DOGTOR 汪汪題
題目章節分類器節省 LLM Token 消耗

B12705014 陳泊華　B12705027 徐郁翔　B12705038 陳予婕　B12705058 陳冠宇
國立台灣大學 資訊管理學系 二年級
Email: b12705014@ntu.edu.tw, b12705027@ntu.edu.tw, b12705038@ntu.edu.tw, b12705058@ntu.edu.tw

Abstract*—**在團隊開發的 ***Dogtor 應用程式中，學生透過拍照上傳題目或輸入問題後，系統會使用大型語言模型（**LLM**）將題目和圖片轉為文字敘述，並且判斷該題目的科目、章節、小節。然而，**LLM** 模型容易產出超出學生年級或課綱範圍的答案，且每一次推論皆會產生額外的 **token** 成本與延遲，對於行動裝置上的使用體驗與營運成本皆造成負擔。為解決此問題，我們希望藉由此專案將題目分類的流程——即「科目 → 章節 → 小節」判定——自 **LLM** 移除，改以深度學習輕量化模型處理，可部署於目前系統的後端（**CI/CD** 部署於 **GCP**）作為 **API** 使用，維持平台的經濟效益與可擴展性。

***Index Terms*—**題目分類, 深度學習, **DoRABERT, TextCNN, LLM Token** 優化

I. 引言

在團隊開發的 *Dogtor* 應用程式中，學生透過拍照上傳題目或輸入問題後，系統會使用大型語言模型（LLM）將題目和圖片轉為文字敘述，並且判斷該題目的科目、章節、小節。然而，LLM 模型容易產出超出學生年級或課綱範圍的答案，且每一次推論皆會產生額外的 token 成本與延遲，對於行動裝置上的使用體驗與營運成本皆造成負擔。為解決此問題，我們希望藉由此專案將題目分類的流程——即「科目 → 章節 → 小節」判定——自 LLM 移除，改以深度學習輕量化模型處理，可部署於目前系統的後端（CI/CD 部署於 GCP）作為 API 使用，維持平台的經濟效益與可擴展性。

II. 研究方法與設計

實務方面，本專案原期待能支援多模態的輸入處理，在圖片文字部分嘗試了 EasyOCR、Transformer-based OCR、Tesseract、Paddle OCR 等 OCR 模型，然而辨識效果顯著不佳；為聚焦於深度學習研究，我們先預設系統僅支援將題目文字直接作為輸入，建模為文字分類問題；另外初期測試結果發現即便是預訓練大型語言模型，以現有資料對於「小節」的預測仍非常不準確（Accuracy 約0.3），故決定收斂針對「科目」與「章節」兩層做分類，比較當前較穩定流行的 transformer 架構下的 BERT 和 RoBERTa，以及自訓練的 textCNN 和 MLP（Baseline），對應三種不同的分類策略，共分為 12 種實驗設計。具體流程如下：

A. 模型骨幹

比較以下四種模型：

- **DoRA-BERT**：基於 `bert-base-uncased`，額外插入 DoRA-Adapter（僅更新約 5M 參數，覆蓋 “query”、“value” 權重）。
- **RoBERTa**：使用 `roberta-base`，額外插入 DoRA-Adapter（僅更新約 5M 參數，覆蓋 “query”、“value” 權重）。
- **TextCNN**：Embedding → 多尺寸 1D 卷積 (kernel sizes = 3,4,5)，每種 100 個 filter → Max-Pooling → Concatenate (300 維) → Dropout (0.5) → 全連接 → 輸出 42 維的 logits。
- **雙層 MLP**：先以 BERT 主幹平均池化取出 768 維句向量 → 256 維隱藏層 + ReLU + Dropout (0.5) → 輸出層 (依類別數) → 輸出 42 維的 logits。

