畢業專題之旅

113.3 暑假篇

時間表(6/26-7/16)

重溫	AI程式語言	
實作	hw	
	test	
論文	language_understanding_paper.pdf	GPT-1 | Paper Explained & PyTorch Implementation - YouTube
	(gpt1)	GPT-1 Paper Explained - YouTube
		GPT Explained!
		GPT，GPT-2，GPT-3 论文精读【论文精读】
	Language Models are Unsupervised Multitask Learners	(GPT-2) Language Models are Unsupervised Multitask Learners | Paper Explained - YouTube
	(gpt2)	GPT-2: Language Models are Unsupervised Multitask Learners - YouTube
		GPT2 Explained! - YouTube
	Language Models for Code Completion: A Practical Evaluation	
額外	Stanford CS224N: Translation, Seq2Seq, Attention	
	NLP course: Week 7 and Week 10	

時間表(7/17-7/30)

重溫	機器學習	
實作	hw	
	test	
論文	Attention Is All You Need	Transformer论文逐段精读
	(Transformer)	Transformer - Attention Is All You Need | Paper Explained
	基於深度學習的醫學文字探勘與序列標註任務	
	Language Models are Few-Shot Learners	GPT-3: Language Models are Few-Shot Learners (Paper Explained) - YouTube
	(gpt3)	GPT-3 - Language Models are Few-Shot Learners | Paper Explained - YouTube
額外	Transformer	GPT原理讲解。什么是Transformer模型？Attention！！ - YouTube
		Transformer架構介紹語衍生模型
	gpt more	GPT - Explained! - YouTube
		(Image-GPT) Generative Pretraining from Pixels | Paper Explained + Colab Notebook - YouTube
		What is GPT in ChatGPT - GPT paper explained - YouTube

時間表(7/31-8/27)

7/31-8/13	重溫	深度學習
	實作	hw
		test
	論文	Improved Unsupervised Chinese Word Segmentation Using Pre-trained Knowledge and Pseudo-labeling Transfer
		Enhance Content Selection for Multi-Document Summarization with Entailment Relation
		CFEVER: A Chinese Fact Extraction and VERification Dataset
8/14-8/27	重溫	自然語言處理
	論文	Exploring the Effectiveness of Pre-training Language Models with Incorporation of Diglossia for Hong Kong Content
		Unsupervised single document abstractive summarization using semantic units
		Improving multi-criteria Chinese word segmentation through learning sentence representation

Week 1

6/26-7/16

論文1

Improving Language Understanding
by Generative Pre-Training
(GPT1)

研究背景與動機

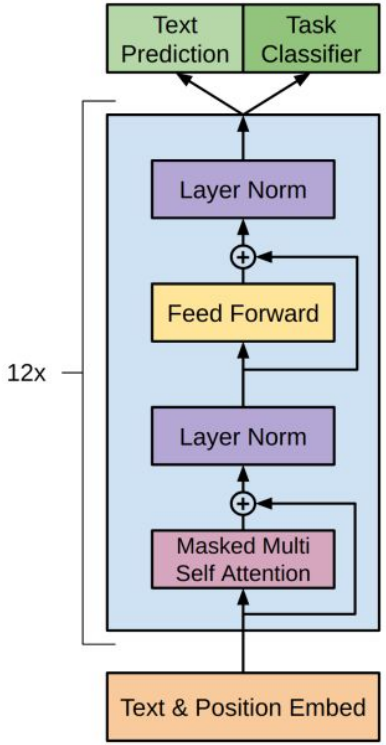
傳統 NLP 模型依賴大量標註資料，成本高、資料稀缺。

本研究提出一種「生成式預訓練 + 任務微調」的方法，能有效利用無標註語料，提升語言理解能力。

開啟了大型語言模型（LLM）時代，是 GPT 系列的起點。

模型架構與技術亮點

模型元件	說明
Transformer 架構	使用 Decoder-only 結構，12 層堆疊
注意力頭數	12 個多頭注意力，捕捉不同語意關係
隱藏層維度	每個 token 表示為 768 維向量
前饋層維度	每層 FFN 擴展至 3072 維，提升非線性表達力
Loss Function	預訓練使用 Cross-Entropy，自回歸語言模型
優化器	Adam，學習率約 6.25e-5，使用 LayerNorm



