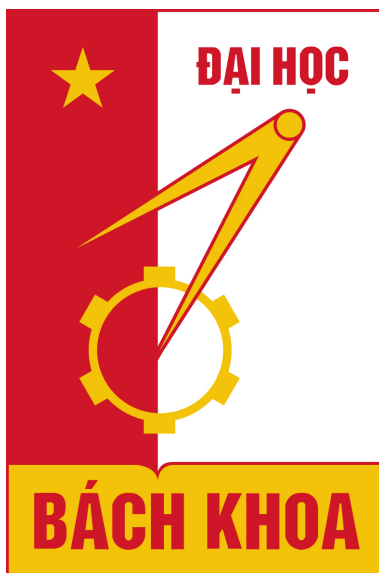


**ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**  
— o0o —



**A Two-Stage LoRA-to-Full Fine-tuning Approach for  
Vietnamese Legal Small Language Models**

**Project 1**

**Giảng viên hướng dẫn:** Bùi Trọng Đức  
**Sinh viên thực hiện:** Võ Hữu Trí Dũng  
**MSSV:** 20235057  
**Ngành:** Khoa học Máy tính K68

**Hà Nội - 2026**

# Mục lục

<b>Lời cảm ơn</b>	<b>3</b>
<b>Tóm tắt</b>	<b>4</b>
<b>1 Giới thiệu</b>	<b>5</b>
<b>2 Các công trình liên quan</b>	<b>7</b>
2.1 Tinh chỉnh tiết kiệm tham số (Parameter-Efficient Fine-Tuning) . . . .	7
2.2 Hợp nhất các bộ điều hợp (Adapter Merging) . . . . .	7
<b>3 Phương pháp</b>	<b>9</b>
3.1 Giai đoạn 1: Huấn luyện LoRA theo từng tác vụ . . . . .	9
3.2 Giai đoạn 2: Huấn luyện toàn bộ tham số trong thiết lập đa tác vụ . .	12
<b>4 Thực nghiệm và Kết quả</b>	<b>14</b>
4.1 Thiết lập thực nghiệm . . . . .	14
4.1.1 Tác vụ pháp lí . . . . .	14
4.1.2 Dữ liệu . . . . .	15
4.1.3 Độ đo đánh giá . . . . .	16
4.2 Kết quả . . . . .	17
<b>5 Kết luận và Hướng phát triển</b>	<b>19</b>
<b>Phụ lục</b>	<b>22</b>

# LỜI CẢM ƠN

Báo cáo này là kết quả của quá trình tự nghiên cứu, triển khai và thử nghiệm trong khuôn khổ đề án Project 1. Đề tài của em được xây dựng dựa trên ý tưởng và cảm hứng từ công trình **MinLegal at VLSP2025-LegalSLM: A Two-Stage LoRA-to-Full Fine-tuning Approach for Vietnamese Legal Small Language Models** [7]. Bài báo này đã mang đến cho em một góc nhìn thực tiễn về việc áp dụng các kỹ thuật fine-tuning hiện đại cho các mô hình ngôn ngữ kích thước nhỏ trong lĩnh vực pháp lý tiếng Việt, từ đó giúp định hướng và là nền tảng khoa học cho đề án.

Em xin gửi lời cảm ơn chân thành đến nhóm tác giả của MinLegal vì những đóng góp quan trọng trong cộng đồng NLP Việt Nam, cũng như vì nguồn cảm hứng để em phát triển phiên bản bài làm phù hợp với phạm vi học phần.

Em xin chân thành cảm ơn thầy **Bùi Trọng Đức** – giảng viên phụ trách học phần Project 1, đã tạo điều kiện và định hướng học thuật để em thực hiện đề tài này. Những buổi học và góp ý từ thầy là nguồn động lực quan trọng giúp em hoàn thiện báo cáo và triển khai huấn luyện mô hình hiệu quả.

Trong quá trình thực hiện, em đã cố gắng hoàn thiện báo cáo một cách nghiêm túc và đầy đủ nhất. Tuy nhiên, do giới hạn về thời gian và kinh nghiệm, báo cáo chắc chắn vẫn còn tồn tại những thiếu sót. Em rất mong nhận được sự góp ý từ thầy để hoàn thiện hơn trong tương lai.

Em xin chân thành cảm ơn!

# TÓM TẮT

Việc xây dựng các mô hình ngôn ngữ kích thước nhỏ đến trung bình ( $\leq 4B$  tham số) cho tác vụ xử lý văn bản tiếng Việt là một thách thức do hạn chế về dữ liệu huấn luyện và tài nguyên tính toán. Trong đồ án này, em áp dụng một quy trình huấn luyện hai giai đoạn kết hợp điều chỉnh LoRA theo từng tác vụ riêng biệt với giai đoạn tinh chỉnh toàn bộ mô hình sau đó. Về bộ dữ liệu huấn luyện, em sử dụng 3,400 mẫu thuộc ba tác vụ pháp lý khác nhau thông qua kỹ thuật tạo dữ liệu dựa trên cơ chế lấy mẫu, được nhóm tác giả công khai cùng với bài báo<sup>1</sup>. Giai đoạn thứ nhất trong phương pháp tập trung huấn luyện các bộ điều hợp (LoRA adapters) theo từng tác vụ cụ thể, và sau đó được hợp nhất vào một mô hình. Tiếp theo, giai đoạn thứ hai tiến hành fine-tuning toàn bộ tham số mô hình trên tất cả mẫu huấn luyện. Kết quả thực nghiệm cho thấy phương pháp hai giai đoạn mang lại hiệu quả vượt trội so với các cách tiếp cận khác như LoRA-only, QLoRA-only hay full fine-tuning. Với phiên bản thử nghiệm tốt nhất, mô hình 2-stage đạt kết quả với điểm trung bình 78.88, cao hơn đáng kể so với full fine-tuning (73.92) và LoRA-only (74.28). Những kết quả này cho thấy chiến lược kết hợp LoRA và full fine-tuning là một hướng tiếp cận hiệu quả và thực tiễn cho việc huấn luyện mô hình ngôn ngữ nhỏ trong bối cảnh tài nguyên hạn chế, đồng thời chứng minh tính khả thi của phương pháp trong phạm vi đồ án học phần. Mã nguồn và dữ liệu được công khai tại: <https://github.com/vohuutridung/IT3150>.

