



# 深度學習 Deep Learning (14)

112-1

朱學亭老師



# 課程大綱

- W1-課程介紹/Introduction
- W2-Python/Colab and TensorFlow
- W3-Numpy/Pandas and PyTorch
- W4-Sklearn and 機器學習
- W5-神經網路, TensorFlow, PyTorch
- W6-載客熱點預測
- W7-自動光學檢查(AOI)-1
- W8-自動光學檢查(AOI)-2
- W9-Midterm presentation
- W10-RNN
- W11-YoloV5
- W12-AICUP 1
- W13-AICUP 2
- W14-GAN
- W15-NLP1
- W16-NLP2
- W17-Final presentation(1)
- W18-Final presentation(2)



# 大綱

- Topic 1: AICUP
- Topic 2: LLM
- Topic 3: LLM-based Data De-identification



# 課程大綱

- W1-課程介紹/Introduction
- W2-Python/Colab and TensorFlow
- W3-Numpy/Pandas and PyTorch
- W4-Sklearn and 機器學習
- W5-神經網路, TensorFlow, PyTorch
- W6-載客熱點預測
- W7-自動光學檢查(AOI)-1
- W8-自動光學檢查(AOI)-2
- W9-Midterm presentation
- W10-RNN
- W11-YoloV5
- W12-AICUP 1
- W13-AICUP 2
- W14-GAN
- W15-NLP1
- W16-NLP2
- W17-Final presentation(1)
- W18-Final presentation(2)



# 大綱

- Topic 1: AICUP
- Topic 2: Review of Deep Learning models
- Topic 3: NLP

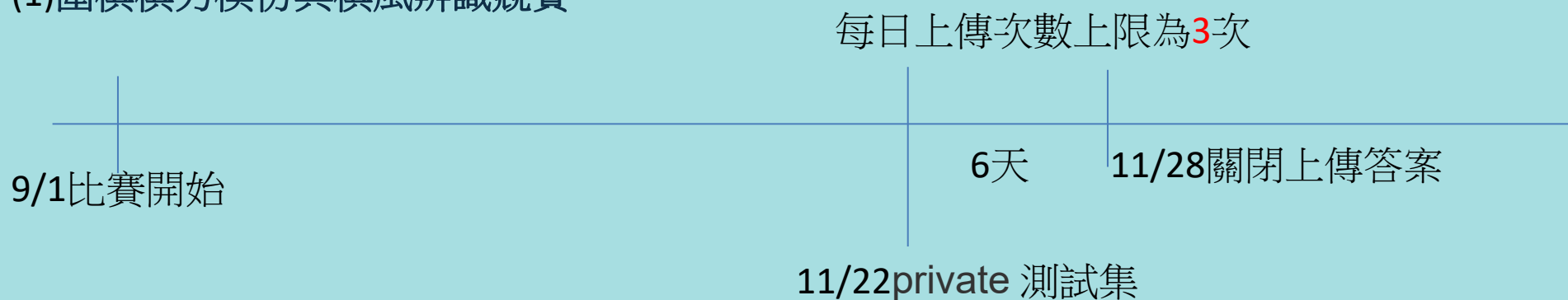


# Topic 1: AICUP



# 時程

## (1) 圍棋棋力模仿與棋風辨識競賽



## (2) 隱私保護與醫學數據標準化競賽



# 比賽結果

- 110021052 許聿萱 0.2491 ()
- 110021059 湯于葳 0.2170 (69)
- 110021190 郭承勳 0.2170 (69)
- 110021075 陳政翔 ?
- 110021181 莊雅婷 0.1440 (80)

