**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TPHCM**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KỲ**

**Hệ thống trả lời câu hỏi từ ảnh chứa văn bản tiếng Việt dựa trên OCR**

**GIẢNG VIÊN HƯỚNG DẪN: MSc. Võ Quang Hoàng Khang**

**SINH VIÊN THỰC HIỆN 1: Nguyễn Thành Trọng**

**MÃ SỐ SINH VIÊN: 22642481**

**SINH VIÊN THỰC HIỆN 2: Ngô Quang Vinh**

**MÃ SỐ SINH VIÊN: 2104775**

*TP HỒ CHÍ MINH, Tháng 05 năm 2025*

**INDUSTRIAL UNIVERSITY OF HO CHI MINH CITY**

**FACULTY OF INFORMATION TECHNOLOGY**



**FINAL PROJECT**

**OCR-based Question Answering System for Vietnamese Text in Images**

**SUPERVISOR: MSc. Vo Quang Hoang Khang**

**STUDENT NAME: Nguyễn Thành Trọng**

**STUDENT CODE: 22642481**

**STUDENT NAME: Ngô Quang Vinh**

**STUDENT CODE: 2104775**

*HO CHI MINH CITY, Month 05 year 2025*

**TÓM TẮT**

Nghiên cứu này đề xuất một mô hình học sâu cải tiến cho bài toán Text-VQA, hỗ trợ trả lời câu hỏi dựa trên văn bản tiếng Việt trong ảnh, đặc biệt phù hợp với các tài liệu giáo dục. Mô hình được huấn luyện và đánh giá trên bộ dữ liệu ViTextVQA từ Hugging Face, sử dụng các chỉ số như BLEU và ROUGE. Kết quả cho thấy mô hình đạt hiệu suất tạm ổn, hứa hẹn ứng dụng thực tiễn trong giáo dục và xử lý tài liệu, hỗ trợ người dùng truy xuất thông tin chính xác và hiệu quả hơn.

**LỜI CẢM ƠN**

Lời đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến Ban Giám hiệu Trường Đại học Công nghiệp TP. Hồ Chí Minh cùng Ban Chủ nhiệm Khoa Công nghệ Thông tin, đã luôn quan tâm và tạo điều kiện thuận lợi cho chúng em trong suốt thời gian học tập tại trường.

Đặc biệt, chúng em xin bày tỏ lòng biết ơn sâu sắc đến thầy hướng dẫn ThS. Võ Quang Hoàng Khang đã trực tiếp hướng dẫn và hỗ trợ chúng em trong suốt quá trình tìm hiểu và học tập bộ môn Thị giác máy tính.

Trong khuôn khổ của bài báo cáo này, chúng em đã cố gắng vận dụng những kiến thức đã học để trình bày một cách rõ ràng và đầy đủ. Tuy nhiên, do hạn chế về kiến thức và kinh nghiệm thực tiễn, chắc chắn bài báo cáo sẽ không tránh khỏi những thiếu sót. Chúng em rất mong nhận được những ý kiến đóng góp từ thầy để có thể hoàn thiện và phát triển hơn trong tương lai.

Chúng em xin kính chúc thầy luôn dồi dào sức khỏe, hạnh phúc và thành công.

Chúng em xin chân thành cảm ơn!

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

**TP. Hồ Chí Minh, ngày…. tháng …. năm….**

CHỮ KÝ CỦA GIẢNG VIÊN

**NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN 1**

**TP. Hồ Chí Minh, ngày…. Tháng …. năm….**

CHỮ KÝ CỦA GIẢNG VIÊN

**NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN 2**

**TP. Hồ Chí Minh, ngày…. tháng …. năm….**

CHỮ KÝ CỦA GIẢNG VIÊN

**LỜI MỞ ĐẦU**

Trong thời đại công nghệ phát triển mạnh mẽ, các hệ thống trí tuệ nhân tạo không ngừng được cải tiến nhằm hỗ trợ con người xử lý thông tin một cách hiệu quả hơn. Việc kết hợp giữa thị giác máy tính và xử lý ngôn ngữ tự nhiên đã mở ra nhiều hướng nghiên cứu mới, đặc biệt là trong các bài toán yêu cầu máy hiểu cả hình ảnh lẫn văn bản. Một trong số đó là bài toán Visual Question Answering (VQA), nơi hệ thống cần trả lời các câu hỏi dựa trên nội dung hình ảnh.