實驗設計與成果

預訓練階段（非監督式）

微調階段（監督式）

預訓練階段(非監督式)

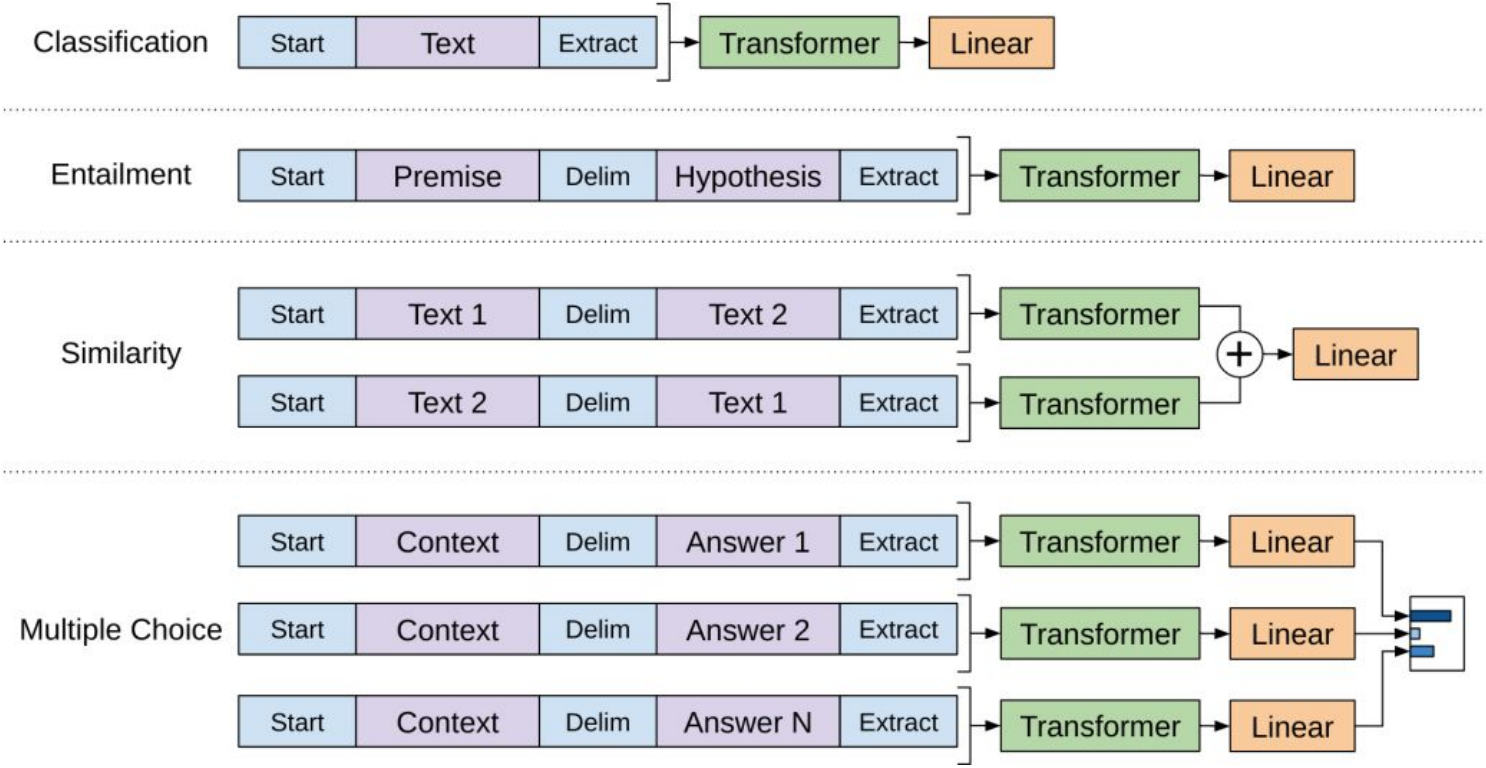
- 使用 BooksCorpus(約 7,000 本書)進行語言建模。
- 模型學習預測下一個詞，建立通用語言理解能力。

