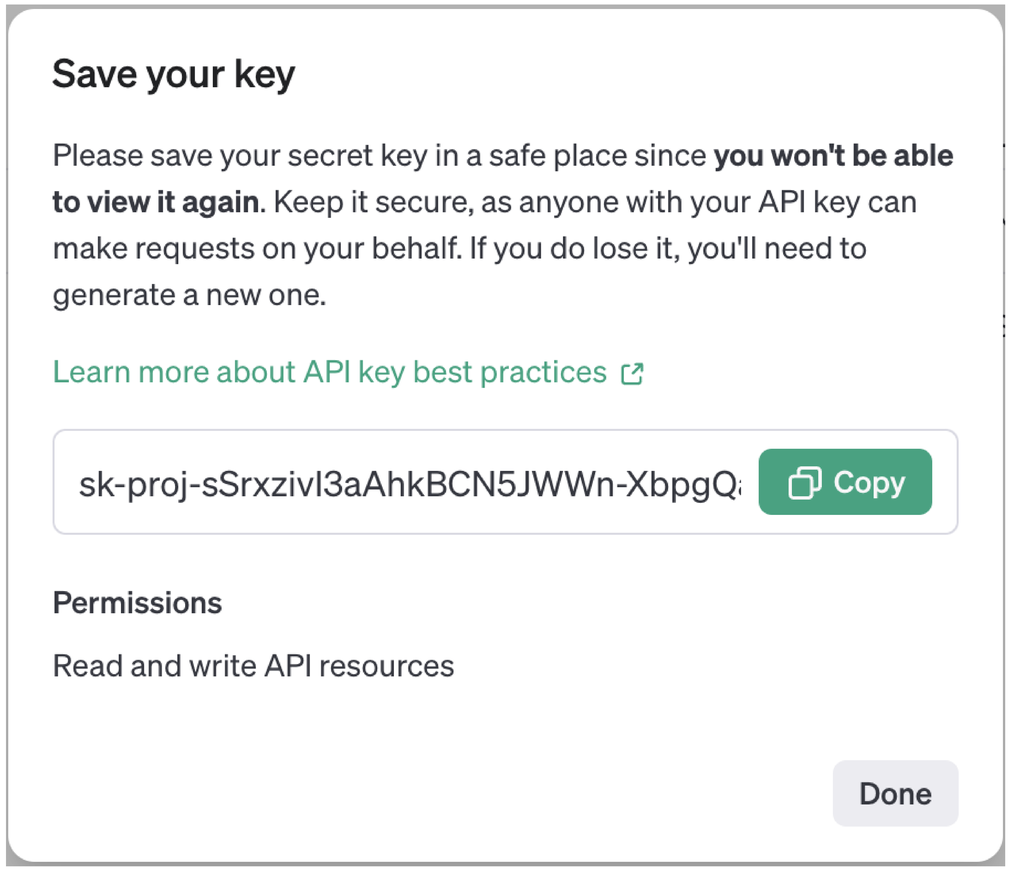




Get GPT API Access

- Create an API key in a pop-up window for secret key.





Python code and datasets

https://github.com/chunshengj/Chico_Demo

Jupyter Notebook on Anaconda

https://chunshengj.github.io/579-class/data_analysis/Install_Anaconda_Jupyter_Package.html

Alternative: Google Colab (cloud-based Jupyter Notebook)

https://drive.google.com/file/d/1YeFIxjdyAxUvm2s72rfJ2OogfN_7Hojx/view?usp=sharing

Group discussion

Two students are paired to discuss the question below and share your discussion with the class:

- Can we trust the results generated by LLMs? Why or why not?