Trong khuôn khổ môn học Thị giác máy tính tại Trường Đại học Công nghiệp TP. Hồ Chí Minh, nhóm chúng em đã thực hiện đề tài “Hệ thống trả lời câu hỏi từ ảnh có chứa văn bản tiếng Việt dựa trên OCR”. Đây là đề tài mang tính ứng dụng cao và cũng là cơ hội để nhóm tiếp cận, thực hành với các mô hình học sâu hiện đại, kết hợp kiến thức về thị giác máy tính và xử lý ngôn ngữ tự nhiên.

Báo cáo này trình bày toàn bộ quá trình thực hiện đề tài: từ tìm hiểu cơ sở lý thuyết, thiết kế mô hình, xử lý dữ liệu, đến đánh giá kết quả và triển khai mô hình thử nghiệm trên giao diện web. Nhóm hy vọng đây sẽ là một bước khởi đầu ý nghĩa cho những nghiên cứu tiếp theo trong lĩnh vực AI hỗ trợ ngôn ngữ tiếng Việt.

**MỤC LỤC**

[**CHƯƠNG 1. TỔNG QUAN 1**](#_Toc198556043)

[**1.1. Lý do chọn đề tài 1**](#_Toc198556044)

[**1.2. Mục tiêu nghiên cứu 2**](#_Toc198556045)

[**1.3. Phạm vi nghiên cứu 2**](#_Toc198556046)

[**1.4. Phương pháp nghiên cứu 2**](#_Toc198556047)

[**1.4.1. Nghiên cứu tài liệu 2**](#_Toc198556048)

[**1.4.2. Xử lý và tiền xử lý dữ liệu 3**](#_Toc198556049)

[**1.4.3. Xây dựng mô hình 3**](#_Toc198556050)

[**1.4.4. Kiểm thử và đánh giá mô hình 3**](#_Toc198556051)

[**1.4.5. Phân tích và cải tiến 4**](#_Toc198556052)

[**CHƯƠNG 2. CÁC NGHIÊN CỨU LIÊN QUAN 5**](#_Toc198556053)

[**2.1. Các mô hình Text-VQA cho tiếng anh 5**](#_Toc198556054)

[**2.2. Các mô hình Text-VQA cho tiếng Việt 5**](#_Toc198556055)

[**2.3. Các kiến trúc trích xuất đặc trưng ảnh 6**](#_Toc198556056)

[**2.3.1. Các kiến trúc CNN truyền thống 6**](#_Toc198556057)

[**2.3.2. Các mô hình Vision Transformer (ViT) 6**](#_Toc198556058)

[**2.3.3. Các kiến trúc lớn tiền huấn luyện đa nhiệm (Foundation Vision Models) 6**](#_Toc198556059)

[**2.4. Hướng nghiên cứu đề xuất 7**](#_Toc198556060)

[**CHƯƠNG 3. CƠ SỞ LÝ THUYẾT 8**](#_Toc198556061)

[**3.1. Visual Question Answering (VQA) 8**](#_Toc198556062)

[**3.2. Text-based Visual Question Answering (Text VQA) 8**](#_Toc198556063)

[**3.3. Nhận diện văn bản trong ảnh (OCR – Optical Character Recognition) 9**](#_Toc198556064)

[**3.4. InternViT 9**](#_Toc198556065)

[**3.5. Các thành phần kỹ thuật liên quan 9**](#_Toc198556066)

[**3.5.1. Convolutional Neural Network (CNN) 9**](#_Toc198556067)

[**3.5.2. LSTM (Long Short-Term Memory) 9**](#_Toc198556068)