微調階段(監督式)

測試 12 個 NLP 任務，包括：

Task	Datasets
Natural language inference	SNLI [5], MultiNLI [66], Question NLI [64], RTE [4], SciTail [25]
Question Answering	RACE [30], Story Cloze [40]
Sentence similarity	MSR Paraphrase Corpus [14], Quora Question Pairs [9], STS Benchmark [6]
Classification	Stanford Sentiment Treebank-2 [54], CoLA [65]

微調階段(監督式)


Classification
Start
Text
Extract
Transformer
Linear
Entailment
Start
Premise
Delim
Hypothesis
Extract
Transformer
Linear
Similarity
Start
Text 1
Delim
Text 2
Extract
Transformer
Start
Text 2
Delim
Text 1
Extract
Transformer
Linear
Multiple Choice
Start
Context
Delim
Answer 1
Extract
Transformer
Linear
Start
Context
Delim
Answer 2
Extract
Transformer
Linear
Start
Context
Delim
Answer N
Extract
Transformer
Linear

成果亮點

在 12 項任務中有 9 項刷新 SOTA 表現：

a. Story Cloze 提升 8.9%
b. RACE 提升 5.7%
c. MultiNLI 提升 1.5%
d. GLUE 總分提升至 72.8（原為 68.9）

關鍵概念解析

零樣本學習（Zero-shot Learning）

- 模型未經特定任務訓練，僅靠語言知識即可推理。
- 展現 GPT 的通用性與語言理解深度。

任務輸入轉換（Prompt Engineering）

- 將不同任務轉換為統一的序列格式，無需改變模型架構。
- 例如：<start> 前提 <delimiter> 假設 <extract> 用於文本蘊涵。

學到什麼

預訓練語言模型能有效遷移至多種 NLP 任務。

Transformer 架構在捕捉長距依賴與語意關係上表現優異。

Prompt 設計是微調成功的關鍵。

零樣本能力讓模型更具彈性與實用性。

論文2

Language Models are Unsupervised Multitask Learners
(GPT2)

研究背景與核心概念

傳統 NLP 模型通常針對單一任務進行有監督學習，需要大量標註資料。

本研究提出：語言模型本身就能透過無監督學習，學會多種任務，只要訓練資料夠大、模型容量夠強。

GPT-2 是基於 Transformer 架構的語言模型，使用了 40GB 的 WebText 資料集進行預訓練。

方法與創新點

使用 純語言建模目標（預測下一個詞）進行訓練，不依賴任務標籤。

模型透過觀察自然語言中的「任務演示」學會執行任務，例如：

“Translate English to French: perfume → parfum”

“Q: Who discovered gravity? A: Isaac Newton”