Takeaways

- Learned what LLMs are and how they work
- Introduced GPT API
- Used GPT API to perform two tasks

Model Evaluation

- Confusion Matrix
  - A table used to describe the performance of a classification model

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Model Evaluation

- Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall or Sensitivity} = \frac{TP}{TP + FN}$$

$$\textbf{F1} = 2\,\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Model Evaluation

- Confusion Matrix

	Yes	No
Yes	5	1
No	0	4

$$Accuracy = \frac{correctly\ predicted\ instances}{total\ number\ of\ instances} = \frac{9}{10} = 90\%$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall or Sensitivity} = \frac{TP}{TP + FN}$$

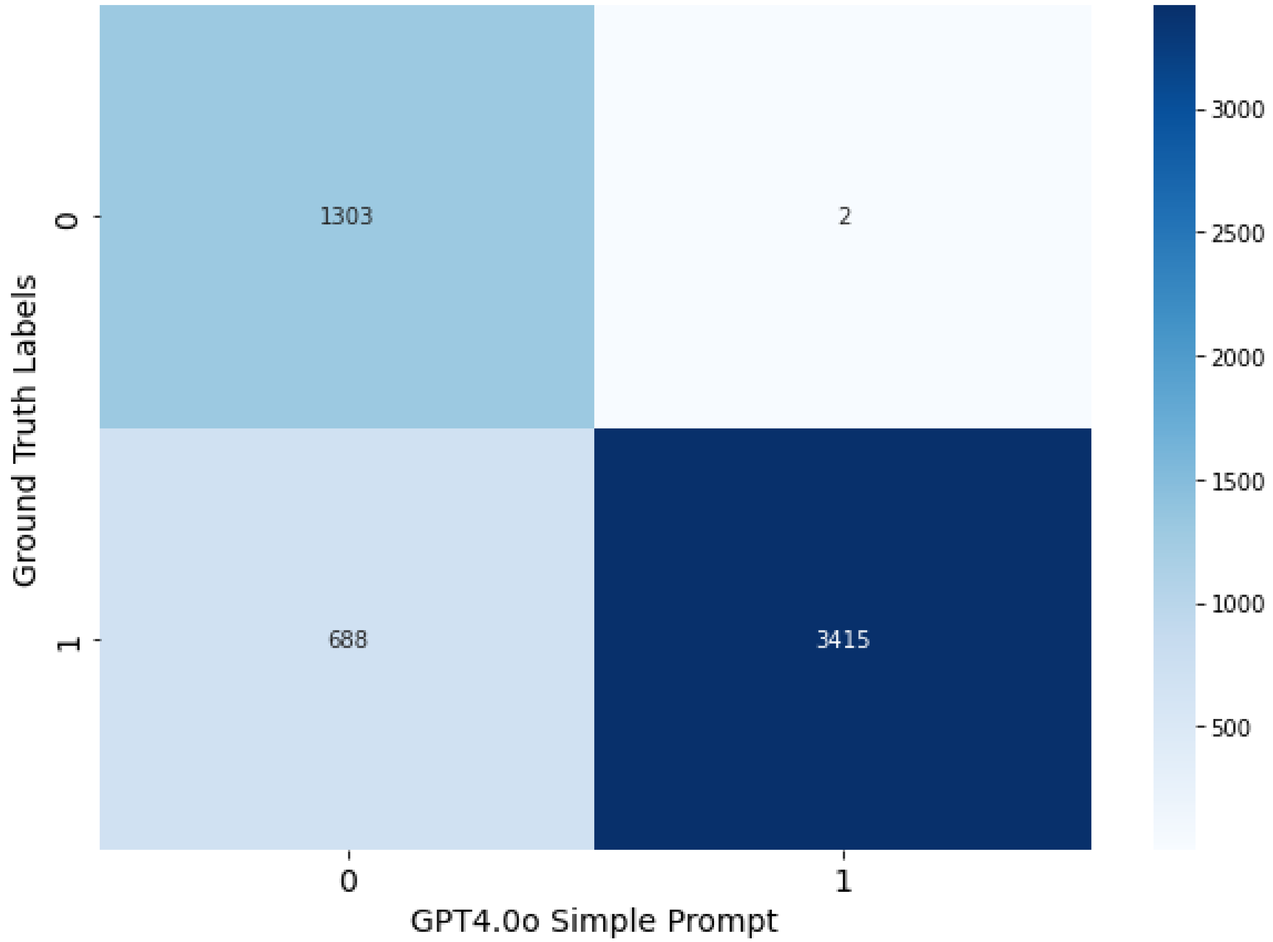
$$\textbf{F1} = 2\,\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Model Evaluation