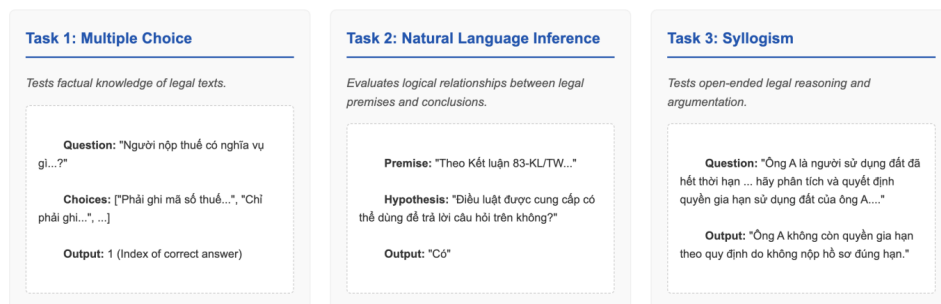
---

<sup>1</sup><https://huggingface.co/datasets/luanngo/Vietnamese-Legal-Chat-Dataset>

# Chương 1

## Giới thiệu

Xử lý văn bản pháp lý là một trong những ứng dụng đầy thách thức trong lĩnh vực Xử lý ngôn ngữ tự nhiên (NLP), đòi hỏi mô hình phải hiểu được các khái niệm ngôn ngữ phức tạp, ngữ cảnh thực tế và khả năng suy luận theo từng tác vụ cụ thể [1]. Trong khi các mô hình ngôn ngữ lớn (LLM) đã thể hiện hiệu năng vượt trội ở những ngôn ngữ như tiếng Anh, việc xây dựng các mô hình hiệu quả cho tiếng Việt vẫn còn bị hạn chế đáng kể bởi độ phức tạp của ngôn ngữ, sự mơ hồ trong phân tách từ, cấu trúc câu linh hoạt và mối quan hệ ẩn giữa các khái niệm pháp lý [8][4]. Thử thách VLSP2025 về các mô hình ngôn ngữ nhỏ cho lĩnh vực pháp lý tiếng Việt<sup>1</sup> được xây dựng nhằm giải quyết những nhu cầu này, tập trung vào các mô hình có kích thước  $\leq 4B$  tham số và ba nhiệm vụ đánh giá cốt lõi: phân loại mức độ hữu ích của trích dẫn pháp lý **NLI**, trả lời câu hỏi trắc nghiệm pháp lý **MCQ**, và lập luận pháp lý dạng văn bản tự do dựa trên suy luận tam đoạn luận **SQA**. Hình 1.1 minh họa ba nhiệm vụ lập luận pháp lý này, thể hiện rõ mức độ phức tạp và đa dạng của bộ khung đánh giá.



Hình 1.1: Tổng quan về ba tác vụ pháp lý trong bộ dữ liệu VLSP2025 LegalSLM

Việc phát triển các hệ thống xử lý ngôn ngữ tự nhiên (NLP) cho lĩnh vực pháp lý tiếng Việt đang đối mặt với nhiều thách thức quan trọng. Thứ nhất, sự khan hiếm các bộ dữ liệu pháp lý tiếng Việt được gán nhãn chất lượng cao đã hạn chế đáng kể quá trình phát triển mô hình, khi các nguồn dữ liệu hiện có vẫn nhỏ hơn rất nhiều so

<sup>1</sup><https://vlsp.org.vn/vlsp2025/eval/legalslm>

với các bộ dữ liệu tương ứng trong tiếng Anh. Thứ hai, các ràng buộc về tính toán trong nhiều tổ chức tại Việt Nam đòi hỏi việc xây dựng các mô hình hiệu quả, có thể vận hành trong điều kiện tài nguyên hạn chế nhưng vẫn duy trì hiệu năng cạnh tranh. Thứ ba, các kỹ thuật tinh chỉnh tiết kiệm tham số hiện có, chẳng hạn như Low-Rank Adaptation (LoRA), vẫn chưa khai thác đầy đủ tiềm năng của việc kết hợp nhiều chiến lược huấn luyện nhằm tối ưu hóa khả năng của các mô hình nhỏ trong những miền chuyên biệt [3][2]. Các phương pháp hiện nay thường áp dụng một phương pháp huấn luyện đồng nhất cho nhiều dạng nhiệm vụ khác nhau, mà chưa xét đến đặc điểm và yêu cầu riêng biệt của từng kịch bản pháp lý.

Đề án này trình bày một khung huấn luyện hai giai đoạn kết hợp huấn luyện LoRA cho từng tác vụ riêng biệt và huấn luyện toàn bộ mô hình cho các mô hình ngôn ngữ trong bài toán pháp lý tiếng Việt. Trước tiên, em huấn luyện từng bộ điều hợp LoRA riêng biệt cho từng tác vụ, sau đó hợp nhất chúng và thực hiện huấn luyện toàn bộ tham số trên tất cả mẫu dữ liệu. Về dữ liệu huấn luyện, em sử dụng bộ dữ liệu được nhóm tác giả của bài báo MinLegal [7] công khai.

Đóng góp chính của đề án là **khung huấn luyện hai giai đoạn** được đề cập ở trên. Em kết hợp tinh chỉnh LoRA chuyên biệt theo từng tác vụ, sau đó tiến hành tinh chỉnh toàn bộ tham số, qua đó đạt được hiệu năng vượt trội so với các phương pháp huấn luyện truyền thống.

Cấu trúc báo cáo được trình bày như sau: Chương 2 trình bày các công trình liên quan về kỹ thuật tinh chỉnh tiết kiệm tham số. Chương 3 mô tả phương pháp huấn luyện hai giai đoạn. Chương 4 trình bày thiết lập thí nghiệm và phân tích kết quả. Cuối cùng, chương 5 đưa ra kết luận và các hướng phát triển trong tương lai.

# Chương 2

## Các công trình liên quan

### 2.1 Tinh chỉnh tiết kiệm tham số (Parameter-Efficient Fine-Tuning)

Tinh chỉnh tiết kiệm tham số (Parameter-Efficient Fine-Tuning) đã nổi lên như một kỹ thuật then chốt trong việc thích ứng các mô hình ngôn ngữ lớn với các miền chuyên biệt trong khi vẫn đảm bảo hiệu quả tính toán. Trong đó, Low-Rank Adaptation (LoRA) [3] là phương pháp tiêu biểu, cho phép tinh chỉnh mô hình bằng cách giữ cố định toàn bộ trọng số gốc và chỉ học thêm các ma trận hạng thấp để biểu diễn phần cập nhật trọng số. Cụ thể, thay vì cập nhật trực tiếp ma trận trọng số  $W$ , LoRA mô hình hóa phần thay đổi dưới dạng  $\Delta W = BA$ , trong đó  $B$  và  $A$  là các ma trận có hạng thấp với số tham số rất nhỏ so với  $W$ . Nhờ cơ chế này, LoRA giảm đáng kể số lượng tham số cần huấn luyện, tiết kiệm bộ nhớ và chi phí tính toán, đồng thời vẫn duy trì khả năng học các biểu diễn phù hợp cho từng tác vụ hoặc miền dữ liệu cụ thể.