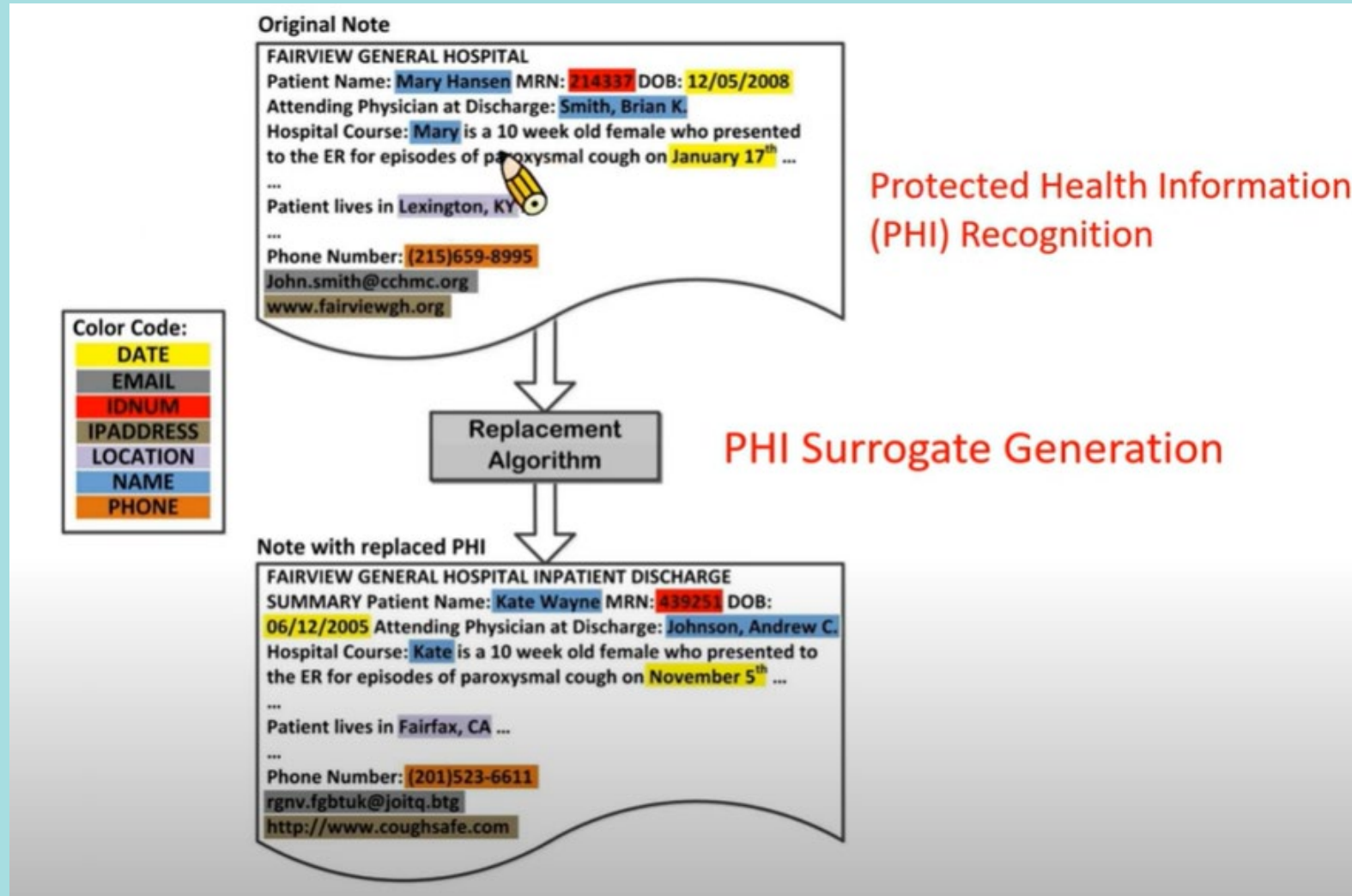




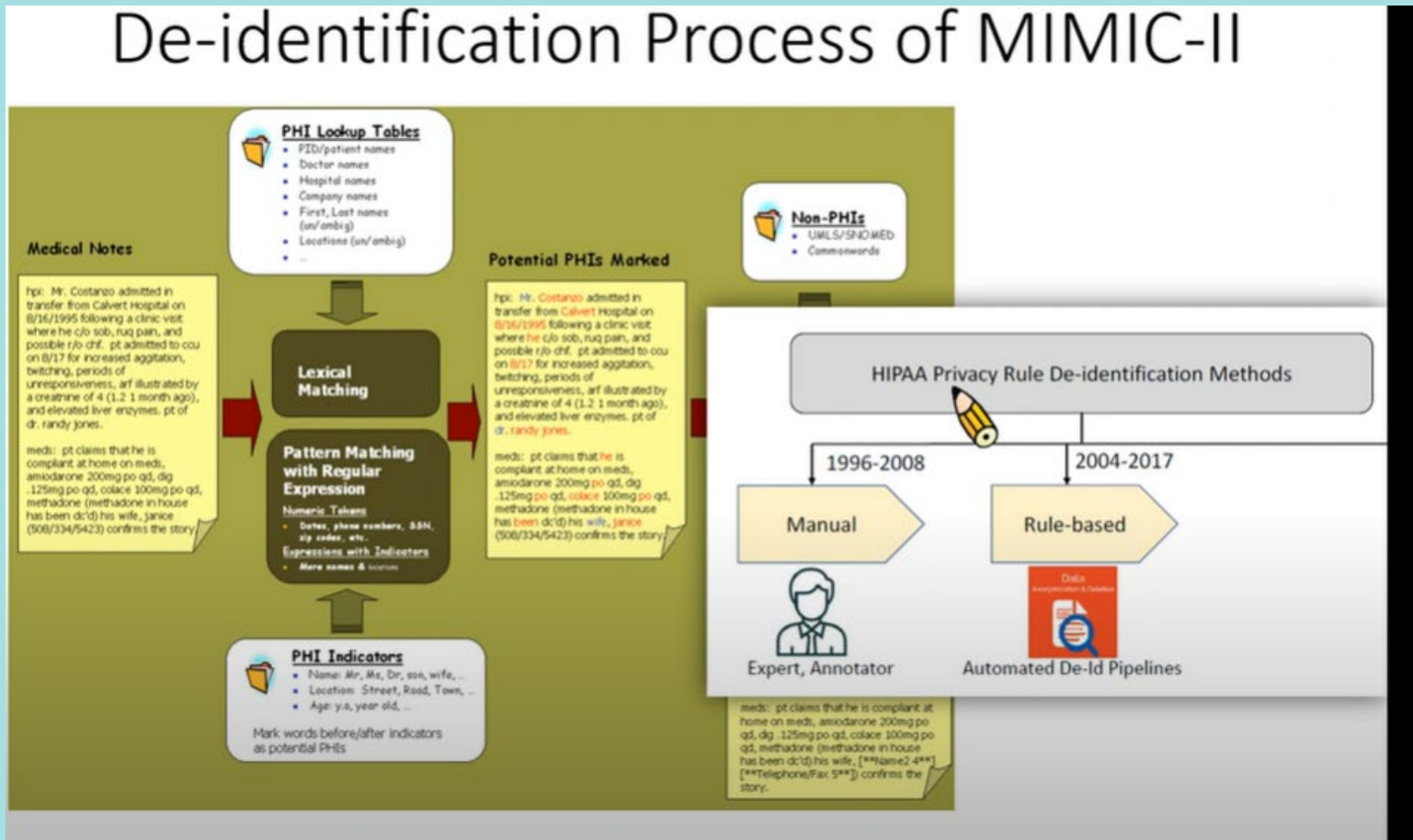
# 受保護健康訊息 ( Protected Health Information; PHI )

PHI 類別	類型定義	範例
姓名	病患名、醫師名、人名	John Doe, Dr. Max, Mr. Smith
職業	無	lawyer, teacher
地點	診間號、部門、醫院、組織、街、城市、州 國家、區號、其他	peri-operative unit-pow, macquarie ward- rhw,12 abc street
年齡	無	23, 98
日期	日期、時間、週期、頻率	24/12/1987, September 26th
聯絡方式	手機號碼、傳真、電子郵件信箱、網址、網 際網路協定位址	+61-421123456、abc@gmail.com、 194.223.1.1
識別符	社群安全碼、醫療紀錄號碼、健康計畫號碼 帳戶、證照號碼、車牌、裝置號碼、生物識 別碼、識別碼	Mrn : 9174338 Id number : 12rl500257
其它	無	