利用 prompt（提示語）來引導模型執行不同任務，

實現 zero-shot learning。

Question	Generated Answer	Correct	Probability
Who wrote the book the origin of species?	Charles Darwin	✓	83.4%
Who is the founder of the ubuntu project?	Mark Shuttleworth	✓	82.0%
Who is the quarterback for the green bay packers?	Aaron Rodgers	✓	81.1%
Panda is a national animal of which country?	China	✓	76.8%
Who came up with the theory of relativity?	Albert Einstein	✓	76.4%
When was the first star wars film released?	1977	✓	71.4%
What is the most common blood type in sweden?	A	✗	70.6%
Who is regarded as the founder of psychoanalysis?	Sigmund Freud	✓	69.3%
Who took the first steps on the moon in 1969?	Neil Armstrong	✓	66.8%
Who is the largest supermarket chain in the uk?	Tesco	✓	65.3%
What is the meaning of shalom in english?	peace	✓	64.0%
Who was the author of the art of war?	Sun Tzu	✓	59.6%
Largest state in the us by land mass?	California	✗	59.2%
Green algae is an example of which type of reproduction?	parthenogenesis	✗	56.5%
Vikram samvat calender is official in which country?	India	✓	55.6%
Who is mostly responsible for writing the declaration of independence?	Thomas Jefferson	✓	53.3%
What us state forms the western boundary of montana?	Montana	✗	52.3%
Who plays ser davos in game of thrones?	Peter Dinklage	✗	52.1%
Who appoints the chair of the federal reserve system?	Janet Yellen	✗	51.5%
State the process that divides one nucleus into two genetically identical nuclei?	mitosis	✓	50.7%
Who won the most mvp awards in the nba?	Michael Jordan	✗	50.2%
What river is associated with the city of rome?	the Tiber	✓	48.6%
Who is the first president to be impeached?	Andrew Johnson	✓	48.3%
Who is the head of the department of homeland security 2017?	John Kelly	✓	47.0%
What is the name given to the common currency to the european union?	Euro	✓	46.8%
What was the emperor name in star wars?	Palpatine	✓	46.5%
Do you have to have a gun permit to shoot at a range?	No	✓	46.4%
Who proposed evolution in 1859 as the basis of biological development?	Charles Darwin	✓	45.7%
Nuclear power plant that blew up in russia?	Chernobyl	✓	45.7%
Who played john connor in the original terminator?	Arnold Schwarzenegger	✗	45.2%

實驗結果與表現

任務類型	效果摘要
閱讀理解	在 CoQA 資料集上達到 55 F1 分數，**超越 3/4 的有監督基線模型** 。
常識推理	在 Winograd Schema Challenge 中達到 70.7% 準確率，**刷新 SOTA 表現**。
長距依賴建模	在 LAMBADA 測試集上準確率達 63.2%，遠高於先前的 19%。
摘要生成	使用 “TL;DR:” 提示語可生成合理摘要，雖略低於專門訓練的摘要模型，但仍具可用性。
翻譯任務	雖然訓練資料幾乎全是英文，GPT-2 仍能進行英法翻譯，BLEU 分數達 11.5。
問答任務	在 Natural Questions 中，Top 1% 高置信度回答準確率達 63.1%。

關鍵概念解析

- Unsupervised Multitask Learning（無監督多任務學習）：

GPT-2 不依賴標註資料，而是透過大量自然語言文本學會執行多種任務（翻譯、問答、摘要等）。

- Prompt Engineering（提示語設計）：

只要給出適當的提示語，模型就能理解任務並執行，實現 Zero-shot、Few-shot 能力。

- Decoder-only Transformer 架構：

GPT-2 使用純解碼器架構，並採用 Masked Self-Attention，確保模型只能看到前面的詞。

- 自回歸生成（ Autoregressive Generation）：

模型逐字生成文本，每次根據前面的 內容預測下一個詞。

學到什麼

- 語言模型的本質：

  語言模型不只是生成文字，它能理解語言結構、語境、甚至隱含的任務意圖。
- 無監督學習的力量：

  不需要人工標註資料，只靠大量自然語言文本就能學會多種任務。
- Prompt 的魔力：

  透過設計提示語(prompt)，你可以引導模型執行翻譯、問答、摘要等任務。
- Zero-shot 能力：

  模型能在沒見過任務的情況下直接執行，這是多任務學習的一大突破。
- Scaling Law(規模法則)：

  模型越大、資料越多，能力越強，這是後續GPT-3、GPT-4 的基礎理論。

與 GPT-1 的差異比較

Parameters	Layers	d_{model}
117M	12	768
345M	24	1024
762M	36	1280
1542M	48	1600

項目	GPT-1	GPT-2
模型架構	Decoder-only Transformer	相同架構，但更深更大
層數	12 層	最多 48 層
參數量	約 1.17 億	最多 15 億
訓練資料	BookCorpus（約 5GB）	WebText（約 40GB）
任務學習方式	須微調（Fine-tuning）	可 Zero-shot、Few-shot
多任務能力	有限	強大，能處理多種 NLP 任務
表現	基本語言建模	多項任務刷新 SOTA 或接近 SOTA