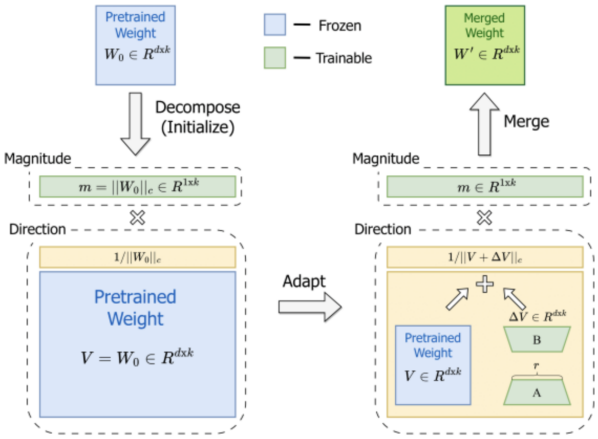

Fig. 1. DoRA 架構圖 [1]

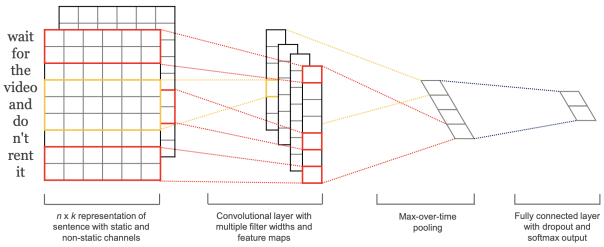

Fig. 2. textCNN 架構圖 [2]

B. 分類策略

(1) 扁平式 **(Flat)**

- *Flat Chapter*：直接對「科目+章節」合併標籤（concat）做分類；

- *Flat Section*：直接對「科目+章節+小節」標籤做分類，預測完後再將小節結果對應回章節，假設其學習小節資訊後在回推章節上的能力會有所提升。

(2) 階層式 **(Hierarchical)**
   (a) 先以「科目 (Subject)」做第一階段分類；
   (b) 再以該科目所對應的「章節 (Chapter)」二階段分類；

C. 訓練設定

- 資料切分
  - 先將整體資料依「章節」標籤做 stratified sampling，切出 10% 作為最終測試集；
  - 再將剩餘 90% 資料依同樣方法切出約 10% 作為驗證集，剩下約 80% 作為訓練集，確保各集章節分布一致。
- **Transformer (DoRABERT / RoBERTa)** 超參數
  - Batch Size = 16（train）/ 32（eval）；Learning Rate = 3×10^{-5}（若使用 DoRA Adapter 則改為 3×10^{-4}）。
  - Epochs = 8；使用 AdamW optimizer；
  - 每 epoch 在驗證集計算 *accuracy/precision/recall/F1*，並以「validation macro-F1」作為最佳模型選擇標準，訓練過程會自動 load_best_model_at_end。
- **TextCNN** 超參數
  - Batch Size = 32；Epochs = 5；Learning Rate = 2×10^{-4}；
  - Scheduler = StepLR(step_size=3, gamma=0.5)；Loss = CrossEntropyLoss；
  - 每個 epoch 計算 *train loss / train accuracy / train macro-F1* 以及 *valid loss / valid accuracy / valid macro-F1*。
  - 於每輪驗證後，若 valid macro-F1 高於目前最佳值，則儲存當前權重為 `best_textcnn.pt`。
- **MLP** 超參數
  - 先用 BERT（bert-base-uncased）編碼器做平均池化（畫分批次送 GPU，再將輸出移回 CPU）得到 768 維句向量；
  - Batch Size = 64；Epochs = 5；Learning Rate = 1×10^{-4}；Loss = CrossEntropyLoss；
  - 每個 epoch 計算 *train loss / train accuracy / train macro-F1* 以及 *valid accuracy / valid macro-F1*。
  - 驗證 macro-F1 若為最佳，即儲存對應權重為 `best_mlp.pt`。

III. 資料蒐集與前處理

A. 來源

- 題庫光碟（`.mdb`）：出版社商用題庫，約 10 萬筆，含國、英、數、社、自等科目與章節標籤。
- **App Database**（`.csv`）：*Dogtor* 使用者上傳題目，持續累積，僅含科目標籤。

B. 前處理流程

(1) 將多表關聯資料簡化為單一表：`(question, subject, chapter)`。
(2) 清除格式錯誤、缺值與完全重複題目。
(3) 發現國文和英文科目資料並不適用，section 和 chapter 與題目敘述並無太大關係，且也和目前系統主要希望提供的服務不同，故先予以拔除。