- Ground truth
  - True classification values used to compare with GPT-generated results for model evaluation
- In the example, our true sentiment values are the star ratings by reviewers.
  - 1 & 2 stars => negative
  - 4 & 5 stars => positive

Model Evaluation

- Draw a Confusion Matrix



Model Evaluation

Sentiment analysis using standard prompt

```
prompt = f"""
What is the sentiment of the following hotel review, which is delimited with triple backticks?

Give your answer as a single word, either "positive" or "negative".

Review text: ```{review}```
"""
```

Models	Precision	Recall	F1	Accuracy
GPT-4o	0.8263	0.9149	0.8487	**0.8717**

Takeaways

- Learned what LLMs are and how they work
- Introduced GPT API
- Used GPT API to perform two tasks
- Learned model evaluation

How can we improve model performance?

Improving GenAI Performance

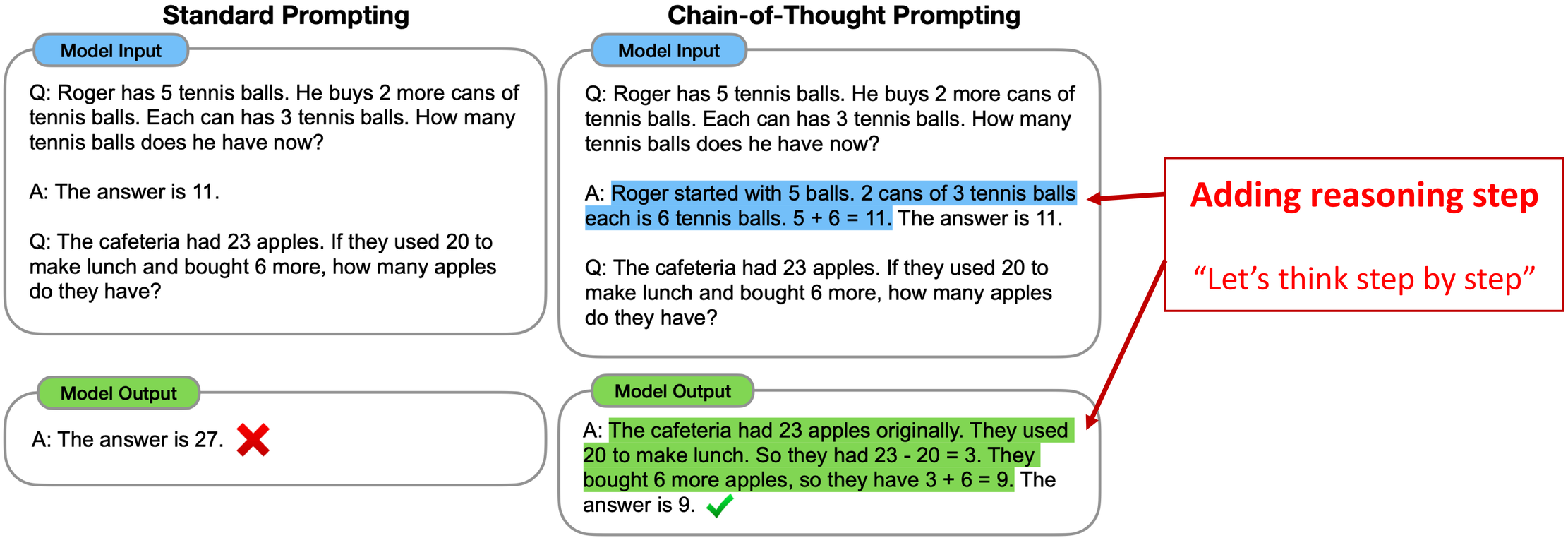
Prompt engineering techniques

- Chain of Thought (CoT)
- Self-Reflection (SR)