論文3

Language Models for Code Completion:
A Practical Evaluation

研究目的

評估程式碼語言模型在真實開發中的補全表現

提出改進方向，讓模型更符合實務需求

研究方法

開發 Code4Me IDE 擴充套件

收集來自 1200+ 開發者的 60 萬筆補全資料

線上 + 離線補全比較

評估模型（如 InCoder）在 12 種程式語言中表現

質性分析 1690 筆補全失敗案例

Language	Online #Valid Completions	Offline Random	Offline Trigger Point
Python	219,822	585	598
Java	104,157	902	898
TypeScript	72,492	897	882
PHP	69,215	712	709
JavaScript	55,550	564	548
Kotlin	27,353	-	-
C++	15,742	708	709
Rust	14,063	838	836
C#	12,971	928	916
Go	10,199	757	756
C	3,626	581	579
Scala	878	971	969
Ruby	-	857	848
Total	606,068	9,300	9,248

Failure category plus label ID	Count
Model-oriented Errors	**5,511**
└ Token Level	3,835
└ (ME-T1) Incorrect variable	1,435
└ (ME-T2) Incorrect function	1,162
└ (ME-T3) Incorrect literal	1,130
└ (ME-T4) Incorrect type	108
└ Statement Level	1,676
└ (ME-S1) Wrong parameter count	613
└ (ME-S2) Wrong semantics	352
└ Untimely termination	318
└ (ME-S3) Early termination	205
└ (ME-S4) Late termination	113
└ Rambled Outputs	249
└ (ME-S5) Looped repetition	171
└ (ME-S6) Copied input context	78
└ (ME-S7) Faulty syntax	144
Application-oriented Errors	**2,030**
└ (AE-1) Mid-token invocation	1,173
└ (AE-2) Insufficient context	482
└ (AE-3) Redundant invocation	240
└ (AE-4) Typographical errors in input	135
User-overridden Outputs	**771**
└ (UO-1) Correct but not accepted	605
└ (UO-2) Valid but not preferred	112
└ (UO-3) Accepted but required change	54