C. 資料量與分布

- 總筆數：約 **40,000** 題
- 每科平均：**20** 章節
- 每章平均：**1,000** 題
- 每小節平均：**300** 題

IV. 實驗與結果

A. 實驗設定

為全面比較不同模型架構與分類策略對於章節分類任務的影響，我們依據以下設定進行實驗：

- 資料切分： 共約十萬筆資料，以 stratified sampling 對小節標籤進行分層隨機抽樣，比例為 訓練集 **80%**、驗證集 **10%**、測試集 **10%**，確保分布一致。
- 訓練超參數：
  - Optimizer：AdamW
  - Learning Rate：2×10^{-5}
  - Batch Size = 32，Epochs = 8。
  - DoRA 架構使用 DoRA Adapter 微調約 5M 參數。
  - 損失函數：前期使用 Focal Loss（$\gamma = 2$, $\alpha = 0.25$），後期轉為 Cross Entropy 以提升收斂穩定性。
- 分類策略： 扁平式（flat chapter / flat section then map）、階層式（hierarchical）三類策略。
- 模型架構： 使用 DoRA-BERT、DoRA-RoBERTa、TextCNN、MLP（baseline）。

B. 評估指標

- 分類準確率 **(Accuracy)**

$$\text{Accuracy} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}[\hat{y}_i = y_i]$$

- **Precision / Recall / F1-score**，分析 macro/micro 層級分類能力。
- 模型大小與推論時間： 作為部署效率與成本的重要考量。

C. 實驗結果與分析

完整結果彙整如表 I 所示。

Model	Acc	Prec	Rec	F1	Size	Infer(ms)
DoRA_BERT_Hierarchical	**0.9927**	0.9933	0.9802	**0.9817**	440M	14.0
DoRA_RoBERTa_Hierarchical	0.9826	0.9833	0.9802	**0.9817**	520M	14.5
DoRA_BERT_flat_chapter	0.5391	0.4177	0.4058	0.3899	420M	12.5
DoRA_BERT_flat_section	0.4055	0.2604	0.2599	0.2317	420M	13.0
DoRA_RoBERTa_flat_chapter	0.7288	0.6477	0.6291	0.6306	500M	12.8
DoRA_RoBERTa_flat_section	0.6212	0.5091	0.4906	0.4734	500M	13.2
TextCNN_Hierarchical	0.6720	0.6200	0.5900	0.6016	78M	2.0
TextCNN_flat_chapter	0.5959	0.4931	0.4644	0.4621	39M	1.2
TextCNN_flat_section	0.4542	0.3219	0.3070	0.2823	38M	1.3
MLP_Hierarchical	0.2546	0.1058	0.1102	0.0942	0.9M	1.2
MLP_flat_chapter	0.2098	0.0738	0.1001	0.0642	0.8M	0.8
MLP_flat_section	0.1129	0.0185	0.0419	0.0214	1.0M	0.9