[**3.5.3. Transformer 9**](#_Toc198556069)

[**3.5.4. Attention Mechanism 10**](#_Toc198556070)

[**3.6. Các phương pháp đánh giá mô hình 10**](#_Toc198556071)

[**3.6.1. BLEU (Bilingual Evaluation Understudy) 10**](#_Toc198556072)

[**3.6.2. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) 10**](#_Toc198556073)

[**CHƯƠNG 4. MÔ HÌNH ĐỀ XUẤT 12**](#_Toc198556074)

[**4.1. Mô hình dựa trên InternVit và LSTM 12**](#_Toc198556075)

[**4.1.1. Khối trích xuất đặc trưng thị giác (Vision Backbone) 12**](#_Toc198556076)

[**4.1.2. Khối mã hoá câu hỏi (Question Encoder) 12**](#_Toc198556077)

[**4.1.3. Khối kết hợp đặc trưng (Feature Fusion) 12**](#_Toc198556078)

[**4.1.4. Bộ giải mã câu trả lời (Decoder) 12**](#_Toc198556079)

[**4.1.5. Kiến trúc mô hình 13**](#_Toc198556080)

[**4.2. Mô hình dựa trên InternViT, PhoBert và cơ chế fusion–decoder generative 13**](#_Toc198556081)

[**4.2.1. Image Encoder – InternViT 13**](#_Toc198556082)

[**4.2.2. Text Encoder (PhoBert) 13**](#_Toc198556083)

[**4.2.3. Khối FusionBlock - Kết hợp thị giác & ngôn ngữ 14**](#_Toc198556084)

[**4.2.4. Khối Generative Decoder – sinh đáp án 15**](#_Toc198556085)

[**4.2.5. Khối Sequence Generation 16**](#_Toc198556086)

[**4.2.6. Kiến trúc mô hình 16**](#_Toc198556087)

[**4.3. Mô hình finetune dựa trên VinTern 18**](#_Toc198556088)

[**4.3.1. Image Encoder – InternViT 18**](#_Toc198556089)

[**4.3.2. Text Encoder – Qwen2-0.5B-Instruct 18**](#_Toc198556090)

[**4.3.3. Fusion & Generative Decoder 18**](#_Toc198556091)

[**4.3.4. Kiến trúc mô hình 19**](#_Toc198556092)

[**CHƯƠNG 5. THỰC NGHIỆM 20**](#_Toc198556093)

[**5.1. Môi trường thực nghiệm 20**](#_Toc198556094)

[**5.2. Tập dữ liệu 21**](#_Toc198556095)

[**5.3. Ứng dụng thực nghiệm 21**](#_Toc198556096)

[**5.3.1. Quy trình huấn luyện mô hình 21**](#_Toc198556097)

[**5.3.2. Cấu hình huấn luyện 21**](#_Toc198556098)

[**5.3.2.1. Mô hình dựa trên InternVit và LSTM 21**](#_Toc198556099)

[**5.3.2.2. Mô hình dựa trên InternViT, PhoBert và cơ chế fusion–decoder generative 22**](#_Toc198556100)

[**5.3.2.3. Mô hình finetune dựa trên VinTern 22**](#_Toc198556101)

[**5.4. Kết quả thực nghiệm 23**](#_Toc198556102)

[**5.4.1. Mô hình dựa trên InternVit và LSTM 23**](#_Toc198556103)

[**5.5. So sánh kết quả giữa các mô hình 23**](#_Toc198556104)

[**InternViT + LSTM 23**](#_Toc198556105)

[**InternViT, PhoBert và cơ chế fusion–decoder generative 23**](#_Toc198556106)

[**CHƯƠNG 1. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 24**](#_Toc198556107)

[**6.1. Kết luận 24**](#_Toc198556108)

[**6.2. Hướng phát triển 24**](#_Toc198556109)

[**1.1. Lý do chọn đề tài 1**](#_Toc198556044)

[**1.2. Mục tiêu nghiên cứu 2**](#_Toc198556045)