# De-identification

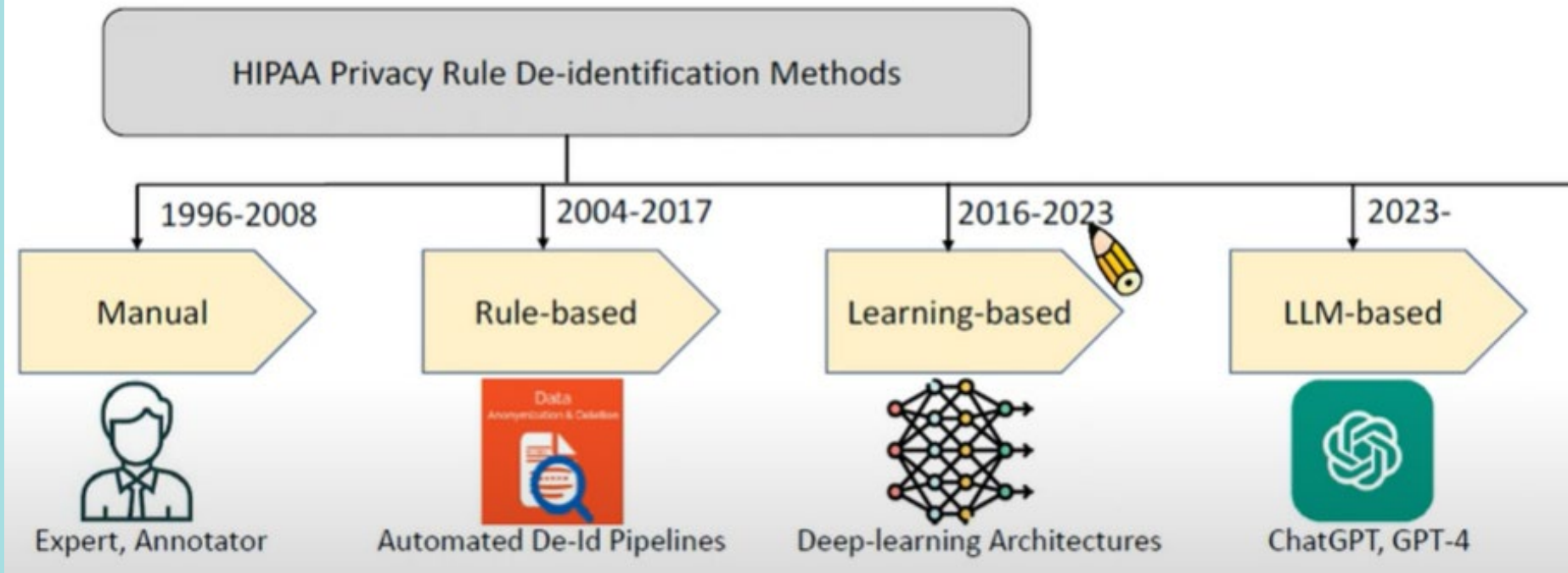


# 傳統的De-identification作法



# LLM-based De-identification

## Development History of De-identification Methods in Accordance with HIPAA



# In Context Learning

- The ability of a model to infer (or learn) the task from input examples
  - The resulting output of the model reflects that new task as if the model had “learned”
    - Generative pre-trained transformer (GPT)
- Zero-shot
  - Given a natural language description of a task at inference time, and anticipate the model to generate the correct response
    - No weights are updated





# Prompting

- A way to turn large language models into a model that performs a specific task
  - Provide the question in natural language and achieve high zero-shot ability across many tasks
- Example

Context →	Q: What is $(2 * 4) * 6$ ?
	A:
Target Completion →	48

**Figure G.42:** Formatted dataset example for Arithmetic 1DC

Context →	Q: What is 17 minus 14?
	A:
Target Completion →	3



**Figure G.43:** Formatted dataset example for Arithmetic 2D-

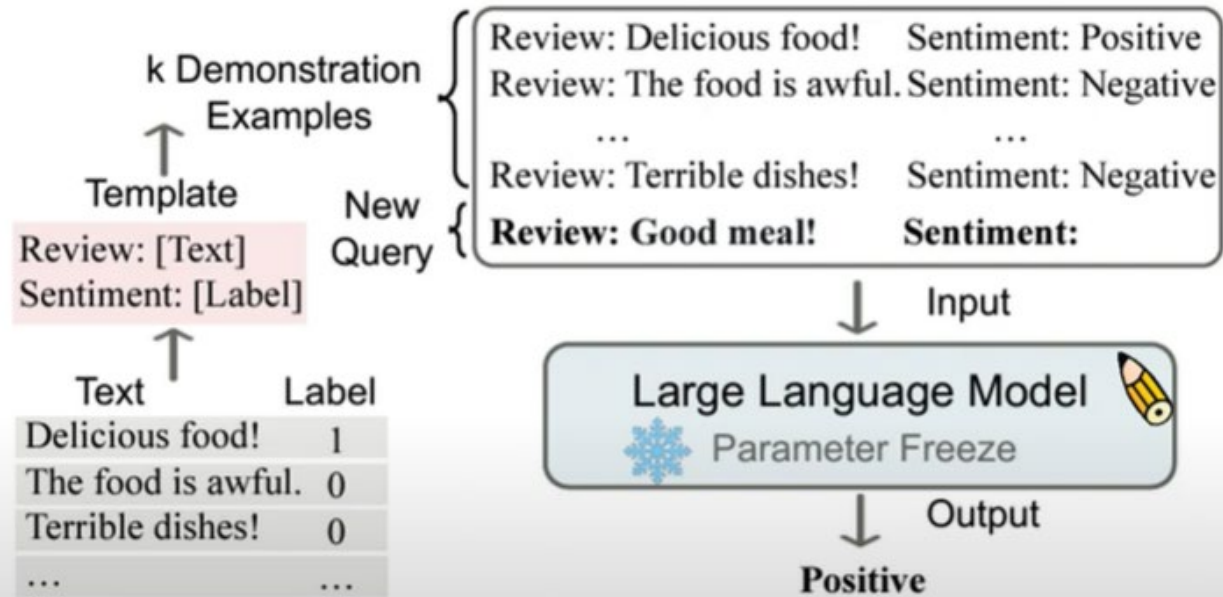
Context →	Q: What is 98 plus 45?
	A:
Target Completion →	143