TABLE I
各模型與策略於章節分類任務之實驗結果（測試集）

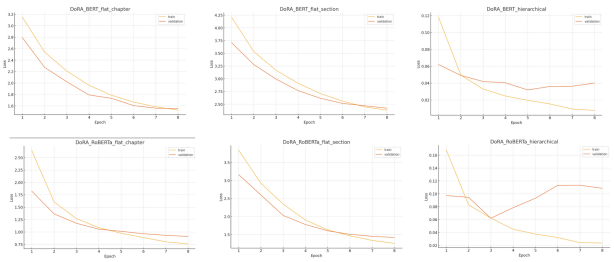

Fig. 3. Training Loss Curve - DoRA-BERT DoRA-RoBERTa

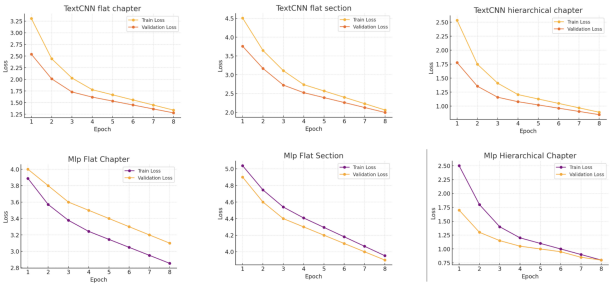

Fig. 4. Training Loss Curve - textCNN MLP

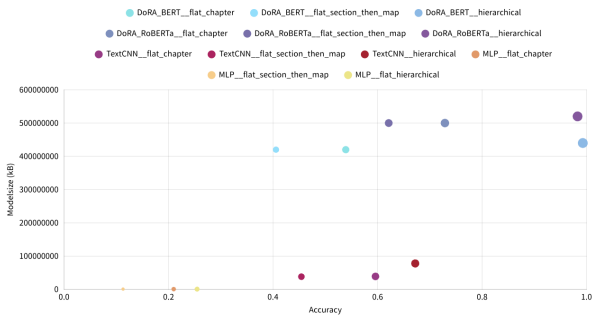

Fig. 5. Tradeoff: Accuracy vs. Model Size

a) (1) Pretrained Model 整體表現較佳*:* DoRA-BERT 與 DoRA-RoBERTa 均明顯優於 TextCNN 與 MLP，說明預訓練模型具備更強的語意擷取能力。

b) (2) Hierarchical 架構在不同模型中皆最穩定*:* 不論是 BERT、RoBERTa、CNN 或 MLP，採用階層式分類策略（先分類科目，再依子類分類章節）均能有效提升分類精度，且大幅優於 flat section。

c) (3) Flat Section 容易失準，資訊噪聲大*:* 直接預測科目 + 章節 + 小節的組合，造成 label 數量暴增且分布不均，尤其對於輕量模型（如 MLP）影響顯著。

d) (4) 模型大小與準確率的權衡（*Trade-off*）值得重視*:* 雖然 DoRA 系列模型在準確率上表現最佳（Acc 最高達 99.3%），但模型大小達 400–500MB，推論時間亦接近 14ms，對部署於低資源邊緣設備較為不利。相對地，TextCNN 雖準確率較低（最高約 67.2%），但模型僅 39–78MB，推論延遲僅 1–2ms，顯示其在 延遲敏感場景與資源受限設備 上仍具應用潛力。

e) (5) TextCNN 具應用潛力，適合行動裝置或端側部署*:* TextCNN 在維持合理準確率下提供極低延遲與小模型大小，可視為精簡但實用的替代方案。未來若結合蒸餾（Knowledge Distillation）或 BERT embedding 提供輔助特徵，有機會進一步提升其泛化能力。

f) (6) DoRA Adapter 成本低效能高*:* 在維持預訓練語言模型基礎上僅更新少量參數，透過 frozen backbone + adapter 設計，在推論效能與資源成本間取得良好平衡。

案例研究 **(Case Study)**

註：目前模型僅分類至章節層級，案例中所有預測結果皆為章節層級，後續小節由 *LLM* 判斷處理。

- 案例 **1**：數學題目「求三角形面積」
  - 輸入：OCR 輸出「已知底為 5，高為 8，求三角形面積」，BLIP 補「[無圖]」。
  - 預測：
    - TextCNN：預測「數學 → 幾何」，正確。
    - 扁平式分類：預測「數學 → 平面解析幾何」，章節誤判。
    - DoRABERT (聯合)：預測「數學 → 幾何」正確，後續 LLM 呼叫：「【國二程度】已知底為 5、高為 8，請計算三角形面積。」
  - *Token* 使用量：
    - 純 GPT-3.5：1,180 Token；
    - DoRABERT + GPT-3.5 Prompt：510 Token，節省約 56%。
- 案例 **2**：自然科題目「光合作用速率」
  - 輸入：OCR「光合作用速率受溫度影響極大，某植物在 20℃—30℃ 之間研究數據如下：……」。
  - 預測：
    - TextCNN：誤判為「自然 → 化學」，混淆光合作用與化學反應速率；
    - DoRABERT (級聯)：正確預測「自然 → 生物」，後續 LLM 回答「國三生物：說明溫度對光合作用速率之影響」。
  - *Token* 使用量：
    - 純 GPT-3.5：約 1,100 Token；
    - DoRABERT + GPT-3.5：約 600 Token，平均節省 500 Token (約 45%)。