[**1.3. Phạm vi nghiên cứu 2**](#_Toc198556046)

[**1.4. Phương pháp nghiên cứu 2**](#_Toc198556047)

[**1.4.1. Nghiên cứu tài liệu 2**](#_Toc198556048)

[**1.4.2. Xử lý và tiền xử lý dữ liệu 3**](#_Toc198556049)

[**1.4.3. Xây dựng mô hình 3**](#_Toc198556050)

[**1.4.4. Kiểm thử và đánh giá mô hình 3**](#_Toc198556051)

[**1.4.5. Phân tích và cải tiến 4**](#_Toc198556052)

[**CHƯƠNG 2. CÁC NGHIÊN CỨU LIÊN QUAN 5**](#_Toc198556053)

[**2.1. Các mô hình Text-VQA cho tiếng anh 5**](#_Toc198556054)

[**2.2. Các mô hình Text-VQA cho tiếng Việt 5**](#_Toc198556055)

[**2.3. Các kiến trúc trích xuất đặc trưng ảnh 6**](#_Toc198556056)

[**2.3.1. Các kiến trúc CNN truyền thống 6**](#_Toc198556057)

[**2.3.2. Các mô hình Vision Transformer (ViT) 6**](#_Toc198556058)

[**2.3.3. Các kiến trúc lớn tiền huấn luyện đa nhiệm (Foundation Vision Models) 6**](#_Toc198556059)

[**2.4. Hướng nghiên cứu đề xuất 7**](#_Toc198556060)

[**CHƯƠNG 3. CƠ SỞ LÝ THUYẾT 8**](#_Toc198556061)

[**3.1. Visual Question Answering (VQA) 8**](#_Toc198556062)

[**3.2. Text-based Visual Question Answering (Text VQA) 8**](#_Toc198556063)

[**3.3. Nhận diện văn bản trong ảnh (OCR – Optical Character Recognition) 9**](#_Toc198556064)

[**3.4. InternViT 9**](#_Toc198556065)

[**3.5. Các thành phần kỹ thuật liên quan 9**](#_Toc198556066)

[**3.5.1. Convolutional Neural Network (CNN) 9**](#_Toc198556067)

[**3.5.2. LSTM (Long Short-Term Memory) 9**](#_Toc198556068)

[**3.5.3. Transformer 9**](#_Toc198556069)

[**3.5.4. Attention Mechanism 10**](#_Toc198556070)

[**3.6. Các phương pháp đánh giá mô hình 10**](#_Toc198556071)

[**3.6.1. BLEU (Bilingual Evaluation Understudy) 10**](#_Toc198556072)

[**3.6.2. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) 10**](#_Toc198556073)

[**CHƯƠNG 4. MÔ HÌNH ĐỀ XUẤT 12**](#_Toc198556074)

[**4.1. Mô hình dựa trên InternVit và LSTM 12**](#_Toc198556075)

[**4.1.1. Khối trích xuất đặc trưng thị giác (Vision Backbone) 12**](#_Toc198556076)

[**4.1.2. Khối mã hoá câu hỏi (Question Encoder) 12**](#_Toc198556077)

[**4.1.3. Khối kết hợp đặc trưng (Feature Fusion) 12**](#_Toc198556078)

[**4.1.4. Bộ giải mã câu trả lời (Decoder) 12**](#_Toc198556079)

[**4.1.5. Kiến trúc mô hình 13**](#_Toc198556080)

[**4.2. Mô hình dựa trên InternViT, PhoBert và cơ chế fusion–decoder generative 13**](#_Toc198556081)

[**4.2.1. Image Encoder – InternViT 13**](#_Toc198556082)

[**4.2.2. Text Encoder (PhoBert) 13**](#_Toc198556083)

[**4.2.3. Khối FusionBlock - Kết hợp thị giác & ngôn ngữ 14**](#_Toc198556084)

[**4.2.4. Khối Generative Decoder – sinh đáp án 15**](#_Toc198556085)