**Figure G.44:** Formatted dataset example for Arithmetic 2D+



# In Context Learning

## New Paradigm: In Context Learning (ICL)

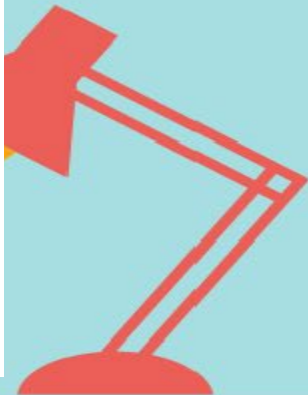


- Learn from analogy
  - No parameter updates



# Causal language models

- Causal language models are frequently used for text generation
  - Use for creative applications like intelligent coding assistant, smart reply, chatbot, etc.
- Causal language modeling predicts the next token in a sequence of tokens, and the model can only attend to tokens on the left
  - The model cannot see future tokens





# Prompt Design

**List the diseases mentioned in the following sentences.**

**Sentence:** Acute liver failure in two patients with regular alcohol consumption ingesting paracetamol at therapeutic dosage.

**Diseases:** Acute liver failure

**Sentence:** Clinical evaluations suggested an initial diagnosis of severe thrombocytopenia and haemolysis.

**Diseases:** thrombocytopenia, haemolysis

- Three main parts of a prompt
  - Overall task instructions
  - A sentence introduction
  - A retrieval message

# Pretraining

- Model at the start:
  - Know nothing about the world
  - Cannot generate any meaningful sentences
- Next word prediction on giant corpora of text data
  - Collected from the Internet
  - Unlabeled
- After **pre-training**
  - Learn to know the natural language
  - Learn to know the knowledge



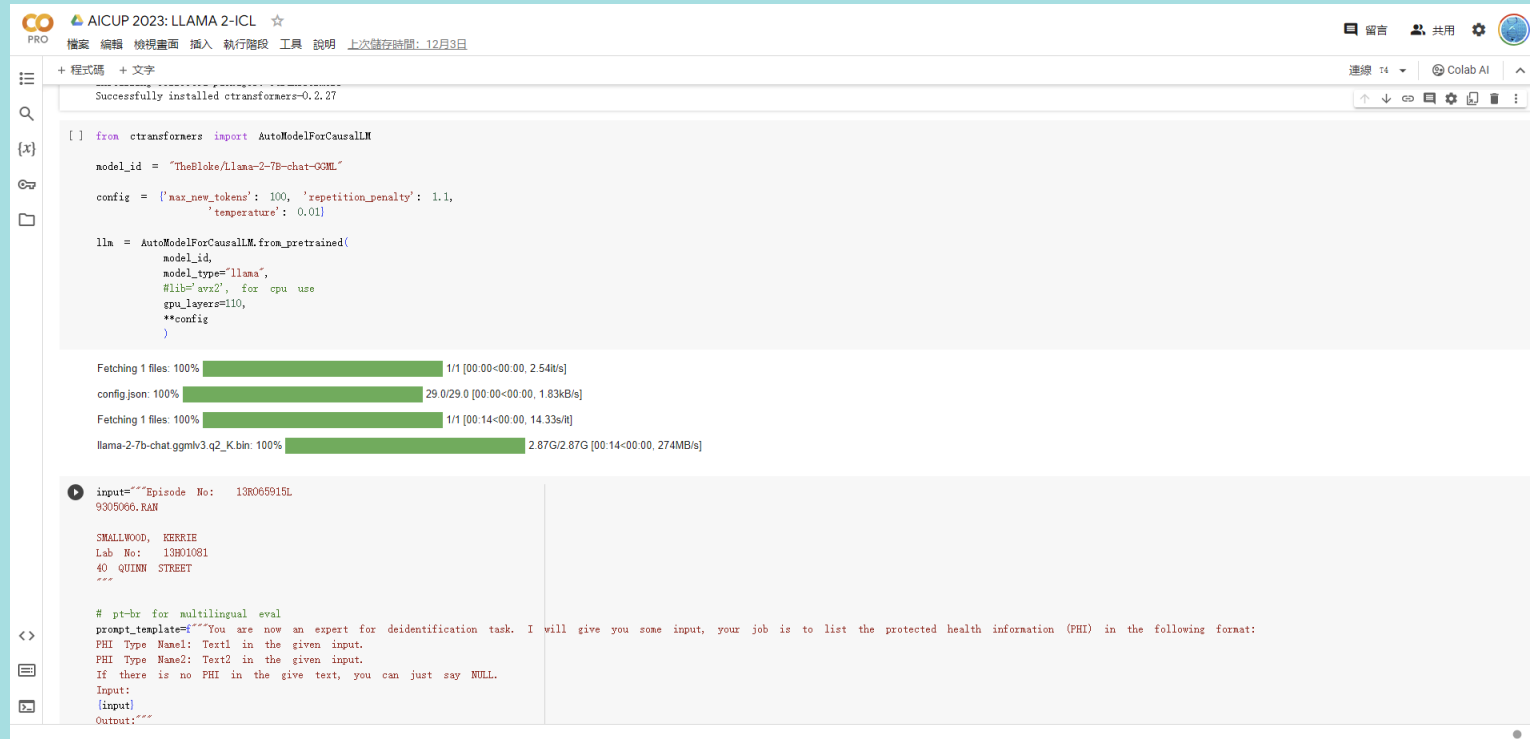
# Fine Tuning

## GPT Fine Tuning

- Fine-tuning is a way to control both the structure and the theme of the text generated by GPT based on the input dataset
- Why?
  - Steer the model to generate more consistent outputs
  - Customize the model to specific use cases
  - Reduce hallucinations
  - **No need to provide as many examples in the prompt**