Trên nền tảng LoRA, nhiều phương pháp mở rộng đã được đề xuất nhằm nâng cao hiệu quả huấn luyện và chất lượng suy luận. QLoRA [2] cho phép tinh chỉnh các mô hình lớn thông qua lượng tử hóa 4-bit; DoRA [6], trong đó trọng số được phân tách thành các thành phần độ lớn và hướng, giúp cải thiện tới 3,7 điểm trên các tác vụ suy luận; và AdaLoRA [12], sử dụng cơ chế phân bổ tham số động dựa trên mức độ quan trọng của từng tầng. Các nghiên cứu gần đây chỉ ra rằng các kỹ thuật tinh chỉnh tiết kiệm tham số có thể đạt hiệu năng tương đương với tinh chỉnh toàn phần, trong khi chỉ sử dụng khoảng 0.1% số tham số của mô hình trong bài toán thích ứng miền pháp lý [5].

### 2.2 Hợp nhất các bộ điều hợp (Adapter Merging)

Sự phát triển của các phương pháp fine-tuning hiệu quả tham số, đặc biệt là LoRA [3], cho phép huấn luyện nhiều bộ điều hợp (adapter) chuyên biệt cho từng nhiệm vụ hoặc miền dữ liệu khác nhau trên cùng một mô hình nền. Tuy nhiên, trong các kịch

bản đa nhiệm hoặc khi cần tích hợp tri thức từ nhiều tác vụ, việc sử dụng riêng lẻ từng adapter trở nên kém hiệu quả. Do đó, bài toán hợp nhất các bộ điều hợp (adapter merging) đã thu hút sự quan tâm trong các nghiên cứu gần đây. Các phương pháp hợp nhất adapter thường tập trung vào việc kết hợp trực tiếp các tham số LoRA đã được huấn luyện, thông qua các kỹ thuật như trung bình tuyến tính, kết hợp có trọng số, hoặc các phương pháp dựa trên phân rã ma trận (SVD). Một số hướng tiếp cận nâng cao hơn như TIES [10] hay DARE [11] được đề xuất nhằm giảm thiểu xung đột giữa các adapter khi chúng được huấn luyện trên các nhiệm vụ khác nhau. Ưu điểm chung của các phương pháp này là không yêu cầu huấn luyện lại toàn bộ mô hình, giúp tiết kiệm đáng kể chi phí tính toán và phù hợp với các mô hình kích thước nhỏ.

Tuy nhiên, nhiều nghiên cứu cho thấy việc chỉ hợp nhất các adapter mà không có bước tinh chỉnh bổ sung có thể dẫn đến suy giảm hiệu năng, đặc biệt trong các tác vụ yêu cầu suy luận phức tạp hoặc khi dữ liệu giữa các nhiệm vụ có sự khác biệt lớn. Các tham số sau khi hợp nhất có thể chứa nhiễu hoặc xung đột tri thức, làm hạn chế khả năng khái quát hóa của mô hình. Trong bối cảnh đó, đề án này kế thừa ý tưởng hợp nhất adapter nhưng kết hợp với một giai đoạn fine-tuning toàn phần sau khi hợp nhất, nhằm cho phép mô hình điều chỉnh lại toàn bộ không gian tham số và khai thác hiệu quả tri thức từ các adapter chuyên biệt. Cách tiếp cận này giúp khắc phục các hạn chế của adapter merging truyền thống và phù hợp với mục tiêu huấn luyện mô hình ngôn ngữ nhỏ trong điều kiện dữ liệu và tài nguyên tính toán hạn chế.

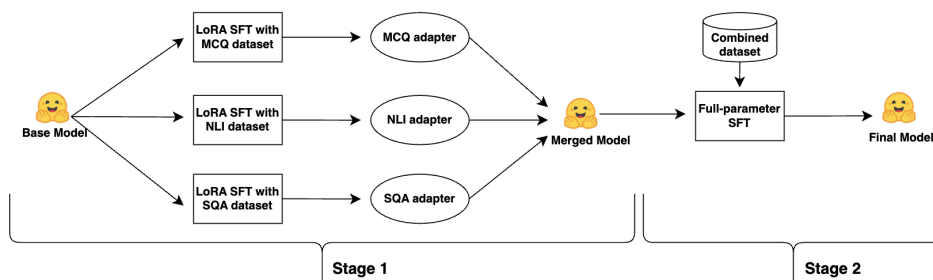


# Chương 3

## Phương pháp

Trong phần này, em trình bày về hướng tiếp cận để giải quyết các bài toán suy luận trên miền dữ liệu pháp lý tiếng Việt, với trọng tâm là kiến trúc huấn luyện hai giai đoạn.

Kết quả thực nghiệm chỉ ra rằng hướng tiếp cận huấn luyện hai bước vượt trội rõ rệt so với các phương pháp tinh chỉnh truyền thống. Cách tiếp cận này giúp giải quyết bài toán phân bố dữ liệu không đồng đều giữa các tác vụ, đồng thời tối đa hóa hiệu quả sử dụng tham số của mô hình.



Hình 3.1: Tổng quan về phương pháp huấn luyện hai giai đoạn được đề xuất trong báo cáo.

### 3.1 Giai đoạn 1: Huấn luyện LoRA theo từng tác vụ

Trong giai đoạn đầu tiên, em tiến hành huấn luyện các bộ điều hợp LoRA riêng biệt cho từng tác vụ lập luận pháp lý. Cụ thể, ba bộ điều hợp LoRA tương ứng với ba tác vụ bao gồm: trả lời câu hỏi trắc nghiệm pháp lý (MCQ), suy luận ngôn ngữ tự nhiên (NLI), và lập luận pháp lý dựa trên tam đoạn luận (SQA). Mỗi bộ điều hợp được huấn luyện trên tập dữ liệu tương ứng với từng tác vụ, nhằm khai thác tối đa các đặc trưng chuyên biệt của từng dạng bài toán. Cách tiếp cận này cho phép mô hình học được các

mẫu lập luận đặc thù của từng nhiệm vụ, đồng thời tránh hiện tượng nhiễu giữa các tác vụ có bản chất khác nhau.