[**4.2.5. Khối Sequence Generation 16**](#_Toc198556086)

[**4.2.6. Kiến trúc mô hình 16**](#_Toc198556087)

[**4.3. Mô hình finetune dựa trên VinTern 18**](#_Toc198556088)

[**4.3.1. Image Encoder – InternViT 18**](#_Toc198556089)

[**4.3.2. Text Encoder – Qwen2-0.5B-Instruct 18**](#_Toc198556090)

[**4.3.3. Fusion & Generative Decoder 18**](#_Toc198556091)

[**4.3.4. Kiến trúc mô hình 19**](#_Toc198556092)

[**CHƯƠNG 5. THỰC NGHIỆM 20**](#_Toc198556093)

[**5.1. Môi trường thực nghiệm 20**](#_Toc198556094)

[**5.2. Tập dữ liệu 21**](#_Toc198556095)

[**5.3. Ứng dụng thực nghiệm 21**](#_Toc198556096)

[**5.3.1. Quy trình huấn luyện mô hình 21**](#_Toc198556097)

[**5.3.2. Cấu hình huấn luyện 21**](#_Toc198556098)

[**5.3.2.1. Mô hình dựa trên InternVit và LSTM 21**](#_Toc198556099)

[**5.3.2.2. Mô hình dựa trên InternViT, PhoBert và cơ chế fusion–decoder generative 22**](#_Toc198556100)

[**5.3.2.3. Mô hình finetune dựa trên VinTern 22**](#_Toc198556101)

[**5.4. Kết quả thực nghiệm 23**](#_Toc198556102)

[**5.4.1. Mô hình dựa trên InternVit và LSTM 23**](#_Toc198556103)

[**5.5. So sánh kết quả giữa các mô hình 23**](#_Toc198556104)

[**InternViT + LSTM 23**](#_Toc198556105)

[**InternViT, PhoBert và cơ chế fusion–decoder generative 23**](#_Toc198556106)

[**CHƯƠNG 1. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 24**](#_Toc198556107)

[**6.1. Kết luận 24**](#_Toc198556108)

[**6.2. Hướng phát triển 24**](#_Toc198556109)

**MỤC LỤC HÌNH ẢNH**

[Hình 1. Kiến trúc mô hình 11](#_Toc184148871)

[Hình 2. Độ chính xác trong quá trình huấn luyện 27](#_Toc184148872)

[Hình 3. Loss trong quá trình huấn luyện 27](#_Toc184148873)

[Hình 4. Ma trận nhầm lẫn của mô hình đề xuất 28](#_Toc184148874)

[Hình 5. Biểu đồ so sánh số liệu các mô hình phổ biến với mô hình đề xuất 29](#_Toc184148875)

[Hình 6. Biểu đồ so sánh số liệu các mô hình thay đổi khối khác với mô hình đề xuất 32](#_Toc184148876)

**MỤC LỤC BẢNG BIỂU**

[Bảng 1. Các phương pháp và tham số tăng cường dữ liệu 24](#_Toc184207958)

[Bảng 2. Kết quả thử nghiệm và xác thực chéo của mô hình đề xuất 26](#_Toc184207959)

[Bảng 3. So sánh các tiêu chí đánh giá của mô hình đề xuất với các biến thể của nó 31](#_Toc184207960)

[Bảng 4: So sánh các tiêu chí đánh giá của mô hình gốc với mô hình đề xuất 33](#_Toc184207961)