Improving GenAI Performance

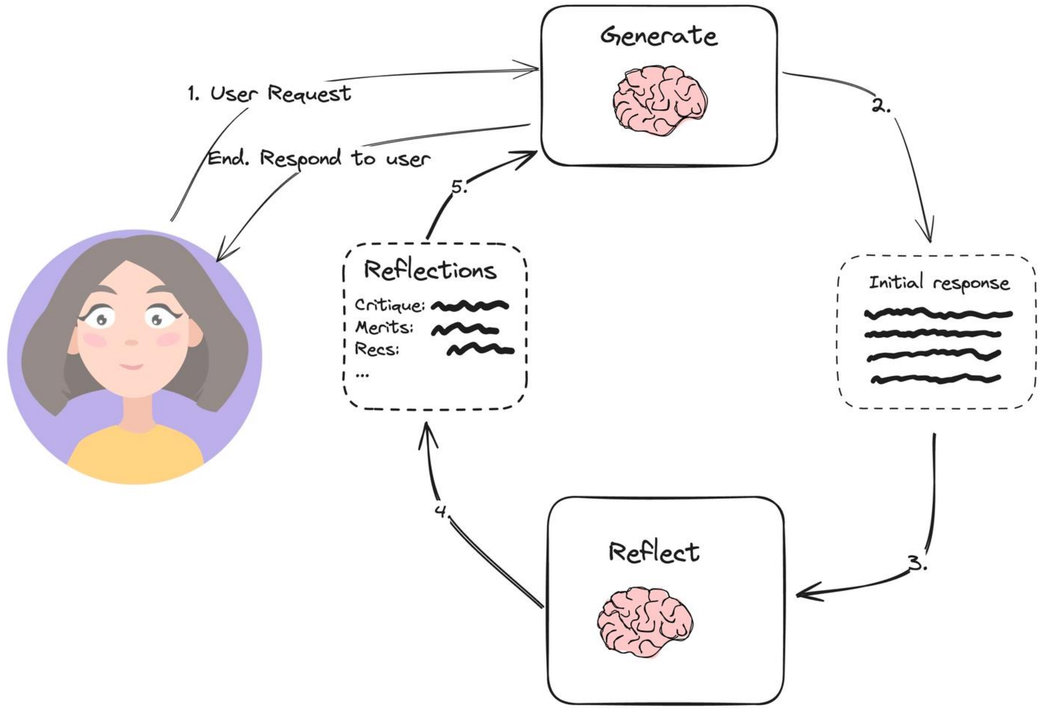
Chain of Thought (CoT)

- Involve a sequence of intermediate reasoning steps



Improving GenAI Performance

- **Self-Reflection (SR)**
  - Involves an LLM reflecting or evaluating its output to understand its reasoning process
  - E.g., ask the model “are you sure?” about the generated output



Improving GenAI Performance

Sentiment analysis using CoT+SR prompt

Models	Prompts	Precision	Recall	F1	Accuracy
GPT-4o	CoT+SR	0.9317	0.9714	0.9491	0.9610

- Let's look at how the model performance can be improved through Python Jupyter Notebook

INNOVATE at Iowa State

Thank You

References

- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models., 35, 24824-24837. *Advances in neural information processing systems*
- Chebbi, K., Ammari, A., Athari, S. A., & Abbass, K. (2024). Do US states' responses to COVID-19 restore investor sentiment? Evidence from S&P 500 financial institutions. *Financial Innovation*, *10*(1), 99.
- Wang, X., Tang, L. R., & Kim, E. (2019). More than words: Do emotional content and linguistic style matching matter on restaurant review helpfulness?. *International Journal of Hospitality Management*, *77*, 438-447.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems, 35, 24824-24837.
- Gola, A. (2024, February 29). *Reflection agents*. LangChain Blog. https://blog.langchain.dev/reflection-agents/
- DeepLearning.AI. (2024). *ChatGPT Prompt Engineering for Developers*. https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/
- DeepLearning.AI. (2024). *Building Systems with the ChatGPT API*. https://www.deeplearning.ai/short-courses/building-systems-with-chatgpt/