Do sự phân bố dữ liệu không đồng đều giữa các tác vụ, mỗi bộ điều hợp LoRA đạt hiệu năng tối ưu tại các thời điểm huấn luyện khác nhau. Vì vậy, em tiến hành lựa chọn checkpoint tốt nhất cho từng bộ điều hợp dựa trên hiệu năng đánh giá trên tập phát triển (validation set), thay vì sử dụng chung một số bước huấn luyện cố định cho tất cả các tác vụ.

Sau khi hoàn tất huấn luyện và lựa chọn các checkpoint tối ưu, ba bộ điều hợp LoRA được hợp nhất với mục tiêu xây dựng một cập nhật hợp nhất  $\Delta W^m$  phản ánh tri thức thu được từ tất cả các tác vụ huấn luyện, đồng thời duy trì tính ổn định trong quá trình tối ưu và hiệu quả về số lượng tham số. Cập nhật này đóng vai trò như một biểu diễn chung, cho phép mô hình tích hợp kiến thức đa tác vụ mà không làm suy giảm đáng kể hiệu năng trên từng tác vụ riêng lẻ. Phương pháp được sử dụng là `ties_svd` kết hợp 2 cơ chế bổ trợ lẫn nhau - TIES [10] và SVD-based low-rank approximation [9].

Cụ thể, mỗi bộ điều hợp LoRA  $i$  bao gồm một tập hợp các cập nhật trọng số dạng thấp, mỗi cập nhật tương ứng với một tầng  $l$  và một mô-đun con  $m$  của mô hình. Với mỗi cặp  $(l, m)$ , cập nhật LoRA được xác định như sau:

$$\Delta W_{l,m}^{(i)} = s_i \cdot B_{l,m}^{(i)} A_{l,m}^{(i)}, \quad (3.1)$$

trong đó  $A_{l,m}^{(i)} \in \mathbb{R}^{r_i \times d_{in}}$ ,  $B_{l,m}^{(i)} \in \mathbb{R}^{d_{out} \times r_i}$ , và  $s_i = \alpha_i / r_i$  là hệ số chuẩn hoá của LoRA.

Do đó, mỗi adapter  $i$  có thể được đặc trưng bởi tập các cập nhật hạng thấp  $\{\Delta W_{l,m}^{(i)}\}_{l,m}$ , trong đó các cập nhật này phản ánh tri thức mà mô hình học được tại từng tầng và từng thành phần chức năng trong quá trình tinh chỉnh. Để đơn giản hoá ký hiệu trong phần trình bày tiếp theo, em lược bỏ các chỉ số tầng và thành phần  $(l, m)$ , và ký hiệu cập nhật LoRA của adapter  $i$  là  $\Delta W^{(i)}$ .

Trước tiên, TIES (TRIM, ELECT SIGN & MERGE) được sử dụng để xử lý hiện tượng xung đột hướng cập nhật giữa các tác vụ. Xuất phát từ tập các cập nhật LoRA  $\{\Delta W^{(i)}\}_{i=1}^N$ , TIES tiến hành hợp nhất thông qua ba bước tuần tự.

1. **Trim.** Ở bước đầu tiên, với mỗi adapter  $i$ , các phần tử có độ lớn nhỏ trong  $\Delta W^{(i)}$  bị loại bỏ bằng cách chỉ giữ lại top- $k\%$  phần tử theo trị tuyệt đối, trong đó  $k$  được điều khiển bởi tham số mật độ *density*  $\in (0, 1]$ . Kết quả của bước này là một cập nhật đã được tinh gọn:

$$\widehat{\Delta W}^{(i)} = \hat{\gamma}^{(i)} \odot \hat{\mu}^{(i)},$$

trong đó  $\hat{\gamma}^{(i)} \in \{-1, 0, +1\}^{d_{out} \times d_{in}}$  biểu diễn dấu của các phần tử còn lại, và  $\hat{\mu}^{(i)}$  là độ lớn tương ứng. Bước Trim giúp loại bỏ nhiễu từ các cập nhật có biên độ nhỏ, qua đó cải thiện tính ổn định của quá trình hợp nhất.

2. **Elect Sign.** Tiếp theo, một ma trận dấu chi phối  $\gamma^m$  được xây dựng để giải quyết sự bất đồng về hướng cập nhật tại từng tham số  $p$  giữa các mô hình khác nhau. Cụ thể, đối với mỗi tham số  $p \in \{1, 2, \dots, d\}$ , các giá trị  $\{\widehat{\Delta W}_p^{(i)}\}_{i=1}^n$  được phân nhóm theo dấu (+1 hoặc -1) và tính tổng độ lớn theo từng hướng. Dấu của  $\gamma_p^m$  là dấu có tổng độ lớn lớn hơn và được tính theo công thức:

$$\gamma_p^m = \text{sgn} \left( \sum_{i=1}^n \widehat{\Delta W}_p^{(i)} \right)$$

Cơ chế Elect đảm bảo rằng hướng cập nhật cuối cùng tại mỗi tham số phản ánh xu hướng chi phối chung của các tác vụ, thay vì đơn thuần là trung bình tuyến tính.

3. **Merge.** Sau khi xác định dấu chi phối, tại mỗi tham số  $p$ , em chỉ giữ lại các giá trị từ những mô hình có dấu trùng với  $\gamma_p^m$  và tính trung bình của chúng. Cụ thể, đặt  $A_p = \{i \in [n] \mid \hat{\gamma}_p^{(i)} = \gamma_p^m\}$ , khi đó giá trị hợp nhất tại tham số  $p$  được xác định bởi:

$$\Delta W_p^m = \frac{1}{|A_p|} \sum_{i \in A_p} w_i \widehat{\Delta W}_p^{(i)}$$

Sau khi áp dụng TIES, ta thu được một cập nhật hợp nhất  $\Delta W^m$  trong không gian tham số gốc. Bước tiếp theo là áp dụng xấp xỉ hạng thấp dựa trên SVD. Ban đầu, phép phân rã SVD được áp dụng lên  $\Delta W^m$ :

$$U, \Sigma, V^\top = \text{SVD}(\Delta W^m),$$

và chỉ giữ lại  $r$  giá trị suy biến lớn nhất để thu được xấp xỉ hạng thấp tối ưu theo nghĩa bình phương tối thiểu:

$$\Delta W^m \approx U_r \Sigma_r V_r^\top,$$

trong đó  $U_r = U[:, :r]$ ,  $\Sigma_r = \text{diag}(\Sigma[:, :r])$ , và  $V_r^\top = V^\top[:, :r]$ . Từ phân rã này, cập nhật hợp nhất được tái tham số về dạng LoRA thông qua:

$$B^m, A^m = U_r \Sigma_r, V_r^\top$$

Đáng chú ý, trong thiết lập của em, hạng  $r$  sau bước SVD được giữ bằng với hạng của các LoRA adapter ban đầu. Do đó, mục tiêu của SVD không nhằm giảm số chiều biểu diễn mà đóng vai trò như một phép chiếu trực chuẩn (*orthonormal projection*). Cụ thể, phép chiếu này bảo toàn các hướng tương ứng với các giá trị suy biến lớn nhất của ma trận trọng số hợp nhất, qua đó nhấn mạnh các thành phần chi phối trong không gian tham số, đồng thời làm suy giảm các thành phần có đóng góp nhỏ hơn phát sinh trong quá trình hợp nhất adapter.