**DANH MỤC CÁC THUẬT NGỮ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| Từ viết tắt | Từ đầy đủ | Nghĩa |
| **VQA** | Visual Question Answering | Hệ thống trả lời câu hỏi dựa trên nội dung hình ảnh, kết hợp thị giác máy tính và xử lý ngôn ngữ tự nhiên để hiểu và trả lời câu hỏi liên quan đến hình ảnh. |
| **TextVQA** | Text-based Visual Question Answering | Một biến thể của VQA, tập trung vào trả lời câu hỏi dựa trên văn bản xuất hiện trong hình ảnh, đòi hỏi khả năng nhận diện văn bản (OCR) và hiểu ngữ nghĩa. |
| **LSTM** | Long Short-Term Memory | Mạng nơ-ron hồi tiếp đặc biệt, được thiết kế để xử lý và ghi nhớ thông tin trong các chuỗi dữ liệu dài, thường dùng trong xử lý ngôn ngữ tự nhiên và VQA để mã hóa câu hỏi. |
| **ViT** | Vision Transformer | Kiến trúc Transformer áp dụng cho xử lý ảnh, chia ảnh thành các patch và sử dụng cơ chế attention để trích xuất đặc trưng hình ảnh, hiệu quả trong các nhiệm vụ thị giác như VQA. |
| **OCR** | Optical Character Recognition | Công nghệ nhận diện và chuyển đổi văn bản trong ảnh thành chuỗi ký tự mà máy tính có thể xử lý, là thành phần quan trọng trong TextVQA. |
| **CNN** | Convolutional Neural Network | Mạng nơ-ron tích chập, được sử dụng rộng rãi trong xử lý ảnh để trích xuất đặc trưng như cạnh, hình dạng, thường làm backbone cho các mô hình thị giác. |
| **PhoBERT** |  | Mô hình Transformer được tiền huấn luyện cho tiếng Việt, tối ưu hóa cho xử lý ngôn ngữ tự nhiên, thường dùng để mã hóa câu hỏi trong các hệ thống VQA tiếng Việt. |
| **InternViT** |  | Một biến thể của Vision Transformer, được tiền huấn luyện trên dữ liệu đa dạng, có khả năng trích xuất đặc trưng hình ảnh mạnh mẽ, phù hợp cho TextVQA và các nhiệm vụ thị giác khác. |

# **TỔNG QUAN**

## Lý do chọn đề tài

Trong thời đại công nghệ hiện nay, hình ảnh chứa văn bản xuất hiện rất phổ biến trong đời sống, chẳng hạn như hóa đơn, bảng hiệu, giấy tờ, thực đơn, nhãn sản phẩm,... Những hình ảnh này không chỉ truyền tải thông tin bằng hình ảnh trực quan mà còn chứa nội dung văn bản quan trọng. Tuy nhiên, để máy tính có thể hiểu và xử lý các loại hình ảnh này thì không chỉ cần nhận diện được đối tượng trong ảnh, mà còn phải đọc được văn bản và hiểu được ngữ nghĩa của nó.

Bài toán Text VQA (Text-based Visual Question Answering) là một hướng nghiên cứu mở rộng từ bài toán VQA truyền thống, trong đó hệ thống phải trả lời câu hỏi dựa trên nội dung văn bản xuất hiện trong ảnh. Đây là một bài toán khá phức tạp vì đòi hỏi sự kết hợp giữa hai lĩnh vực: thị giác máy tính để xử lý hình ảnh và xử lý ngôn ngữ tự nhiên để hiểu câu hỏi và sinh câu trả lời.

Hiện nay, phần lớn các nghiên cứu về Text VQA được thực hiện chủ yếu cho tiếng Anh, nơi có sẵn nhiều dữ liệu và công cụ hỗ trợ. Trong khi đó, tiếng Việt lại chưa có nhiều hệ thống hiệu quả do hạn chế về dữ liệu và đặc điểm ngôn ngữ phức tạp. Một số nghiên cứu đã bắt đầu tiếp cận bài toán này cho tiếng Việt nhưng nhìn chung vẫn còn nhiều thách thức, đặc biệt trong việc áp dụng vào các tình huống thực tế.

Chính vì vậy, nhóm chúng em chọn đề tài này với mong muốn được tìm hiểu sâu hơn về bài toán Text VQA, đồng thời thử nghiệm xây dựng một hệ thống có khả năng trả lời câu hỏi từ hình ảnh chứa văn bản tiếng Việt. Qua đó, nhóm hy vọng có thể đóng góp một phần nhỏ vào việc mở rộng ứng dụng của thị giác máy tính và xử lý ngôn ngữ tự nhiên trong ngôn ngữ tiếng Việt, cũng như rèn luyện thêm kiến thức và kỹ năng liên quan đến mô hình AI đa phương thức.