# LLM-based Data De-identification



```
Successfully installed ctransformers-0.2.27

[ ] from ctransformers import AutoModelForCausalLM

model_id = "TheBloke/Llama-2-7b-chat-GGML"

config = {'max_new_tokens': 100, 'repetition_penalty': 1.1,
          'temperature': 0.01}

lla = AutoModelForCausalLM.from_pretrained(
    model_id,
    model_type="llama",
    #lib='avx2', for cpu use
    gpu_layers=110,
    **config
)

Fetching 1 files: 100% ██████████ 1/1 [00:00<00:00, 2.54kB/s]
config.json: 100% ██████████ 29.0/29.0 [00:00<00:00, 1.83kB/s]
Fetching 1 files: 100% ██████████ 1/1 [00:14<00:00, 14.33s/it]
llama-2-7b-chat-ggmlv3.q2_K.bin: 100% ██████████ 2.87G/2.87G [00:14<00:00, 274MB/s]

input="Episode No: 13R005915L
9305066.RAM

SMALLWOOD, KERRIE
Lab No: 13H01081
40 QUINN STREET
---

# pt-br for multilingual eval
prompt_template="""You are now an expert for deidentification task. I will give you some input, your job is to list the protected health information (PHI) in the following format:
PHI Type Name1: Text1 in the given input.
PHI Type Name2: Text2 in the given input.
If there is no PHI in the give text, you can just say NULL.
Input:
[input]
Output:"""
```



# 如何微調你的 LLM ？

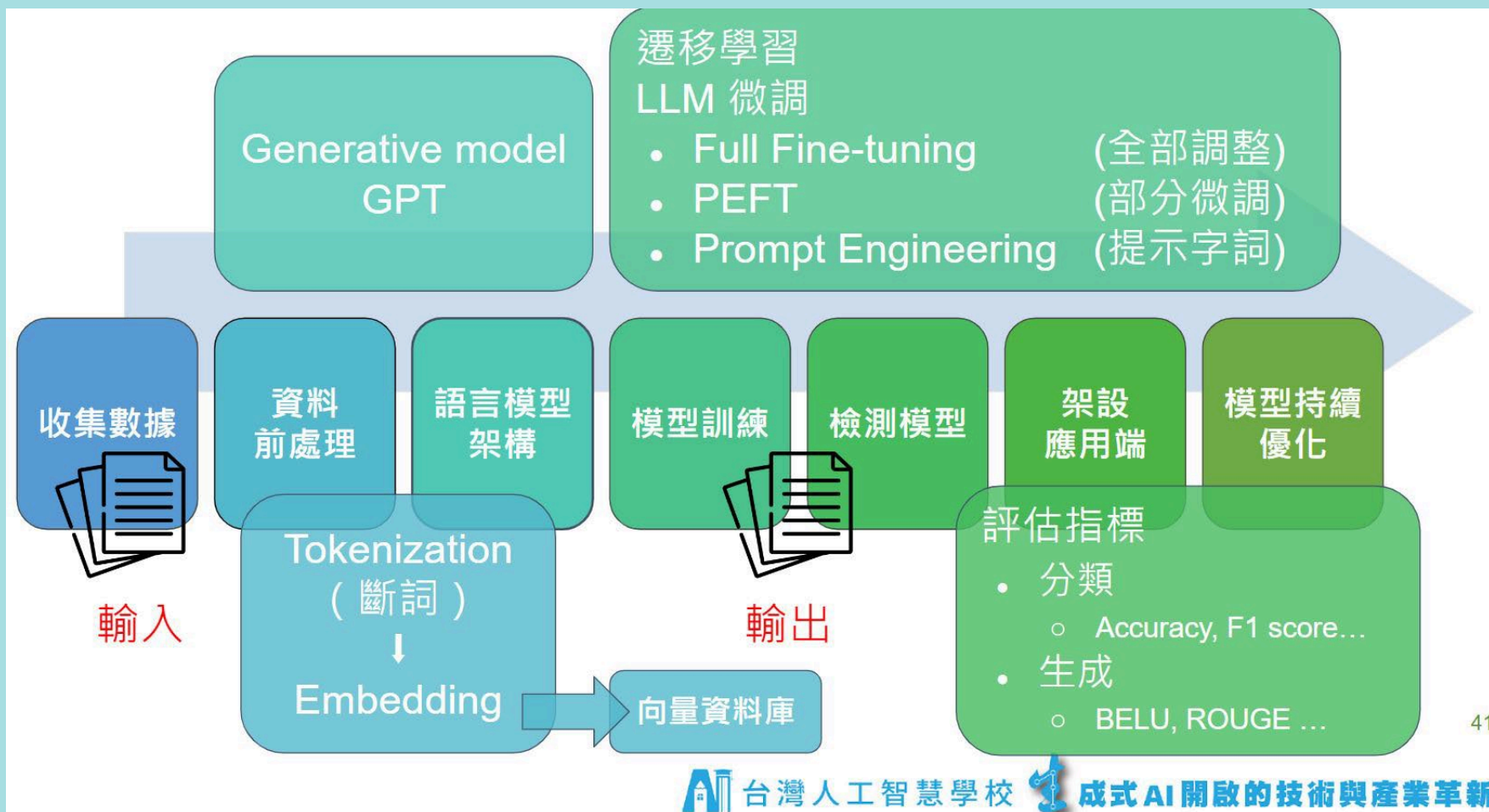
- Prompt Engineering → 提示字詞
- RAG(Retrieval Augmented Generation)→ 檢索手冊
- Parameter-efficient Fine-tuning(PEFT) → 部分微調
- Full Fine-tuning → 整體微調

## 微調 LLM 總結

方法項目	LLM 重頭訓練	Full Fine-Tuning	PEFT	Prompt Engineering
資料集	巨量	大量	少量	x
運算資源	巨量	巨量	少量	x
時間 ( 相同資料量 )	較長	較長	較短	x
精準度 ( 特定領域資料集 )	較高	中等	中等	較低