Nhờ đó, phương pháp `ties_svd` vừa đảm bảo tích hợp tri thức đa tác vụ một cách có chọn lọc thông qua TIES, vừa duy trì được cấu trúc hạng thấp và hiệu quả tham số của LoRA thông qua SVD, tạo ra một bộ điều hợp hợp nhất ổn định và hiệu quả.

Sau bước trên, ta thu được một bộ điều hợp hợp nhất duy nhất  $(A^m, B^m)$ , đại diện cho cập nhật hạng thấp tổng hợp từ tất cả các tác vụ huấn luyện ở giai đoạn trước. Để chuẩn bị cho giai đoạn huấn luyện tiếp theo, em tiến hành tích hợp trực tiếp cập nhật này vào mô hình nền. Cụ thể, thao tác này thực hiện việc cộng tuyến tính cập nhật hợp nhất vào trọng số gốc:

$$\forall \ell \in \mathcal{L}_{\text{LoRA}} : \quad W^{(\ell)} \leftarrow W^{(\ell)} + sB^{m(\ell)}A^{m(\ell)},$$

trong đó  $\mathcal{L}_{\text{LoRA}}$  là tập các tầng của mô hình được áp dụng LoRA,  $\Delta W^{(l)}$  là trọng số của mô hình nền, và  $s = \alpha/r$  là hệ số chuẩn hóa của adapter. Sau khi cập nhật được gộp vào không gian tham số gốc, toàn bộ các mô-đun adapter liên quan được loại bỏ khỏi kiến trúc mạng, trả về một mô hình có kiến trúc đồng nhất với mô hình nền nhưng đã được dịch chuyển trong không gian tham số theo hướng học được từ các tác vụ đa nhiệm.

## 3.2 Giai đoạn 2: Huấn luyện toàn bộ tham số trong thiết lập đa tác vụ

Kết thúc giai đoạn 1, ta thu được một mô hình nền đã được kết hợp tham số theo hướng tổng hợp từ nhiều tác vụ. Trên cơ sở đó, Giai đoạn 2 tiến hành huấn luyện toàn bộ tham số của mô hình trong một thiết lập đa tác vụ. Không giống như giai đoạn đầu chỉ tinh chỉnh các tham số hạng thấp, giai đoạn này cho phép mọi trọng số của mô hình cùng tham gia tối ưu, từ đó tăng khả năng điều chỉnh và khai thác hiệu quả các biểu diễn đã được định hướng bởi adapter hợp nhất.

Cụ thể, dữ liệu huấn luyện được tổ chức dưới dạng các cặp instruction–response, trong đó mỗi mẫu đầu vào mô tả đầy đủ ngữ cảnh tác vụ, yêu cầu và các lựa chọn (nếu có) thông qua nội dung ngôn ngữ tự nhiên, còn đầu ra là phản hồi tương ứng. Định dạng chi tiết của các mẫu huấn luyện được minh họa trong chương tiếp theo.

Về mặt tối ưu, quá trình huấn luyện ở Giai đoạn 2 sử dụng một hàm mất mát duy nhất trên toàn bộ tập dữ liệu, cụ thể là hàm mất mát negative log-likelihood được áp dụng đồng nhất cho mọi mẫu huấn luyện. Do đó, mô hình được tối ưu hoá trực tiếp trên phân phối dữ liệu đa tác vụ tổng hợp, không sử dụng cơ chế cân bằng thủ công hay trọng số riêng cho từng nhiệm vụ. Với một mẫu huấn luyện  $(x, y)$ ,  $\mathcal{T}$  là tập các token được tính loss, hàm mất mát được định nghĩa như sau:

$$\mathcal{L}_{\text{NLL}}(x, y) = -\frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \log p_{\theta}(y_t \mid x, y_{<t}),$$

Việc huấn luyện toàn bộ tham số từ một điểm khởi tạo đã được hợp nhất mang lại hai lợi ích chính. Thứ nhất, mô hình có khả năng tinh chỉnh sâu các biểu diễn chung

đã được định hướng bởi các adapter đa tác vụ ở giai đoạn trước, vượt ra ngoài phạm vi điều chỉnh bị ràng buộc bởi các cập nhật hạng thấp. Thứ hai, quá trình huấn luyện toàn bộ tham số từ một điểm khởi tạo đã được hợp nhất cho phép mô hình tiếp tục điều chỉnh các biểu diễn chung ở mức toàn cục, qua đó đạt được một trạng thái cân bằng nhất quán giữa các tác vụ, thay vì chỉ kết hợp các cập nhật cục bộ ở không gian hạng thấp.

Nhờ sự kết hợp giữa hợp nhất tham số hiệu quả ở Giai đoạn 1 và huấn luyện toàn bộ tham số Giai đoạn 2, mô hình cuối cùng vừa kế thừa được tri thức chuyên biệt của từng tác vụ, vừa thể hiện khả năng khái quát hoá tốt hơn trong thiết lập đa tác vụ, đồng thời giúp giảm thiểu nguy cơ mất ổn định và giao thoa bất lợi giữa các tác vụ so với việc huấn luyện trực tiếp từ mô hình tiền huấn luyện ban đầu.

# Chương 4

## Thực nghiệm và Kết quả

### 4.1 Thiết lập thực nghiệm

#### 4.1.1 Tác vụ pháp lí

Xuyên suốt dự án này, mô hình được huấn luyện và đánh giá trên ba tác vụ pháp lí như sau:

1. **Multiple Choice (MCQ)** Tác vụ này kiểm tra khả năng hiểu và ghi nhớ kiến thức pháp luật dạng thực chứng (factual knowledge). Mô hình được cung cấp một câu hỏi pháp lí kèm theo các phương án trả lời (thường là 4 lựa chọn), trong đó chỉ có *một đáp án đúng* dựa trên quy định pháp luật hiện hành. Nhiệm vụ của mô hình là chọn ra phương án chính xác nhất.