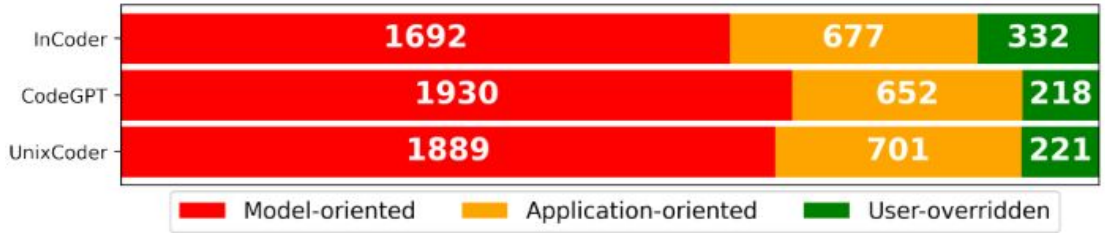
關鍵發現

InCoder 表現最佳，尤其是 Python 和 Java 等主流語言

離線評估 ≠ 真實使用表現

模型限制佔失敗原因最大宗（66.3%）

開發者操作錯誤與使用者覆寫也是原因



關鍵概念解析

1. **補全評估要看「實境」而非「離線」**

- 離線 benchmark 無法反映開發者真實使用情境。
- 真實 IDE 裡的操作、語境、錯誤容忍度才是決定補全好壞的關鍵。

2. **模型 vs 使用者：補全失敗的雙重責任**

- 失敗原因三大類：
  - **模型理解錯誤（ 66.3%）**
  - **開發者使用偏差（ 24.4%）**
  - **使用者改寫補全（ 9.3%）**

3. **語言模型的強項與弱點**

- 主流語言（Python、Java）因為訓練資料多，補全表現明顯較好。
- 非主流語言（像 Rust、Lua）補全易偏離語意。

4. **混合式補全是未來趨勢**

- 單靠語言模型難以處理所有情境。
- 結合 **規則式方法（語法、編譯器）** + **語言模型預測** 能彌補各自不足。

5. **IDE 補全應注重「使用者體驗」**

- 使用者需要：
  - 可理解的補全邏輯
  - 可控的建議選擇
  - 對模型補全結果的即時回饋

額外

Stanford CS224N:
Translation, Seq2Seq, Attention

機器翻譯的歷史演進

- 從 1950 年代的規則式翻譯，到 1990–2010 年代的統計式機器翻譯（SMT），再到現今的神經機器翻譯（NMT）。

- SMT 依賴語言模型與翻譯模型的組合，但需要大量人工設計與特徵工程。

Seq2Seq 模型架構

- 使用兩個 RNN（或 LSTM）組成：編碼器將輸入語句轉換為上下文向量，解碼器根據該向量生成目標語句。

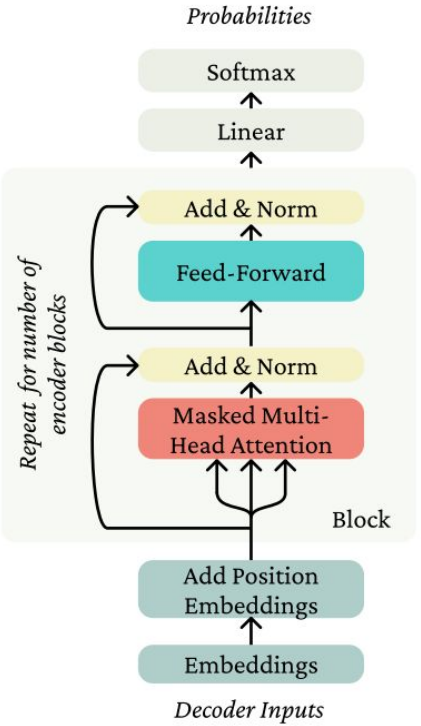
- 支援不同長度的輸入與輸出序列，適用於翻譯、摘要、對話等任務。

解碼策略

- 貪婪解碼：每一步選擇機率最高的詞，效率高但可能錯失最佳翻譯。

- 波束搜尋（Beam Search）：保留多個候選翻譯路徑，

在效率與品質間取得平衡。



注意力機制（Attention）

- 解決 Seq2Seq 模型的「資訊瓶頸」問題，讓解碼器在每一步能聚焦於輸入序列的不同部分。

- 提升翻譯品質、可解釋性，並成為後續 Transformer 架構的基礎。

額外

NLP course: Week 7

主題定位

Decoding Strategies:

說明文字生成的「解碼方法」，聚焦在模型輸出的策略選擇與影響

BERT and its Family:

探討語言模型的演進脈絡，並聚焦在 BERT 及其延伸模型的架構、預訓練策略與應用場景

NLP course: Week 7

BERT and its Family

語言模型演進脈絡

類型	模型代表	特性與轉變
靜態嵌入	Word2Vec, GloVe	每個詞的向量固定，不隨語境改變。
上下文嵌入	ELMo	使用雙向 LSTM，詞意依語境而變。
Transformer 模型	BERT, GPT	深度編碼器或解碼器，能捕捉複雜語意與關係。

BERT 架構與任務

項目	說明
預訓練任務	- Masked Language Modeling：隨機遮蔽詞彙讓模型預測 - Next Sentence Prediction：判斷句子間邏輯關係
下游任務應用	- 句子分類（情緒分析、垃圾郵件偵測） - 問答系統（Q&A） - 命名實體識別（NER）
Fine-tuning	BERT 可針對不同任務加上新層結構並進行微調，增強專業任務表現。

BERT 家族模型差異

模型	MLM策略	特殊改進點
BERT	靜態遮蔽	原始設計
RoBERTa	動態遮蔽	去除NSP，資料量加倍，效果普遍優於BERT
SpanBERT	Span遮蔽	強化 span 表達能力，適合片語理解

Week 7 Quiz

簡答題

請說明static embeddings 和contextual embeddings 的差異和使用情境?