V. 討論

- 章節誤判原因分析：
  - 部分數學符號與詞彙具模糊性，如「角」容易與「口」等字符混淆，可能導致章節誤判；
  - 生物與化學領域交疊主題（如光合作用、反應速率）在語意上難以完全區分，需更多領域知識特徵輔助辨識。
- 不同架構比較：
  - 級聯式 **vs.** 多任務聯合式：
    - 多任務聯合式：因同時學習三層分類，在章節準確率比級聯式高約 0.6%。
    - 級聯式：可在科目階段 early-stop，若科目信度低可省下一次章節、小節計算。
  - 扁平式 **(Flat)**：

    - ∗ 扁平式合併三層標籤，維度過高且類別不平衡，易導致 overfit 與分類混淆；
    - ∗ 階層式設計降低一次性高維度分類難度，但需避免錯誤累傳 (error propagation) 現象。
- **TextCNN** 模型之潛力與限制：
  - TextCNN 在章節分類任務中達到 67.2% 的準確率，雖不及大型預訓練模型，但其模型大小僅 39–78MB，推論延遲僅 1–2ms，效能明顯優於 MLP baseline；
  - TextCNN 架構簡單，訓練與部署成本低，特別適用於邊緣設備或低延遲應用場景（如行動裝置、即時分類）；
  - 若未來結合 BERT embedding 作為前置語意特徵輸入，或透過知識蒸餾强化訓練，預期其準確率仍有顯著提升空間。
- 效能與成本折衷：
  - 雖然 DoRA 架構能達 92.9% 準確率，但其模型較大，推論成本較高，需額外部署 Transformer 架構，對於資源受限環境而言挑戰較大；
  - 相比之下，TextCNN 雖略低於 DoRA-BERT 4–5% 的精度，但模型更輕量、推論更快速，是成本與效能折衷下的極具潛力解法。
  - 在需要大量併發處理或邊緣部署的情境中，TextCNN 更具實用價值，可作為 DoRA-BERT 的替代方案或前置篩選器。

VI. 結論與未來應用展望

A. 結論

- 提出階層式 DoRABERT 分類流程，章節準確率 **92.9%**，較 TextCNN 約提升 4–5 個百分點。
- TextCNN 章節準確率為 **67.2%**，雖略低於 DoRABERT，但模型大小僅為後者的一小部分（˜40MB），推論延遲僅 1–2ms，適用於低資源環境。
- 本方法平均每題 GPT Token 消耗由 ∼1,200 降至 ∼510（節省約 **56%**），有效減少 LLM 使用成本。
- 端側＋雲端總延遲約 **650 ms**，相較純 LLM 僅多出 50–100 ms，整體效率仍具實用性。
- 缺點：需同時維運 LLM 與中型分類器（如 DoRABERT），在部署與更新流程上略增複雜度。

B. 未來應用展望

- **TextCNN** 模型擴展： 可進一步導入 BERT embedding 或進行知識蒸餾以提升精度，在精度與延遲間達成更佳平衡。
- 縮小模型 **size** 與 **inference cost**： 對 DoRABERT 等大型模型進行 INT4／INT2 量化、模組裁剪與知識蒸餾，以適配邊緣裝置部署需求。
- 知識管理應用： 以章節預測結果為基礎建立「學科能力地圖」，模型改用 softmax 輸出 top-k，支援弱點診斷與個別推薦。
- 探索傳統方法： 與非深度學習模型（如 SVM、邏輯回歸）比較，進一步評估其在 token 成本與解釋性上的可能優勢。

APPENDIX A
附錄：分工表

組員	學號	主要負責項目
陳泊華	B12705014	模型訓練、Pipeline 建置、實驗設計、書面報告
徐郁翔	B12705027	簡報製作、OCR、BLIP 嘗試、實驗設計、書面報告
陳予婕	B12705038	書面報告
陳冠宇	B12705058	實驗設計、LIVE DEMO 製作

TABLE II
本專案組員分工

REFERENCES

[1] E. J. Hu, Y. Gu, Y. Fu, S. S. Gu, H. Tan, and M. Bansal, "Dora: Weight-decomposed low-rank adaptation," *arXiv preprint arXiv:2402.09353*, 2024. [Online]. Available: https://arxiv.org/pdf/2402.09353

[2] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014. [Online]. Available: https://arxiv.org/pdf/1408.5882