Tác vụ này chủ yếu đánh giá năng lực đọc hiểu điều luật, nhận diện khái niệm pháp lí, và áp dụng trực tiếp quy định mà không đòi hỏi suy luận phức tạp nhiều bước.

2. **Natural Language Inference (NLI)** Tác vụ NLI tập trung đánh giá mối quan hệ logic giữa văn bản pháp luật và một câu hỏi pháp lí cụ thể. Mô hình được cung cấp:

- một đoạn trích từ văn bản luật (legal document),
- một câu hỏi pháp lí cụ thể (specific question),

và phải trả lời liệu đoạn luật đó có đủ thông tin để trả lời câu hỏi hay không (Có/Không).

Bản chất của tác vụ này là kiểm tra khả năng suy diễn bao hàm (entailment) trong ngữ cảnh pháp luật: mô hình cần hiểu nội dung điều luật, phạm vi áp dụng, điều kiện và đối tượng điều chỉnh để xác định tính liên quan và đầy đủ của thông tin.

3. **Syllogism Question Answering (SQA)** Đây là tác vụ khó nhất, nhằm đánh giá năng lực lập luận và diễn giải pháp lí ở mức độ cao. Mô hình được cung cấp một tình huống pháp lí hoặc câu hỏi phức hợp, yêu cầu đưa ra chuỗi lập luận đầy đủ, bao gồm:

- tiền đề lớn (dẫn chiếu điều luật, nghị định, thông tư),
- tiền đề nhỏ (phân tích việc áp dụng quy định vào tình huống cụ thể),
- kết luận cuối cùng có căn cứ pháp luật rõ ràng.

Khác với hai tác vụ trước, SQA không chỉ yêu cầu đáp án đúng/sai mà còn đánh giá chất lượng lập luận, tính logic, và khả năng giải thích bằng ngôn ngữ tự nhiên, phản ánh gần sát với cách lập luận của con người trong thực tiễn pháp lí.

### 4.1.2 Dữ liệu

Bảng 4.1: Kích thước bộ dữ liệu huấn luyện và đánh giá trên ba tác vụ

	<b>Train</b>	<b>Test</b>
Multichoice Questions	803	146
NLI Questions	745	150
Syllogism Questions	1989	144
<b>Total Samples</b>	<b>3537</b>	<b>440</b>

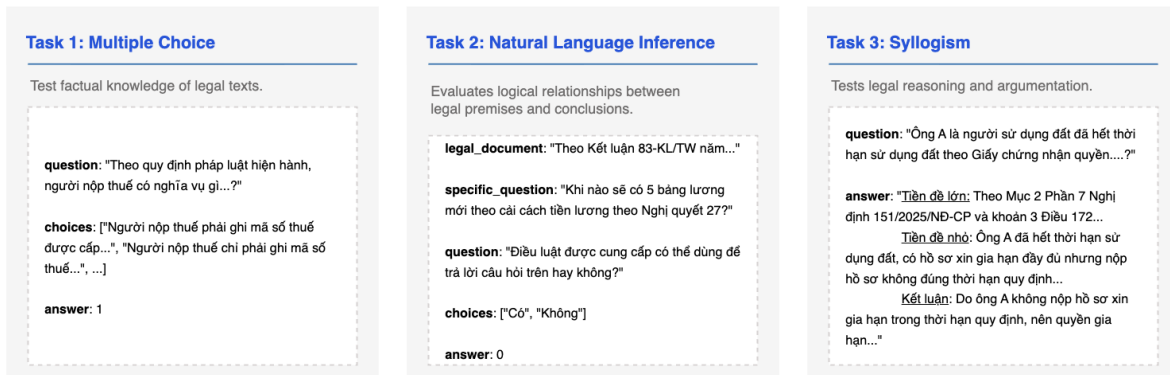
**Dữ liệu huấn luyện** Tập dữ liệu huấn luyện<sup>1</sup> được nhóm tác giả [7] chia sẻ bao gồm 3537 mẫu cho ba tác vụ, được định dạng theo format hội thoại ShareGPT nhằm đảm bảo tương thích với các mô hình ngôn ngữ lớn được tinh chỉnh theo chỉ dẫn (*instruction-tuned LLMs*). Định dạng tập huấn luyện được minh họa ở Hình 4.1.

<sup>1</sup><https://huggingface.co/datasets/vohuutridung/Vietnamese-Legal-Chat-Dataset>



Hình 4.1: Định dạng mẫu dữ liệu huấn luyện

**Dữ liệu đánh giá** Tập dữ liệu đánh giá<sup>2</sup> là bộ Public Test trong cuộc thi VLSP 2025 - LegalSLM<sup>3</sup>. Bộ đánh giá có khoảng 150 mẫu cho mỗi tác vụ, giúp đánh giá hiệu quả độ chính xác của mô hình trên các tác vụ pháp lý này.



Hình 4.2: Tổng quan về bộ dữ liệu đánh giá

### 4.1.3 Độ đo đánh giá

Trong nghiên cứu này, em sử dụng các thiết lập đánh giá khác nhau cho từng tác vụ nhằm phản ánh chính xác đặc thù của từng bài toán.

Đối với hai tác vụ MCQ và NLI, việc đánh giá được thực hiện bằng framework

<sup>2</sup><https://huggingface.co/datasets/vohuutridung/Public-Test>

<sup>3</sup><https://vlsp.org.vn/vlsp2025/eval/legalslm>



`lm-evaluation-harness`<sup>4</sup>. Cả hai tác vụ đều được thiết lập dưới dạng bài toán trắc nghiệm nhiều lựa chọn (*multiple-choice*), trong đó mô hình được yêu cầu lựa chọn một đáp án đúng từ tập các phương án cho trước. Hiệu năng được đo bằng độ chính xác (Accuracy), được tính là tỷ lệ phần trăm số mẫu mà mô hình dự đoán đúng so với tổng số mẫu trong tập đánh giá. Cách tiếp cận này phù hợp với bản chất phân loại rời rạc của cả hai tác vụ và đảm bảo tính tái lập trong quá trình đánh giá.