特性	Static Embeddings	Contextual Embeddings
向量是否固定	是(一詞一向量)	否(根據語境變化)
是否考慮上下文語意	不考慮	會考慮
表示能力	基本語意	深層語意(同詞不同義)
使用情境	主題建模、簡單語意分析等	機器翻譯、問答、文字生成等

額外

NLP course: Week 10

主題定位

Decoding Strategies:

探討語言模型在文字生成時如何從眾多候選詞中選擇最合適的下一個字詞。

NLG Evaluations:

講解如何客觀地評估一段由模型生成的文字是否「像人類寫的」、「真實」或「有創意」。

GPT3, InstructGPT, and RLHF:

完整解析 GPT-1 到 GPT-3.5（InstructGPT）演化過程，以及人類回饋強化學習（RLHF）的訓練策略

NLP course: Week 10

decoding_v2

解碼策略種類與特性

解碼策略	原理說明	優點	缺點	常見用途
Greedy Decoding	每步選機率最高的字	快速、簡單	容易犯早期錯誤	快速應答、簡短指令
Beam Search	維持多條候選路徑，最終選機率總和最高者	穩定、可避免早期錯誤	容易重複、偏向短句、缺乏多樣性	翻譯、正式摘要
Top-k Sampling	在機率最高的 k 個字中隨機選一	可提升創造力與多樣性	需要調整 k 值，不同任務影響大	故事生成、創意場景
Top-p Sampling	挑選累積機率超過 p 的最小集合後進行抽樣	語意自然、動態適應語境	p 值仍需設定，但比 Top-k 更穩定	對話系統、開放式生成任務

額外技巧

Softmax 溫度：

調整生成文字的多樣性（高溫 = 更隨機）。

長度正規化：

避免 Beam Search 偏好短句。

混合策略：

例如 Beam + Sampling，兼具穩定與創造力。

NLP course: Week 10

NLG_evaluations

評估指標總整理

指標名稱	目的說明	備註
Perplexity	衡量模型預測下一字的困難程度	越低代表模型越精準
BLEU / ROUGE	和參考答案比較文字重疊度	傳統翻譯評估指標，適用有標準答案的任務
Self-BLEU	比較自己生成文字間的重疊率	越低表示多樣性越高
Repetition %	計算輸出中重複片段的比例	可用於偵測語句退化（text degeneration）
HUSE	結合統計與人類偏好，綜合評分	目前最全面，可反映人類評價與語意品質

NLP course: Week 10

GPT3_InstructGPT_RLHF_0421

演進概覽

模型	時間	技術改進	關鍵貢獻
GPT-1	2018	Transformer Decoder + 預測下一字	開啟生成式語言模型的新方向
GPT-2	2019	更大模型、更多資料	支援多任務語言能力
GPT-3	2020	Few-shot 學習、175B 巨量參數	不需微調也可完成多種任務
InstructGPT	2022	RLHF 訓練模型理解人類指令	真實性高、偏見低、貼近人類期待

InstructGPT 的三階段訓練

Supervised Fine-Tuning (SFT):