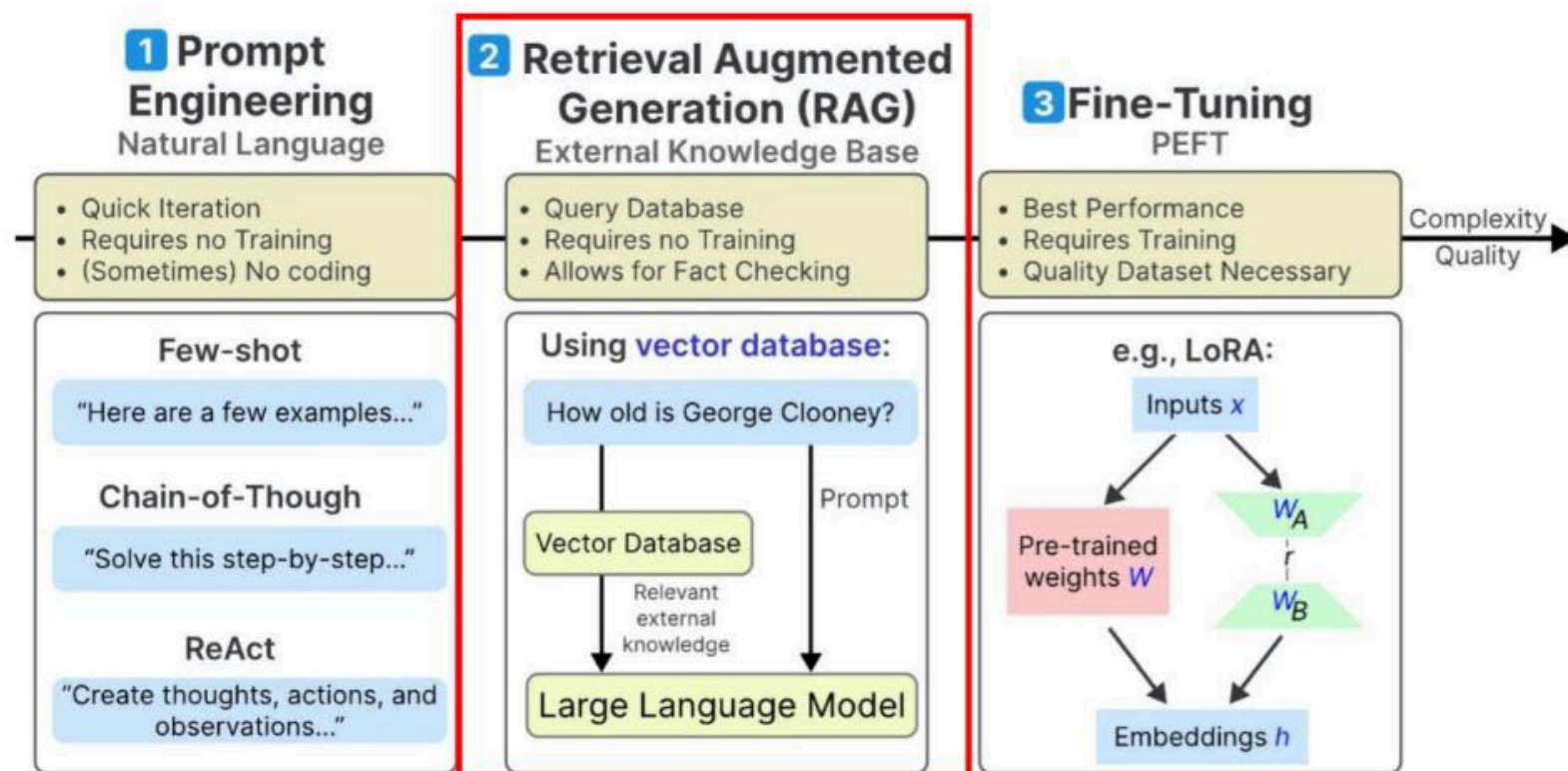
# 如何微調你的 LLM？考慮資料集？運算資源？時間？準確率？





# RAG

## 讓語言模型產出適當回應的幾種方法



Reference: <https://www.maartengrootendorst.com/blog/improving-llms/>



# 4行學生成式AI

four line.py

```
1  #pip install langchain(此行在你的CMD或終端(Terminal)中運行)
2
3  from langchain.llms import OpenAI
4
5  llm = OpenAI(temperature = 0.9)
6
7  text = "請告訴我如何泡一杯好喝的咖啡"
8
9  print(llm(text))
```

TERMINAL

PROBLEMS

OUTPUT

DEBUG CONSOLE

JUPYTER

PS C:\Users\user\Documents\Langchain> & C:/Users/user/python.exe c:/Users/user

泡一杯好喝的咖啡的關鍵在於選擇咖啡豆、研磨的方式以及泡咖啡的方法。以下是一個簡

4. 控制泡咖啡的時間：根據使用的咖啡器具，泡咖啡的時間可能會有所不同。通常，濃

5. 保持器具乾淨：經常清潔咖啡器具，以防止殘留的咖啡油和殘渣影響咖啡的風味。清

以上步驟是基本的泡咖啡方法，當然還有其他複雜的咖啡泡法可以挑戰，取決於你的個人

PS C:\Users\user\Documents\Langchain> □





# Topic 2: REVIEW OF DEEP LEARNING MODELS



# Deep Learning models

- Encoder-Decoder
- Seq2Seq ( Sequence-to-Sequence )
- GAN ( Generative Adversarial Network )
- RNN 、 LSTM 、 Transformer 和 BERT
- 自然語言處理 (NLP) 領域的重要模型



# Encoder-Decoder

- **概念：** Encoder-Decoder 是一種架構，用於處理序列到序列的問題，其中輸入序列經過編碼（ Encoder ）後，生成一個固定大小的向量，然後該向量再經過解碼（ Decoder ）生成輸出序列。
- **應用：** 常見應用包括機器翻譯（ 將一種語言的句子翻譯成另一種語言 ）、文本摘要、語音識別等。

