

---

# 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	<a href="#">language_understanding_paper.pdf</a>	<a href="#">GPT-1   Paper Explained &amp; PyTorch Implementation - YouTube</a>
	(gpt1)	<a href="#">GPT-1 Paper Explained - YouTube</a>
		<a href="#">GPT Explained!</a>
		<a href="#">GPT, GPT-2, GPT-3 论文精读【论文精读】</a>
	<a href="#">Language Models are Unsupervised Multitask Learners</a>	<a href="#">(GPT-2) Language Models are Unsupervised Multitask Learners   Paper Explained - YouTube</a>
	(gpt2)	<a href="#">GPT-2: Language Models are Unsupervised Multitask Learners - YouTube</a>
		<a href="#">GPT2 Explained! - YouTube</a>
	<a href="#">Language Models for Code Completion: A Practical Evaluation</a>	
	<a href="#">Stanford CS224N: Translation, Seq2Seq, Attention</a>	
額外	<a href="#">NLP course: Week 7 and Week 10</a>	

# 時間表 (7/17-7/30)

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

# 時間表 (7/31-8/27)

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

---

---

# Week 1

6/26-7/16

---

---

---

# 論文1

— Improving Language Understanding  
by Generative Pre-Training  
(GPT1) —

---

# 研究背景與動機

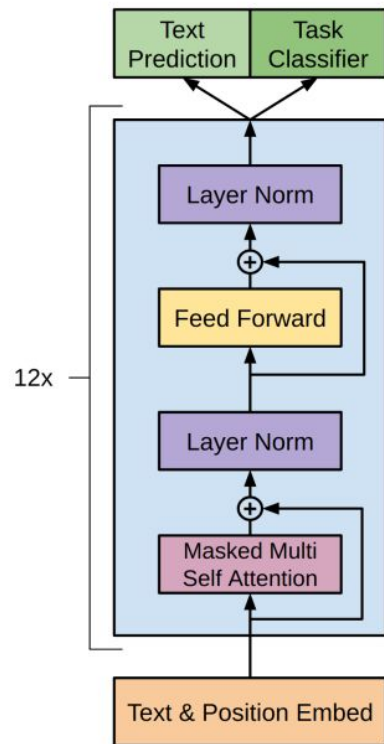
傳統 NLP 模型依賴大量標註資料，成本高、資料稀缺。

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)

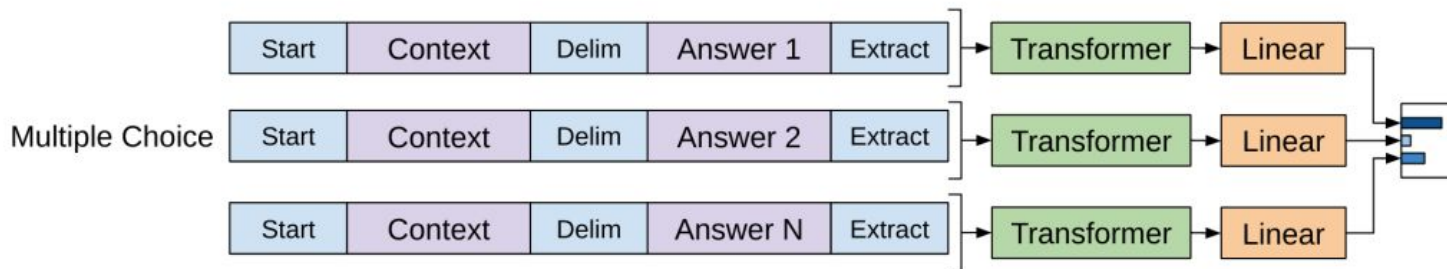
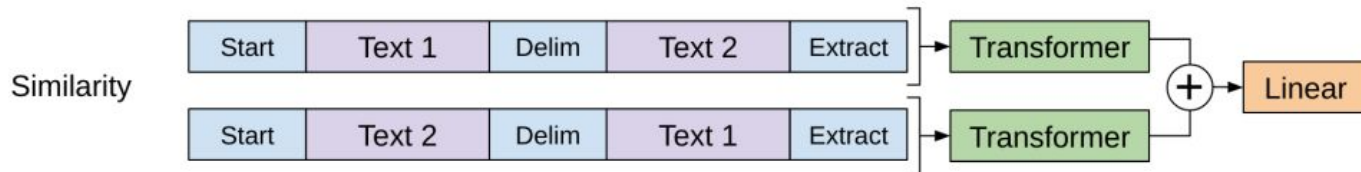
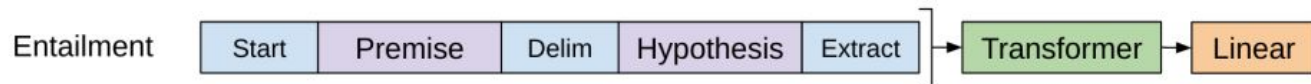
- 使用 BooksCorpus(約 7,000 本書)進行語言建模。
- 模型學習預測下一個詞, 建立通用語言理解能力。