## Mục tiêu nghiên cứu

Mục tiêu của đề tài là phát triển một hệ thống có khả năng trả lời câu hỏi dựa trên nội dung trong ảnh chứa văn bản tiếng Việt. Hệ thống cần hiểu và kết hợp hiệu quả giữa đặc trưng thị giác và ngôn ngữ để đưa ra câu trả lời phù hợp với ngữ cảnh.

Cụ thể, đề tài hướng đến:

* Khai thác đặc trưng từ ảnh và câu hỏi để phục vụ cho bài toán TextVQA tiếng Việt.
* Thiết kế mô hình kết hợp thông tin đa phương thức một cách hiệu quả.
* Đánh giá độ chính xác và phân tích hiệu quả của mô hình trên tập dữ liệu thử nghiệm.

## Phạm vi nghiên cứu

Đề tài tập trung vào bài toán Text VQA (Text-based Visual Question Answering) trên ảnh có chứa văn bản tiếng Việt, với đầu vào là ảnh và câu hỏi, đầu ra là câu trả lời dạng văn bản.

Phạm vi cụ thể bao gồm:

* Xử lý dữ liệu từ bộ ViTextVQA, bao gồm ảnh, câu hỏi và câu trả lời tiếng Việt.
* Xử lý và huấn luyện mô hình trên bộ dữ liệu ViTextVQA.
* Tạo một tập dữ liệu kiểm thử (test set) tương tự để đánh giá mô hình.
* Sử dụng các kỹ thuật thị giác máy tính và xử lý ngôn ngữ tự nhiên để xây dựng mô hình Text VQA.
* Xây dựng giao diện web đơn giản để demo mô hình trả lời câu hỏi từ ảnh.
* Đánh giá hiệu quả mô hình dựa trên độ chính xác của câu trả lời.

## Phương pháp nghiên cứu

1. **Nghiên cứu tài liệu**

Để xây dựng mô hình text VQA hiệu quả, việc nghiên cứu tài liệu là bước đầu tiên quan trọng. Các tài liệu liên quan đến text VQA, đặc biệt là những nghiên cứu áp dụng trên văn bản tiếng Việt trong ảnh, sẽ được nghiên cứu kỹ lưỡng. Việc này giúp hiểu rõ về các phương pháp, mô hình đã được sử dụng trong các bài toán tương tự, cũng như những thách thức khi xử lý văn bản tiếng Việt trong ảnh.

1. **Xử lý và tiền xử lý dữ liệu**

Dữ liệu đầu vào của hệ thống bao gồm các bức ảnh có chứa văn bản tiếng Việt, kèm theo câu hỏi và câu trả lời. Các bước tiền xử lý dữ liệu bao gồm:

* Tiền xử lý ảnh: Các ảnh sẽ được chuẩn hóa kích thước, chuyển đổi thành các định dạng phù hợp với mô hình.
* Tiền xử lý văn bản: Câu hỏi và câu trả lời sẽ được xử lý qua các bước như tokenization, loại bỏ từ dừng, và chuyển đổi thành các embedding vector phù hợp với mô hình học sâu.

Bộ dữ liệu ViTextVQA sẽ được sử dụng làm dữ liệu huấn luyện chính, trong khi một bộ dữ liệu kiểm thử tương tự sẽ được tạo ra để đánh giá độ chính xác của mô hình.

1. **Xây dựng mô hình**

* Kết hợp InternViT-300M và PhoBERT để tận dụng khả năng trích xuất đặc trưng hình ảnh và mã hóa ngôn ngữ tiếng Việt.
* Tích hợp cơ chế self-attention và cross-attention trong FusionBlock để tăng cường khả năng kết hợp thông tin đa phương thức giữa ảnh và câu hỏi.
* Sử dụng kỹ thuật Transformer Decoder hoặc LSTM để sinh câu trả lời, tối ưu hóa bằng chiến lược tìm kiếm như beam search hoặc top-k sampling nhằm tăng tính đa dạng và chính xác của câu trả lời.