Đối với tác vụ SQA, do đầu ra của mô hình ở dạng câu trả lời tự do với cấu trúc lập luận phức tạp, các độ đo dựa trên so khớp chuỗi không phản ánh đầy đủ chất lượng câu trả lời. Vì vậy, em áp dụng phương pháp đánh giá dựa trên mô hình chấm điểm (LLM-as-a-judge). Cụ thể, các câu trả lời do mô hình sinh ra được đánh giá bởi mô hình *Qwen3-32B-AWQ*, đóng vai trò như một giám khảo tự động. Mô hình chấm điểm được cung cấp câu hỏi, đáp án tham chiếu và câu trả lời của mô hình, sau đó đưa ra điểm số trong khoảng từ 1 đến 10 dựa trên các tiêu chí bao gồm: tính chính xác của tiền đề pháp lý, diễn giải sự kiện, cấu trúc suy luận logic, mức độ tuân thủ pháp luật Việt Nam, tính nhất quán với đáp án tham chiếu, và mức độ tránh sinh thông tin không có căn cứ. Điểm đánh giá cuối cùng cho tác vụ SQA được tính bằng cách lấy trung bình điểm số trên toàn bộ tập đánh giá.

## 4.2 Kết quả

Bảng 4.2: Kết quả so sánh giữa các phương pháp huấn luyện.

Training Method	MCQ (%)	NLI (%)	SQA (%)	Average (%)
Two-stage Training (Ours)	85.62	<b>94.00</b>	57.01	<b>78.88</b>
LoRA Training Only	<b>87.67</b>	77.33	57.85	74.28
Full Fine-tuning	82.88	78.67	<b>60.21</b>	73.92
QLoRA Training Only	78.08	66.67	58.40	67.72

Kết quả thực nghiệm của em đã chứng minh được sự hiệu quả của khung huấn luyện hai giai đoạn được đề xuất trong báo cáo so với các phương pháp huấn luyện truyền thống trên cả ba tác vụ pháp lý.

Kết quả trên cho thấy một số nhận định quan trọng liên quan đến các phương pháp huấn luyện trong bài toán xử lý ngôn ngữ tự nhiên pháp lý tiếng Việt. Cách tiếp cận huấn luyện hai giai đoạn của em đạt hiệu năng trung bình cao nhất với mức 78.88%, cao hơn phương pháp kế tiếp 4.6%, thể hiện sự vượt trội đáng kể so với các phương pháp còn lại, kể cả các phương pháp huấn luyện toàn bộ tham số hay huấn luyện hiệu quả về tham số. Phương pháp này đặc biệt hiệu quả đối với các tác vụ MCQ và NLI, khi độ chính xác của mô hình của em trên 2 tác vụ lần lượt là 85.62% và 94%, đồng thời vẫn duy trì hiệu năng cạnh tranh đối với tác vụ SQA - một bài toán có độ khó cao hơn.

<sup>4</sup><https://github.com/EleutherAI/lm-evaluation-harness>

Ngoài ra, phân bố hiệu năng giữa các tác vụ cho thấy những khác biệt căn bản về mức độ phức tạp trong suy luận pháp lý tiếng Việt. Các tác vụ MCQ và NLI đạt hiệu năng khá tốt, trong khi bài toán SQA vẫn đưa ra những thách thức đáng kể đối với các phương pháp (60% trên tất cả các hướng tiếp cận). Sự chênh lệch này phản ánh quá trình chuyển dịch từ việc đánh giá kiến thức pháp lý mang tính thực tiễn sang các kỹ năng suy luận pháp lý phức tạp hơn.

Ở đây, phương tiếp huấn luyện hai giai đoạn thể hiện ưu thế rõ rệt trong việc thu hẹp khoảng cách này, khi vừa duy trì hiệu năng khá cạnh tranh trên tác vụ suy luận SQA, vừa đạt độ chính xác cao trên các tác vụ phân loại. Sự cân bằng này cho thấy tính hiệu quả của phương pháp này trong việc đáp ứng yêu cầu đa dạng trong miền pháp lý tiếng Việt.

## Chương 5

# Kết luận và Hướng phát triển

Trong báo cáo này, em đã trình bày hướng tiếp cận huấn luyện hai giai đoạn cho lớp mô hình ngôn ngữ nhỏ trên miền pháp lý tiếng Việt, với chiến lược áp dụng tinh chỉnh các bộ LoRA adapter theo từng tác vụ kết hợp bước huấn luyện toàn bộ tham số sau đó. Kết quả thực nghiệm chỉ rõ kết quả vượt trội của phương pháp khi so với các hướng tiếp cận truyền thống, và đạt được độ chính xác trung bình 78.88% trên ba tác vụ.

Trong các nghiên cứu tiếp theo, em dự định mở rộng và đào sâu phương pháp hiện tại theo một số hướng chính. Thứ nhất, nghiên cứu sẽ tập trung phân tích ảnh hưởng của các chiến lược kết hợp (*combination\_type*) khác nhau trong quá trình hợp nhất các adapter, cũng như đánh giá tác động của các tổ hợp trọng số (*weight combinations*) lên hiệu năng tổng thể của mô hình. Việc khảo sát các cấu hình này không chỉ giúp làm rõ vai trò của từng thành phần trong quá trình huấn luyện mà còn góp phần xác định các chiến lược kết hợp tối ưu cho từng loại tác vụ pháp lý. Thứ hai, để củng cố tính khái quát, nhất quán (*consistency*) của phương pháp, em sẽ tiến hành kiểm chứng hướng tiếp cận trên nhiều kiến trúc và mô hình nền khác nhau. Hướng mở rộng này nhằm chứng minh rằng phương pháp không mang tính đặc thù cho một kiến trúc riêng lẻ, mà có khả năng áp dụng rộng rãi cho nhiều mô hình ngôn ngữ quy mô nhỏ và trung bình trong các bài toán NLP pháp lý tiếng Việt.

# Tài liệu tham khảo

- [1] Farid Ariai, Joel Mackenzie, and Gianluca Demartini. 2025. Natural language processing for the legal domain: A survey of tasks, datasets, models, and challenges. *ACM Computing Surveys*, 58(6):1–37.
- [2] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Proceedings of the Thirty-seventh Conference on Neural Information Processing Systems*.
- [3] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *Proceedings of the International Conference on Learning Representations*.
- [4] Van-Hoang Le, Duc-Vu Nguyen, Kiet Van Nguyen, and Ngan Luu-Thuy Nguyen. 2025. Optimizing legal document retrieval in vietnamese with semi-hard negative mining. *CoRR*, abs/2507.14619.
- [5] Jonathan Li, Rohan Bhambhoria, and Xiaodan Zhu. 2022. Parameter-efficient legal domain adaptation. In *Proceedings of the Natural Legal Language Processing Workshop 2022*, pages 119–129, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- [6] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. DoRA: Weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 32100–32121. PMLR.
- [7] Ngo Dinh Luan and Kiet Nguyen Van. 2025. MinLegal at VLSP2025-LegalSLM: A two-stage LoRA-to-full fine-tuning approach for Vietnamese legal small language models. In *Proceedings of the 11th International Workshop on Vietnamese Language and Speech Processing*, pages 179–185, Hanoi, Vietnam. Association for Computational Linguistics.
- [8] Tien Son Pham, Doan Hieu Nguyen, Dai An Nguyen, and Viet Sang Dinh. 2024. Improving Vietnamese legal document retrieval using synthetic data. In *Proceed-*

*ings of the International Symposium on Information and Communication Technology*, pages 378–393. Springer.