# 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)



# 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， <b>超越 3/4 的有監督基線模型</b> 。
常識推理	在 Winograd Schema Challenge 中達到 70.7% 準確率， <b>刷新 SOTA 表現</b> 。
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	Offline	
	#Valid Completions	Random	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
<b>Model-oriented Errors</b>	<b>5,511</b>
└ 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
<b>Application-oriented Errors</b>	<b>2,030</b>
└ (AE-1) Mid-token invocation	1,173
└ (AE-2) Insufficient context	482
└ (AE-3) Redundant invocation	240
└ (AE-4) Typographical errors in input	135
<b>User-overridden Outputs</b>	<b>771</b>
└ (UO-1) Correct but not accepted	605
└ (UO-2) Valid but not preferred	112
└ (UO-3) Accepted but required change	54

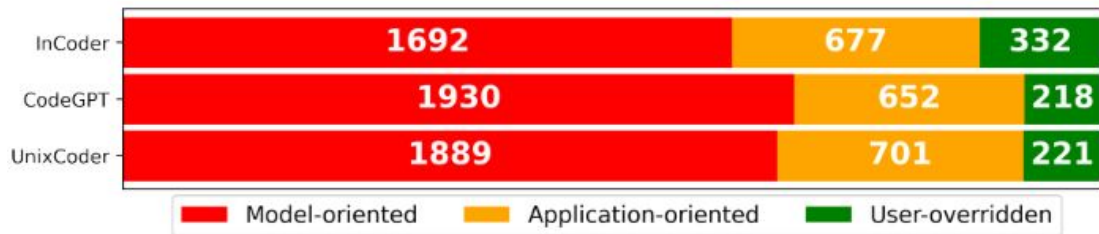
# 關鍵發現

InCoder 表現最佳, 尤其是 Python 和 Java 等主流語言

離線評估  $\neq$  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

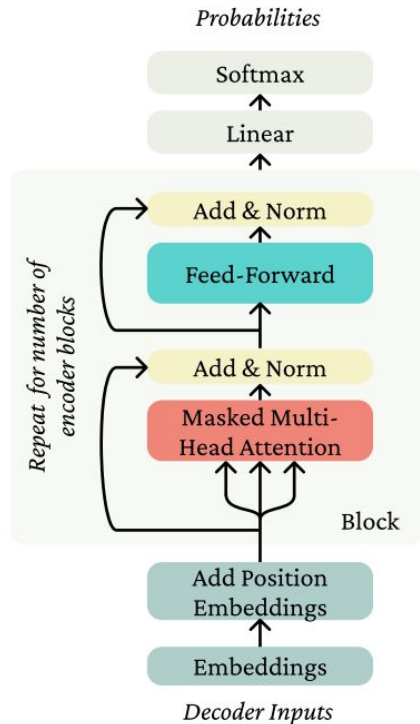
- 從 1950 年代的規則式翻譯，到 1990-2010 年代的統計式機器翻譯(SMT)，再到現今的神經機器翻譯(NMT)。
- SMT 依賴語言模型與翻譯模型的組合，但需要大量人工設計與特徵工程。

# 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	<ul style="list-style-type: none"><li>- Masked Language Modeling: 隨機遮蔽詞彙讓模型預測</li><li>- Next Sentence Prediction: 判斷句子間邏輯關係</li></ul>
下游任務應用	<ul style="list-style-type: none"><li>- 句子分類(情緒分析、垃圾郵件偵測)</li><li>- 問答系統(Q&amp;A)</li><li>- 命名實體識別(NER)</li></ul>
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

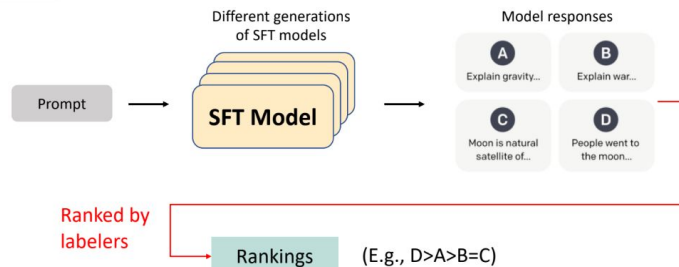
Supervised Fine-Tuning (SFT):

用人類標註資料微調模型。



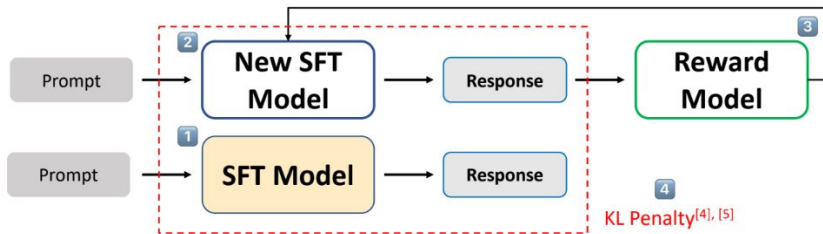
Reward Model Training:

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

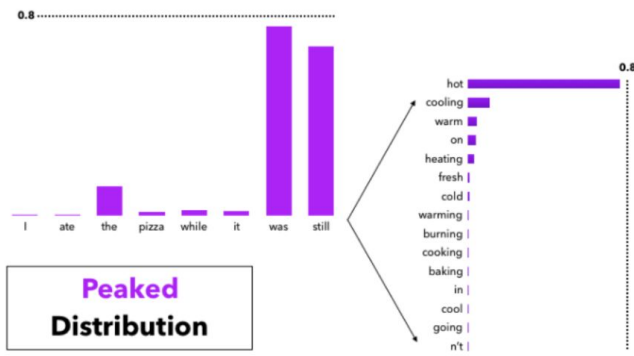
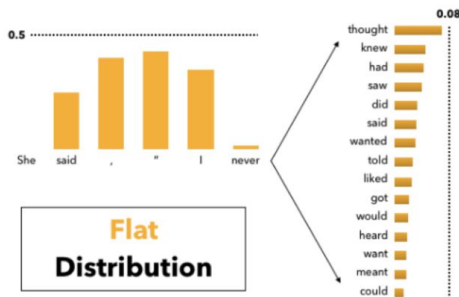
## 選擇題

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。

---

---

# Week 2

7/17-7/30

---

---

# 時間表 (7/17-7/30)

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

---

# 論文1

— Attention Is All You Need —  
(Transformer)(2017)

---

# 研究背景與動機

- 序列建模的挑戰：傳統的 RNN 和 LSTM 雖然能處理序列資料，但在長距離依賴(long-range dependencies)上表現不佳，且訓練效率低。
- 並行化困難：RNN 結構需要逐步處理序列，無法有效並行運算，限制了模型的擴展性。
- 目標：設計一種完全基於注意力機制的架構，去除 RNN，提升效率與表現。

# 模型架構與技術亮點

元件	功能與亮點
Self-Attention	每個詞都能與其他詞互相注意，捕捉全域語境，支援並行運算。
Multi-Head Attention	多個注意力頭可學習不同語意關係，提升表達能力。
Positional Encoding	為了保留序列順序資訊，加入位置編碼(使用 sin/cos 函數)。
Encoder-Decoder 架構	Encoder 處理輸入序列，Decoder 生成輸出序列，兩者透過 attention 橋接。
Layer Normalization & Residual Connection	穩定訓練過程，加速收斂。



# 實驗設計與成果

任務：

機器翻譯(英→德、英→法)，使用 WMT 2014 資料集。

比較基準：

與RNN-based Seq2Seq 模型(如 GNMT)比較。

成果：

在 BLEU 分數上超越所有先前模型。

訓練速度快上數倍，推論效率也更高。

模型參數雖多，但可高度並行化，適合 GPU 訓練。

# 關鍵概念解析

- Scaled Dot-Product Attention

核心公式: 
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

- Head 的意義

每個 head 學習不同語意關係，例如語法結構、主詞與動詞的連結等。

- 位置編碼的作用

Transformer 沒有遞迴結構，需額外加入位置資訊來保留序列順序。

# 學到什麼

注意力機制的力量：

只靠 attention 就能建構強大的語言模型，打破 RNN 的限制。

架構簡潔但強大：

Transformer 結構模組化，便於擴展與修改，成為後續 BERT、GPT 等模型的基礎。

效率與表現兼具：

不僅準確率高，還能大幅提升訓練與推論效率，改變 NLP 的技術路線

---

# 論文2

—— 基於深度學習的醫學文字  
探勘與序列標註任務(2024) ——

---

# 研究背景與動機

醫學文本的挑戰：

醫療紀錄、病歷、診斷報告等資料常以非結構化文字呈現，包含大量專業術語、縮寫與上下文依賴。

序列標註需求：

如疾病命名實體識別(NER)、症狀抽取、藥物關聯分析等任務，需精準標註文字中的關鍵資訊。

動機：

傳統規則式或機器學習方法在語意理解與泛化能力上有限，深度學習提供更強的語言建模能力，能有效提升醫學文字探勘的準確性與效率。

# 實驗設計與成果

資料來源：

可能使用電子病歷(EMR)、臨床筆記、PubMed 摘要等醫學語料。

任務設定：

如疾病名稱抽取、藥物-症狀關聯標註、醫學事件時間序列分析等。

成果亮點：

模型在多個序列標註任務上達到 SOTA(state-of-the-art)表現。

展現出深度模型在處理醫學語境、專業術語與上下文依賴上的優勢。

實驗結果具備臨床應用潛力，如輔助診斷、知識圖譜建構等。

# 關鍵概念解析

- 序列標註 (Sequence Labeling)