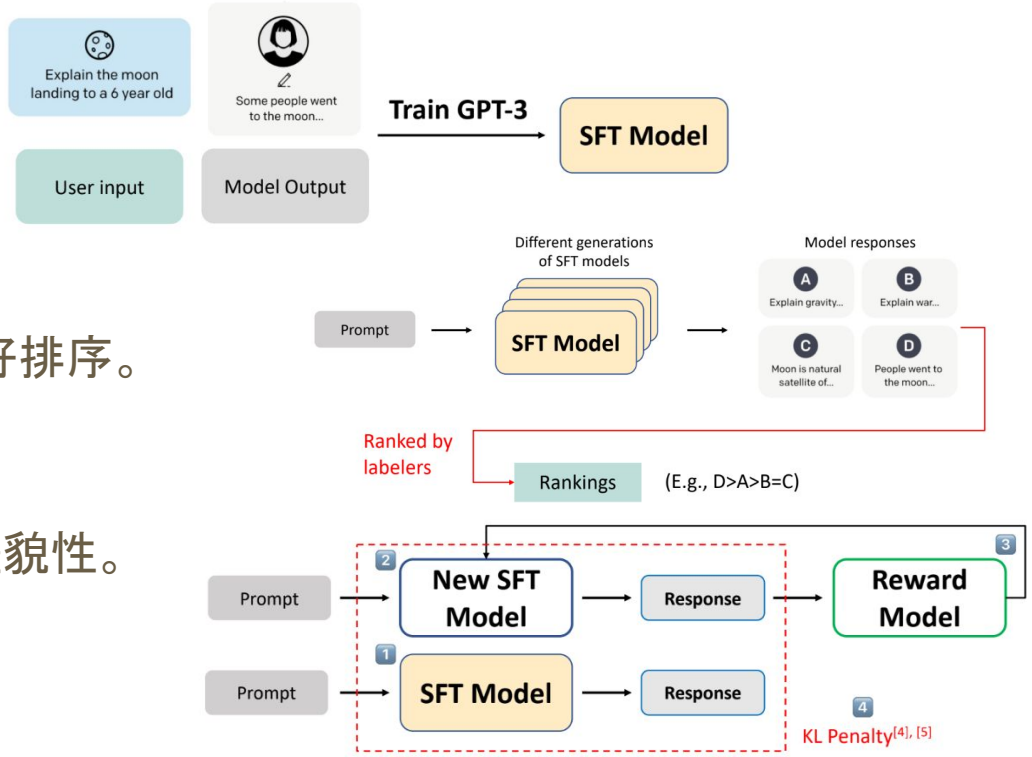
用人類標註資料微調模型。

Reward Model Training:

訓練模型評分回答的好壞，依人類喜好排序。

Reinforcement Learning (PPO):

透過回饋調整模型行為，改善語意、禮貌性。



Week 10 Quiz

選擇題

Q1: Softmax的值比較大，模型的輸出機率分布會？ (1) 比較多樣 (2) 比較固定

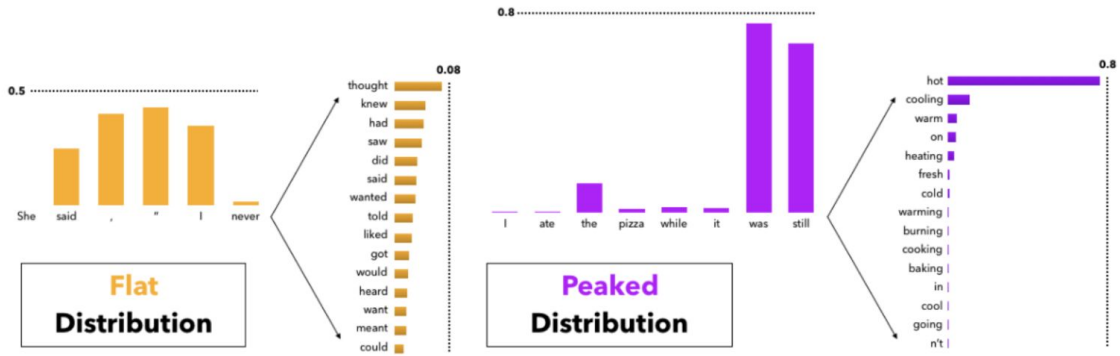
(1) 比較多樣

Softmax 值較大

- 機率分布會變得平坦。
- 原本差距不大的字詞機率會更加接近。
- 輸出較多樣性。

-當 溫度較低：

- 機率分布會變得更尖 銳。
- 高機率字詞被強烈偏好。
- 輸出變得更保守、固定。



Week 10 Quiz

簡答題

Q2: 為什麼 Top-p sampling 相較於 Top-k sampling 比較不需要調整參數？

Top-p sampling 根據累積機率總和動態選取候選字詞，能自動適應不同語境的詞彙分布，因此只需設定一個通用的 p 值（如 0.9 或 0.95）即可達到穩定效果；

而 Top-k 固定選 k 個字詞，需依任務內容反覆調整 k，否則容易過度限制或引入雜訊。

這使得 Top-p 在實作上參數調整負擔較小且效果更穩定。