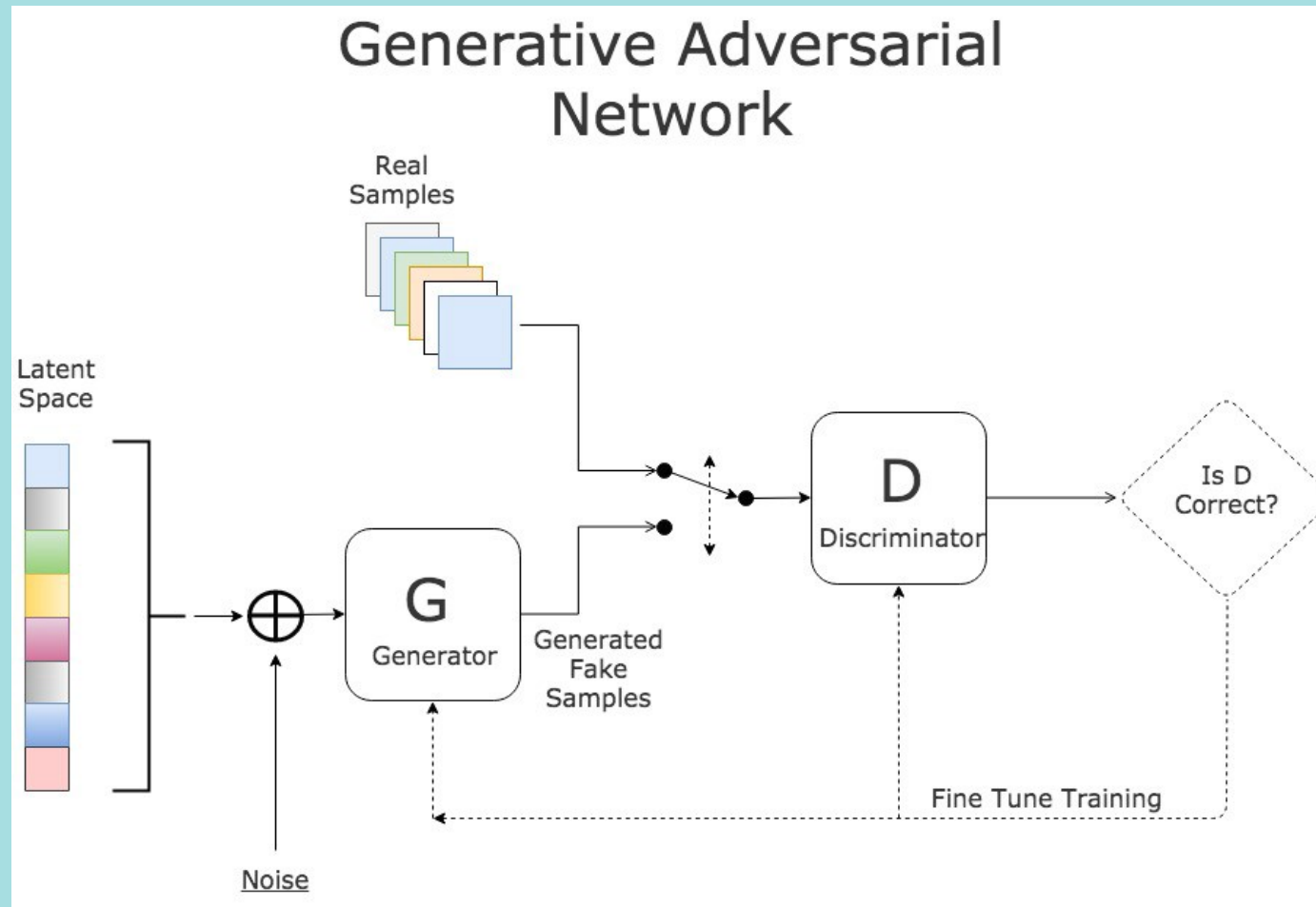


# GAN ( Generative Adversarial Network )

- **概念：** GAN 是一種生成模型，由一個生成器（ Generator ）和一個判別器（ Discriminator ）組成。生成器試圖生成與真實數據相似的樣本，而判別器試圖區分生成器生成的樣本和真實數據。這兩個網路透過對抗過程相互學習。
- **應用：** GAN 在圖像生成、風格轉換、影片生成等領域取得了卓越的成果。它可以生成逼真的樣本，甚至可以生成不存在的人臉、藝術品等。



# Generative Adversarial Network



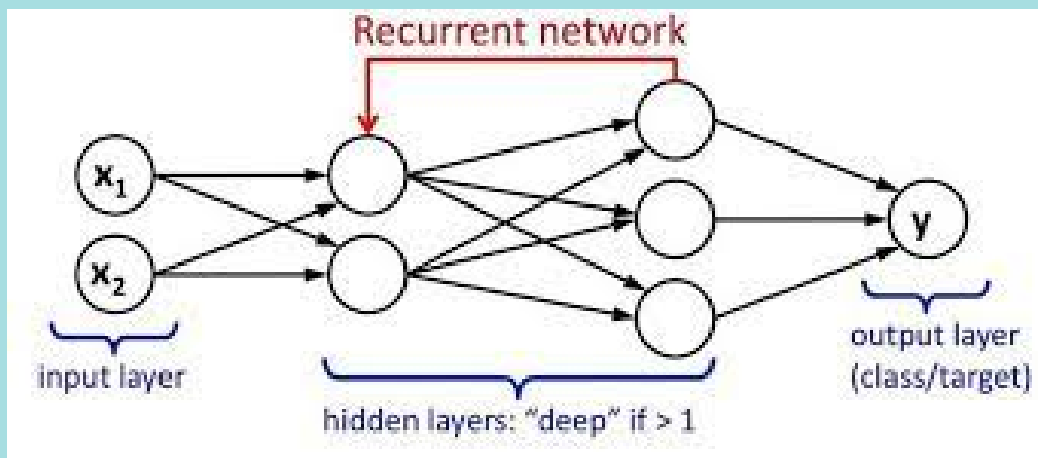
# Seq2Seq ( Sequence-to-Sequence )

- **概念**：Seq2Seq 是一種特殊的 Encoder-Decoder 架構，專門用於處理序列到序列的任務。它主要包含兩個部分，一個是編碼器（Encoder），另一個是解碼器（Decoder）。
- **應用**：Seq2Seq 常見的應用包括機器翻譯、對話生成、文字摘要等。在這種架構中，編碼器將輸入序列轉換成固定大小的向量，然後解碼器使用該向量生成輸出序列。