對文字序列中的每個 token 進行分類，如 BIO 標註法。

- NER (命名實體識別)

辨識疾病、藥物、症狀等醫學實體，為醫學 NLP 的核心任務之一。

- 上下文語意建模

利用深度模型捕捉語境，避免詞義模糊與誤判。

- 知識強化嵌入

將醫學知識庫資訊融入嵌入層，提升模型對專業術語的理解。

# 學到什麼

**深度學習能有效處理醫學語境：**

尤其在非結構化資料中，模型能自動學習語意與關聯。

**序列標註任務的多樣性與挑戰性：**

醫學文本標註不僅是語言問題，更牽涉到專業知識與臨床邏輯。

**跨領域整合的重要性：**

結合 NLP 技術與醫學知識，才能真正落地應用於臨床輔助與醫療資訊系統。

**模型選擇與資料品質密不可分：**

醫學語料的稀疏性與敏感性，對模型訓練提出更高要求。



---

# 論文3

— Language Models are  
Few-Shot Learners  
(GPT-3)(2020) —

---

# 研究背景與動機

傳統 NLP 模型的限制：

需針對每個任務進行微調 (fine-tuning)，需要大量標註資料。

難以快速適應新任務，與人類「看幾個例子就能學會」的能力差距甚大。

目標：

探討是否能透過「擴大語言模型規模」來實現類似人類的 few-shot 學習能力。

減少微調需求，讓模型能透過「上下文提示」直接執行任務。

# 模型架構與技術亮點

模組/技術	說明
GPT-3 架構	自回歸 Transformer 模型，使用 masked causal LM 訓練方式。
1750 億參數	比 GPT-2 大 10 倍，是當時最大規模的非稀疏語言模型。
純文本互動式任務指定	不進行微調，僅透過自然語言提示(prompt)指定任務與示例。
Few-shot / Zero-shot / One-shot	探討不同提示方式對模型表現的影響。

# 實驗設計與成果

任務範疇：

機器翻譯、問答、完形填空、常識推理、算術、新聞生成等。

設定方式：

不進行梯度更新或微調，僅透過 prompt 提供任務描述與少量示例。

成果亮點：

在多個 NLP 任務中，GPT-3 的 few-shot 表現接近甚至超越微調模型。

能生成高品質新聞文章，難以與人類撰寫區分。

在 SuperGLUE 等基準上，模型表現隨參數量與提示數量提升而顯著增強。

# 關鍵概念解析

- Few-shot Learning

僅提供少量示例(如 10–100)，模型即能執行新任務。

- In-Context Learning

模型在推論時透過上下文提示學習任務，不需參數更新。

- Prompt Engineering

設計有效的提示語句以引導模型正確執行任務，是使用 GPT-3 的關鍵技術。

- Scaling Laws

模型規模與表現之間存在可預測的關係，參數越多，泛化能力越強。

# 學到什麼

模型規模是關鍵：

擴大模型參數數量能顯著提升語言理解與生成能力。

語言模型具備元學習能力：

GPT-3 能在推論階段「即時學習」任務規則，類似人類的學習方式。

微調不再必要：

對許多任務而言，只需設計良好的 prompt 即可達到高效表現。

開啟新範式：

從「預訓練 + 微調」轉向「預訓練 + prompt 指導」，影響後續 InstructGPT、ChatGPT 等模型設計。

---

# 實作

— 中文GPT文本補全實驗 —

---

# 中文GPT文本補全實驗

我分別用了3個方法，從簡單逐步嘗試即時/多選擇，分別是這些方法：

- CLI 單次補全 (命令列)
- Gradio 即時網頁補全
- Flask API + CLI 補全迴圈

另外，用了4個模型作一些比較，模型分別為：

- 原始 GPT-2 中文
- ckiplab/gpt2-base-chinese
- IDEA-CCNL/Wenzhong-GPT2-110M
- uer/gpt2-chinese-cluecorpussmall

還有，作過一些微調，當中max\_length這個會比較具體



# 方法比較

方法編號	技術方式	優點	缺點
方法一	CLI 單次補全 (命令列)	<ul style="list-style-type: none"><li>- 寫法簡單、適合快速測試模型</li><li>- 支援多個句子生成</li><li>- 容易嵌入到其他流程中</li></ul>	<ul style="list-style-type: none"><li>- 無互動性, 只能一次性生成</li><li>- 無候選選項或信心分數</li><li>- 無法持續接續或擴展語境</li></ul>
方法二	Gradio 即時網頁補全	<ul style="list-style-type: none"><li>- 使用者介面直觀、易上手</li><li>- 即時補全、自動更新</li><li>- 可本地或遠端部署</li></ul>	<ul style="list-style-type: none"><li>- 僅產生單句, 互動性低</li><li>- 無選項選擇、無接句功能</li><li>- 輸出長度與控制不 夠彈性</li></ul>
方法三	Flask API + CLI 補全迴圈	<ul style="list-style-type: none"><li>- 有互動輸入、選擇與續句能力</li><li>- 顯示 log-prob 信心分數</li><li>- 可拓展成 Web API 或 GUI 工具</li></ul>	<ul style="list-style-type: none"><li>- 使用門檻稍高(需理解 CLI 流程)</li><li>- 沒有圖形界面(需手動建 GUI)</li><li>- 結果無法直接儲存或複製分享</li></ul>