- [9] George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. 2025. Model merging with svd to tie the knots. In *Proceedings of the International Conference on Learning Representations*.
- [10] Prateek Yadav, Derek Tam, Leshem Choshen, Colin A. Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models. In *Proceedings of the Conference on Neural Information Processing Systems*.
- [11] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Proceedings of the Forty-first International Conference on Machine Learning*.
- [12] Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. In *Proceedings of the International Conference on Learning Representations*.

# Phụ lục

## A. Chi tiết huấn luyện

### A.1. Cấu hình giai đoạn 1

Hyperparameter	Value
Base model	qwen3-1.7b-legal-pretrain
LoRA rank ( $r$ )	16
LoRA scaling factor ( $\alpha$ )	32
LoRA dropout	0.1
Target modules	<code>all_linear</code>
Training epochs	3
Learning rate	$2 \times 10^{-4}$
Maximum sequence length	4096

### A.2. Cấu hình giai đoạn 2

Hyperparameter	Value
Training epochs	2
Learning rate	$5 \times 10^{-5}$
Maximum sequence length	4096

### A.3. Cấu hình hợp nhất adapter

Hyperparameter	Description	Value
Adapters	Adapters to be merged	[mcq, nli, sqa]
Weights	Contribution weights	[1.0, 1.0, 1.5]
New adapter name	Output adapter identifier	merge
Combination type	Merging strategy	ties_svd
Majority sign method	Sign resolution rule	total
Density	Sparsity control factor	1.0

### A.4. Phần cứng huấn luyện

Các thí nghiệm trong nghiên cứu này được thực hiện trên hai thiết lập phần cứng khác nhau. Đối với các lần huấn luyện có quy mô vừa phải và có thể đáp ứng trong giới hạn tài nguyên miễn phí, em sử dụng GPU NVIDIA Tesla P100 được cung cấp bởi nền tảng Kaggle. Trong trường hợp các thí nghiệm yêu cầu tài nguyên tính toán lớn hơn (huấn luyện toàn bộ tham số), em thuê GPU trên nền tảng vast.ai.

## B. Phân tích kết quả bổ sung

### B.1. Kết quả mô hình sau giai đoạn 1

Khi tiến hành đánh giá mô hình với adapter hợp nhất sau giai đoạn 1 (chưa được huấn luyện trong giai đoạn 2), em thu được kết quả như trình bày trong Bảng 1. Kết quả này cho thấy hiệu năng của mô hình còn hạn chế khi chỉ áp dụng bước hợp nhất adapter, từ đó làm rõ vai trò của giai đoạn 2 trong việc cải thiện hiệu suất tổng thể.

Bảng 1: Hiệu năng của adapter hợp nhất sau mỗi bước.

Model	MCQ (%)	NLI (%)	SQA (%)	Average (%)
Stage 1	74.66	67.33	52.43	64.81
Stage 1 + Stage 2	85.62	94.00	57.01	78.88

### B.2. So sánh hiệu năng giữa các cấu hình hợp nhất adapter khác nhau

Bảng 2 chỉ ra rằng hiệu năng của phương pháp huấn luyện hai giai đoạn chịu ảnh hưởng đáng kể bởi cả chiến lược hợp nhất adapter (*combination type*) và trọng số đóng góp của từng adapter thành phần (*weights*). Nhìn chung, các cấu hình sử dụng trọng số không đồng đều có xu hướng cải thiện hiệu suất trung bình so với thiết lập trọng số

Bảng 2: Hiệu năng của phương pháp huấn luyện hai giai đoạn với các cấu hình hợp nhất adapter khác nhau. Trong đó, *combination type* chỉ chiến lược hợp nhất adapter, còn *weights* biểu thị trọng số đóng góp của từng adapter thành phần.

Combination type	Weights	MCQ (%)	NLI (%)	SQA (%)	Average (%)
linear	[0.3, 0.3, 0.4]	78.77	51.33	52.78	60.96
linear	[1, 1, 1]	84.25	83.33	24.93	64.17
linear	[1, 1, 2]	<b>87.67</b>	72.67	<b>59.31</b>	<b>73.22</b>
cat	[0.3, 0.3, 0.4]	82.19	74.67	55.56	70.81
cat	[1, 1, 1]	82.88	<b>91.33</b>	45.35	73.19
cat	[1, 1, 2]	<b>87.67</b>	60.00	57.57	68.41
dare_linear	[0.3, 0.3, 0.4]	83.56	63.33	54.72	67.20

đồng nhất, cho thấy tầm quan trọng của việc điều chỉnh mức đóng góp tương đối giữa các adapter nguồn. Xét theo từng chiến lược hợp nhất, phương pháp *linear* đạt hiệu năng cao nhất trên tác vụ MCQ khi sử dụng cấu hình trọng số [1, 1, 2], đồng thời cho kết quả cân bằng hơn so với các cấu hình còn lại. Tuy nhiên, với trọng số đồng nhất [1, 1, 1], hiệu năng trên tác vụ SQA suy giảm mạnh, dẫn đến điểm trung bình thấp hơn đáng kể. Đối với chiến lược *cat*, các cấu hình cho thấy hiệu năng tương đối ổn định trên cả ba tác vụ, đặc biệt là trên tác vụ NLI với điểm số cao nhất đạt 91.33%. Mặc dù không phải lúc nào cũng đạt điểm cao nhất trên từng tác vụ riêng lẻ, phương pháp này mang lại hiệu suất trung bình tốt nhất trong bảng, phản ánh khả năng kết hợp thông tin từ nhiều adapter theo cách hiệu quả hơn. Tổng thể, các kết quả trong Bảng 2 khẳng định rằng việc lựa chọn chiến lược hợp nhất adapter và cấu hình trọng số phù hợp là yếu tố then chốt để đạt được hiệu năng cao và ổn định trong thiết lập huấn luyện hai giai đoạn.