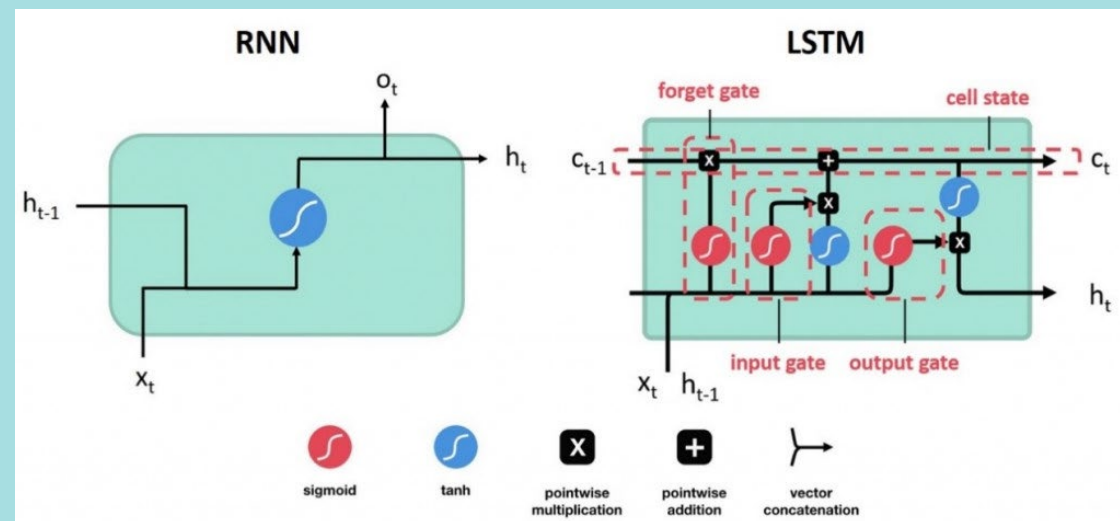
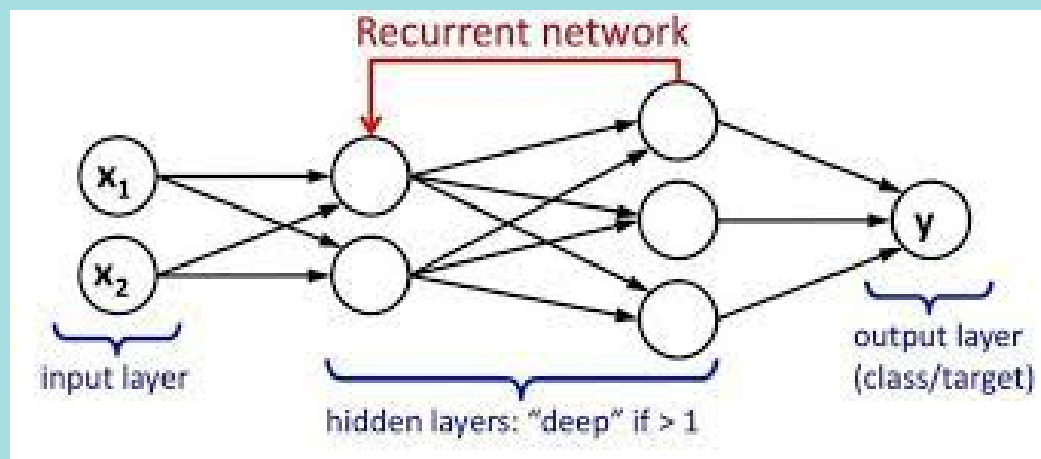
# RNN

- RNN 是一種迴圈神經網路 (Recurrent Neural Network)，可以處理序列資料。
- RNN 的輸入是序列中的一個元素，輸出是序列中的下一個元素。



# LSTM

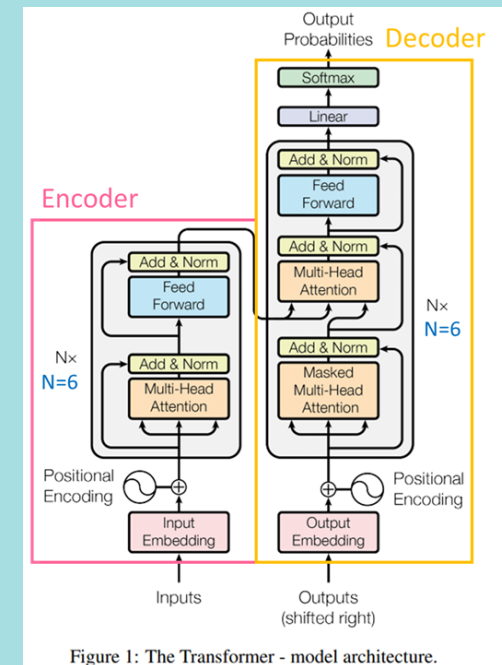
- LSTM 是一種 RNN 的變體，可以解決 RNN 在處理長序列時容易出現的梯度消失問題。
- LSTM 使用了記憶單元 (memory cell) 來儲存序列中的信息。





# Transformer

- Transformer 是一種不需要使用迴圈的 NLP 模型，可以直接處理序列中的所有元素。
- Transformer 使用了自注意力 (self-attention) 機制來處理序列中的依賴關係。



# BERT

- BERT 是一種基於 Transformer 的模型，在大量文字和程式碼資料上進行訓練。
- BERT 可以用於許多 NLP 任務，包括自然語言理解 (NLU)、自然語言生成 (NLG) 和機器翻譯。



# 模型特點

模型	特點	應用
RNN	可以處理序列資料	語音辨識、機器翻譯、文字分類
LSTM	解決了 RNN 的梯度消失問題	語音辨識、機器翻譯、文字分類
Transformer	不需要使用迴圈	語言模型、機器翻譯、問答系統
BERT	基於 Transformer	自然語言理解、自然語言生成、機器翻譯





Thanks!

Q&A