# 模型比較

模型名稱	風格傾向	簡繁體支援	品質評估	備註
原始 GPT-2 中文	混亂、口語不通	不能穩定處理繁體，易出錯	*	偏實驗用途，非實際場景推薦
ckiplab/gpt2-base-chinese	清晰分詞、風格制式	強繁體支援，分詞較穩定	**	適合繁體任務，輸出比較偏向政治
IDEA-CCNL/Wenzhong-GPT2-110M	敘事感強，有創意但語法不穩定	偏簡體，繁體輸入會轉寫或失準	***	適合模擬口語或情境式生成
uer/gpt2-chinese-cluecorpussmall	口語自然、近網民語言	偏繁體，但輸出有時候會出現簡體	***	美食評論風格明顯，實用性佳

## max\_length 實測分析：

長度設定	輸出風格	流暢度	結尾中斷感
max_length = 20	通常僅一兩句，偏口語	中等（需良好接句）	結尾通常突然結束或缺詞
max_length = 50	可構成段落，有情境	高（語意連貫）	結尾略突兀

---

---

# Week 3

7/31-8/28

---

---

# 時間表 (7/31-8/27)

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

---

# 論文1

Improved Unsupervised Chinese Word  
Segmentation Using Pre-trained Knowledge  
and Pseudo-labeling Transfer  
(無監督中文斷詞)(2023)

---

# 研究背景與動機

中文斷詞的挑戰：

中文沒有空格作為詞界標記，斷詞是 NLP 的前置任務。

無監督方法不依賴人工標註，但準確率與效率常不理想。

既有方法的瓶頸：

使用 BERT probing(如 Perturb Masking)雖然有效，但推理成本高，每個 token 需多次 forward pass。

研究動機：

如何在不依賴人工標註的前提下，提升斷詞準確率並降低訓練成本。

結合 Segmental Language Model (SLM) 與 BERT 的優勢，提出更高效的 pseudo-labeling transfer 框架。

# 模型架構與技術亮點

模組	說明
<b>SLM (Segmental Language Model)</b>	無監督斷詞模型，從原始語料中學習 segment 結構。
<b>Pseudo-labeling Transfer</b>	將 SLM 產生的斷詞結果作為 pseudo labels，轉移給 BERT classifier。
<b>BERT Classifier</b>	將斷詞任務轉為序列標註任務，使用 BERT 進行分類。
<b>兩階段訓練架構</b>	第一階段訓練 SLM，第二階段使用 pseudo labels 訓練 BERT。



# 實驗設計與成果

## 資料集：

使用 8 個中文斷詞基準資料集 (如 PKU、MSR、AS、CityU 等)，涵蓋簡繁體語料。

## 評估指標：

使用 F-score 評估斷詞準確率。

## 成果亮點：

在 7 個資料集上達到 SOTA 表現。

相較於 Perturb Masking 方法，訓練時間顯著下降。

模型架構簡潔，易於部署與擴展。

# 關鍵概念解析

- **Unsupervised CWS**

不依賴人工標註的中文斷詞方法。

- **Segmental Language Model**

基於 segment 概率建模的語言模型，適合斷詞任務。

- **Pseudo-labeling**

使用模型預測結果作為標籤，進行自我訓練或知識轉移。

- **Knowledge Transfer**

將一模型的知識(SLM)轉移給另一模型(BERT)以提升表現。

# 學到什麼

## 無監督 $\neq$ 效能低：

透過架構設計與知識轉移，無監督斷詞也能達到 SOTA。

## 模型協同更勝單一模型：

SLM + BERT 的兩階段架構展現出互補優勢。

## 效率與準確率可以兼得：

避免高成本 probing 技術，改用 pseudo-labeling 提升效率。

## 繁體中文斷詞可受惠：

此方法適用於繁體語料，對你關注的 TW-GPT、Taiwan-LLM 等模型有直接啟發。

---

# 論文2

Enhance Content Selection for  
Multi-Document Summarization with  
Entailment Relation (蘊涵關係)(2020)

---

# 研究背景與動機

## 多文本摘要的挑戰：

多篇新聞報導常有資訊重疊(redundancy)與資訊差異(conflict)。

現有方法多以單篇角度處理，忽略跨文件間的語意關係。

## 動機：

希望透過語意蘊涵關係(entailment)來判斷哪些句子能被其他句子支持，進而選出更具代表性且一致性的內容。

提升摘要的資訊涵蓋率與語意一致性，避免矛盾或重複。

# 模型架構與技術亮點

模組/技術	說明
<b>Entailment-based Selection</b>	使用語意蘊涵模型判斷句子間的支持關係，選出被最多句子支持的句子作為摘要候選。
<b>Sentence Graph Construction</b>	建立句子間的蘊涵圖(entailment graph)，節點為句子，邊表示支持關係。
<b>Ranking Strategy</b>	根據句子被支持的程度進行排序，選出最具代表性的句子。
<b>融合傳統抽取式方法</b>	將蘊涵選句與 TF-IDF、LexRank 等方法結合，提升摘要多樣性與語意深度。

# 實驗設計與成果

## 資料集：

使用多篇新聞報導組成的多文本資料集，可能包含中文新聞語料。

## 評估方式：

使用 ROUGE 分數評估摘要品質。

人工評估語意一致性與資訊涵蓋率。

## 成果亮點：

引入蘊涵關係後，摘要在語意一致性上顯著提升。

相較於純抽取式方法，能更有效避免語意衝突與冗餘。

展現出語意推理在多文本摘要中的應用潛力。

# 關鍵概念解析

- **Entailment Relation**

若句子 A 能推論出句子 B, 則 A 蘊涵 B, 表示語意支持關係。

- **Multi-Document Summarization**

從多篇文件中選出代表性內容, 生成摘要。

- **Sentence Graph**

將句子視為節點, 語意支持關係為邊, 進行圖排序以選句。

- **Semantic Consistency**

摘要中句子間語意不衝突, 邏輯一致, 是高品質摘要的重要指標。



# 學到什麼

## 語意推理能強化摘要品質：

透過 entailment 判斷句子間支持關係，有助於選出更具代表性的內容。

## 跨文件語意整合是關鍵：

多文本摘要不只是抽取句子，更需要理解文件間的語意互動。

## 圖結構有助於內容選擇：

句子蘊涵圖提供了結構化的選句依據，提升摘要的邏輯性與涵蓋率。

## 中文語料也可受惠：

此方法具語言無關性，可應用於繁體中文新聞摘要任務。

---

# 論文3

CFEVER: A Chinese Fact Extraction  
and VERification Dataset  
(事實抽取與驗證)(2024)

---

# 研究背景與動機

## 事實查核的重要性：

在假訊息氾濫的時代，自動化事實驗證系統成為媒體與社群平台的關鍵工具。

## 中文資源的缺乏：

英文已有 FEVER 等資料集，但中文缺乏具規模、標註完整的事實驗證資料。

## 目標：

建立一個涵蓋多樣主題、具高標註品質的中文事實驗證資料集，促進中文 NLP 的發展與應用。

# 模型架構與技術亮點

技術模組	說明
<b>Claim Generation</b>	從中文維基百科人工撰寫 30,012 條主張 (claims)，涵蓋多領域知識。
<b>三分類標註</b>	每條主張標註為 Supports、Refutes 或 Not Enough Info。
<b>證據標註</b>	支持或反駁的主張皆附上證據句，可能來自單一或多個維基頁面。
<b>基準模型設計</b>	採用 FEVER pipeline: retrieval → sentence selection → claim verification。

# 實驗設計與成果

## 資料集規模：

30,012 條主張，標註一致性達 Fleiss' kappa 0.7934，顯示標註品質高。

## 基準實驗：

使用 FEVER pipeline 模型在 CFEVER 上進行測試。

結果顯示：現有英文模型在中文任務上表現不佳，凸顯 CFEVER 的挑戰性與必要性。

## 貢獻：

提供完整訓練、驗證、測試集。

建立 Leaderboard 平台 供社群提交模型結果。

# 關鍵概念解析

## Fact Verification

判斷主張是否為真，並提供支持或反駁的證據。

## Claim Classification

將主張分類為 Supports、Refutes、Not Enough Info。

## Evidence Retrieval

從維基百科中找出與主張相關的證據句。

## 中文維基語料處理

將維基頁面轉為 JSONL 格式，每句編號，方便模型定位與引用。

# 學到什麼

## 中文 NLP 邁向事實查核應用：

CFEVER 為中文語言模型提供了事實驗證的標準任務與資料。

## 資料品質決定模型上限：

高一致性標註與多樣主張類型使 CFEVER 成為嚴謹的 benchmark。

## 跨語言模型需語境調整：

英文 FEVER 模型無法直接遷移至中文，需針對語料與語意特性調整。

## 繁體中文潛力巨大：

此資料集雖以簡體為主，但架構可延伸至繁體語料，適合你打造繁中事實查核工具。

---

# 論文4

Exploring the Effectiveness of Pre-training  
Language Models with Incorporation of  
Diglossia for Hong Kong Content  
(雙言現象)(2025)

---



# 研究背景與動機

## Diglossia 現象：

香港語言環境中存在「高低語碼」的雙言結構——正式場合使用繁體中文(高語碼)，日常交流則以粵語(低語碼)為主。

## 語言模型挑戰：

現有中文語言模型多以書面語為主，忽略口語與地區變體，導致在香港語境下表現不佳。

## 研究動機：

探討 diglossia 對語言模型預訓練的影響。

建立首個香港語境的公開預訓練語料與模型。

推動香港本地語言技術的發展與社群參與。

# 模型架構與技術亮點

模組	說明
語料建構	自建香港語境語料，涵蓋繁體中文與粵語，來源包括新聞、論壇、社群媒體。
Diglossia 標註策略	將語料標註為高語碼（書面繁中）或低語碼（口語粵語），保留語碼混用特性。
預訓練架構	採用 ELECTRA 架構，具辨識能力強、訓練效率高的特性。
模型版本	採用 ELECTRA 架構，具辨識能力強、訓練效率高的特性。

# 實驗設計與成果

## 實驗任務：

文本分類(新聞 vs 社群)、情感分析(正面 vs 負面)、粵語理解(口語句子分類)

## 比較模型：

HKELECTRA(含 diglossia)、HKELECTRA-noDiglossia(移除粵語語料)、中文 RoBERTa、MacBERT(baseline)

## 成果亮點：

含 diglossia 的模型在所有任務上表現優於 baseline。

移除 diglossia 後，模型在粵語任務上顯著退化。

顯示語碼混用對香港語言模型的預訓練至關重要。

# 關鍵概念解析

## Diglossia

同一語言社群中並存兩種語碼：高語碼（正式書面語）與低語碼（口語方言）。

## ELECTRA

一種 discriminative 預訓練架構，透過判斷 token 是否被替換來學習語言表示。

## 語碼混用

在同一文本中混合使用繁體中文與粵語，是香港語境的常見語言現象。

## 語言資源建構

包括語料收集、語碼標註、清理與格式化，為模型預訓練提供高品質輸入。

# 學到什麼

## 地區語言特性不可忽略：

香港語境中的 diglossia 是語言模型表現的關鍵因素。

## 預訓練語料決定模型能力：

語碼混用語料能顯著提升模型在地區任務上的表現。

## 繁體中文與粵語需並重：

單一語碼模型無法涵蓋香港語言的多樣性。

## 社群參與與開源推動：

研究團隊已在 Hugging Face 開源模型與語料，鼓勵後續研究與應用。

---

# 論文 5

Unsupervised single document abstractive  
summarization using semantic units  
(語意單元)(2022)

---

# 研究背景與動機

## 摘要任務的挑戰：

抽象式摘要 (Abstractive Summarization) 需要模型生成新句子，難度遠高於擷取式摘要，且通常仰賴大量人工標註資料。

## 語意單元的潛力：

作者認為「語意單元」是文本中最具資訊價值的基本構成，若能自動識別其頻率與重要性，便可進行摘要生成。

## 研究動機：

打破對人工標註的依賴，探索無監督方法。

利用語意單元頻率作為摘要生成的依據。

提供低資源語言環境下的可行解法。

# 模型架構與技術亮點

模組	說明
語意單元抽取	利用 spaCy 等工具將文本切分為語意單元(如事件、主語-動詞-賓語結構)。
頻率估計機制	訓練模型自動學習每個語意單元在原文中的出現頻率，作為重要性指標。
句子篩選	根據語意單元頻率選出代表性句子，作為摘要生成的基礎。
摘要生成器	使用 Transformer-based 模型生成抽象式摘要，融合高頻語意單元。
兩階段訓練流程	第一階段學習語意單元頻率，第二階段進行摘要生成，完全無需人工標註。



# 實驗設計與成果

## 資料集：

CNN/Daily Mail(英文新聞摘要任務)，但方法可泛化至其他語言。

## 比較基準：

與其他無監督方法(如 Lead-3、TextRank)比較。

與部分小型預訓練模型(如 BART-small)比較。

## 成果亮點：

在 ROUGE-1/2/L 指標上超越其他無監督方法。

使用參數量遠少於大型預訓練模型，卻達到競爭性表現。

展現出在低資源環境下的實用性與可擴展性。

# 關鍵概念解析

## Semantic Units

文本中具語意的基本結構，如主謂賓、事件、名詞短語等，是摘要的核心資訊來源。

## Unsupervised Summarization

不依賴人工標註資料，透過自動化機制進行摘要生成。

## Frequency-based Selection

利用語意單元的出現頻率判斷其重要性，選出代表性句子。

## Abstractive vs Extractive

抽象式摘要會重新生成句子，擷取式摘要則直接取原文句子。

# 學到什麼

**語意單元是摘要的關鍵：**

比起單純句子選擇，語意單元提供更細緻的資訊粒度。

**無監督方法也能抽象生成：**

透過語意頻率與句子篩選，可在無標註情況下達成抽象式摘要。

**低資源語言的突破口：**

此方法不依賴大規模語料或預訓練模型，適合繁體中文、粵語等語言應用。

**模型設計需兼顧語言結構：**

語意單元的定義與抽取方式需根據語言特性調整，

對你設計繁中生成工具包非常有啟發。

---

# 論文 6

Improving multi-criteria Chinese word  
segmentation through learning sentence  
representation(多標準)(2023)

---

# 研究背景與動機

## 中文斷詞的多標準困境：

不同語料庫(如 PKU、MSR、AS)對斷詞標準定義不一，導致模型難以泛化。

## 現有方法的侷限：

多數模型依賴詞表或上下文特徵，但忽略整體語句語意，容易在 OOV(未見詞)上失效。

## 研究動機：

引入「句子表示學習」以捕捉語境語意，提升模型對不同斷詞標準的適應力。

探索無監督句子表示方法在斷詞任務中的潛力。

# 模型架構與技術亮點

模組	說明
<b>Sentence Representation Module</b>	使用多種 dropout mask 訓練句子表示，提升語意理解能力。
<b>Multi-Criteria Learning Framework</b>	將不同語料標準視為多任務學習，透過共享表示與特定分支達成泛化。
<b>Dropout-based Contrastive Learning</b>	利用不同 dropout 版本的句子進行對比學習，強化語意一致性。
<b>融合策略</b>	將句子表示嵌入至斷詞模型中，與 token-level 特徵共同作用。

# 實驗設計與成果

## 資料集：

九個中文斷詞語料庫(如 PKU、MSR、CityU、AS 等)，涵蓋多種標準。

## 比較模型：

Baseline: BERT、ZEN、CWS-BiLSTM。

本研究: 加入句子表示模組的改良版。

## 成果亮點：

在 6/9 語料庫上達成 F1 分數 SOTA。

在 8/9 語料庫上提升 OOV recall, 顯示對未見詞的強化能力。

實驗證明: 句子表示模組能顯著提升多標準斷詞的穩定性與泛化力。

# 關鍵概念解析

## Multi-Criteria CWS

中文斷詞任務中，不同語料庫有不同標準，需設計能同時學習多標準的模型。

## Sentence Representation

將整句語意轉為向量表示，捕捉語境與語義，有助於斷詞判斷。

## Dropout Contrastive Learning

利用 dropout mask 產生不同句子版本，進行對比學習以強化語意一致性。

## OOV Recall

模型在未見詞上的召回率，是衡量泛化能力的重要指標。



# 學到什麼

**語意理解是斷詞的關鍵：**

不只是字元層級的特徵，整句語意能顯著提升斷詞準確度。

**多標準學習需語境支撐：**

句子表示能作為不同標準間的橋樑，提升模型一致性。

**對比學習可強化語意表示：**

Dropout-based 對比學習是一種高效且無監督的語意強化方式。

**繁體中文斷詞潛力巨大：**

此方法可延伸至繁中語料，尤其適合你正在打造的繁中 NLP 工具包。

---

---

實作

— 中文文本補全實驗 —

---

---

# 中文文本補全實驗

我分別用了4個模型作一些比較, 模型分別為:

- WoBERT
- Taiwan-LLM
- RoFormer-Chinese
- ChineseRoBERTa
- ChatGLM3

還有, 作過README整理

# 模型比較

模型	輸入長度影響	自然度 / 重複情況	簡繁體支援	品質評估
ChatGLM3	效果佳	高自然度, 少重複	簡體佳, 繁體可用但較偏 簡體語料風格	★★★★
ChineseRoberta	短輸入可以, 長輸入佳	自然度中等, 短輸入易重複詞	簡體佳, 繁體需轉換	★★★
RoFormerChinese	短輸入差, 長輸入不佳	自然度低, 易出現符號重複	簡體佳, 繁體需轉換	★
Taiwan-LLM	效果佳	高自然度, 少重複	繁體佳, 簡體可用	★★★★ ★
WoBERT	短輸入差, 長輸入佳	自然度中等, 適合詞語補全	簡體佳, 繁體需轉換	★★

# 模型分析

模型	任務形式	分析
ChatGLM3	Chat式生成	指令式 Chat 模型，對 prompt 依從性高，適合需控格式與語氣的應用
ChineseRoberta	Masked LM	類 BERT 模型，對長上下文處理良好，短輸入語境不足時易詞語重複
RoFormerChinese	自回歸生成 (Rotary Position Embedding)	解碼時易陷入符號 loop，可透過調整 decoding 參數改善
Taiwan-LLM	Chat/生成式	對繁體中文有針對性優化，語料在地化程度高，對臺灣用語表現佳
WoBERT	Masked LM (BERT 架構)	Masked LM 架構，長文本詞彙預測精準，長句生成能力有限