1. **Kiểm thử và đánh giá mô hình**

* Ứng dụng phương pháp train-test split (80% huấn luyện, 20% kiểm thử) trên bộ dữ liệu ViTextVQA (32,251 mẫu huấn luyện, 8,063 mẫu kiểm thử) để đánh giá hiệu suất mô hình.
* So sánh kết quả với các mô hình như mô hình dựa trên InternViT và LSTM, và các mô hình VQA tiếng Việt khác (InternViT, PhoBert và fusion–decoder generative) dựa trên các chỉ số BLEU, và ROUGE (ROUGE-1, ROUGE-2, ROUGE-L).
* Phân tích độ ổn định của mô hình thông qua các chỉ Phân tích độ ổn định của mô hình thông qua các chỉ số đánh giá và kiểm tra tính phù hợp của câu trả lời trong ngữ cảnh đời sống thường ngày.

1. **Phân tích và cải tiến**

* Áp dụng các kỹ thuật tăng cường dữ liệu (data augmentation) như xoay, lật ảnh, và chuẩn hóa văn bản tiếng Việt để tăng tính đa dạng của dữ liệu.
* Tối ưu hóa mô hình thông qua fine-tuning các lớp cuối của InternViT và điều chỉnh siêu tham số (learning rate, batch size, beam size) để nâng cao hiệu suất và khả năng ứng dụng thực tế trong giáo dục.

# **CÁC NGHIÊN CỨU LIÊN QUAN**

Trong những năm gần đây, bài toán Visual Question Answering (VQA), đặc biệt là Text-based VQA, đã thu hút được sự quan tâm mạnh mẽ từ cộng đồng nghiên cứu thị giác máy tính và xử lý ngôn ngữ. Nhiều phương pháp đã được đề xuất nhằm tăng cường khả năng hiểu nội dung hình ảnh có chứa văn bản.

## Các mô hình Text-VQA cho tiếng anh

Một trong những mô hình tiên phong trong lĩnh vực này là M4C (Multimodal Multi-Copy Mesh) được đề xuất bởi Singh et al. M4C sử dụng cơ chế copy từ các token văn bản được trích xuất bởi OCR trong ảnh, kết hợp với trích xuất đặc trưng từ ảnh và câu hỏi để sinh câu trả lời. M4C được huấn luyện trên tập TextVQA – một trong những bộ dữ liệu đầu tiên chú trọng đến câu hỏi liên quan đến văn bản trong ảnh.

Tiếp theo đó, mô hình TAP (Text-Aware Pretraining) đề xuất tận dụng văn bản OCR như một phần trong giai đoạn tiền huấn luyện (pretraining), giúp mô hình có kiến thức tốt hơn về văn bản trong hình ảnh. TAP sử dụng kiến trúc encoder-decoder với bộ sinh Transformer để sinh câu trả lời tự nhiên từ thông tin hình ảnh và văn bản.

Gần đây, các mô hình nền quy mô lớn như BLIP-2 đã mở ra một hướng đi mới cho bài toán VQA và TextVQA. BLIP-2 kết hợp encoder ảnh frozen (ViT-G/14) và LLM như FlanT5, cho phép mô hình hiểu và tạo văn bản ở cấp độ ngữ nghĩa cao, đồng thời dễ dàng thích ứng với các ngôn ngữ và nhiệm vụ khác nhau.

## Các mô hình Text-VQA cho tiếng Việt

So với tiếng Anh, tiếng Việt là một ngôn ngữ có ít tài nguyên hơn và vẫn còn hạn chế trong lĩnh vực VQA. Một đóng góp đáng chú ý là **ViTextVQA** , một bộ dữ liệu quy mô lớn chứa các ảnh thực tế có chứa văn bản tiếng Việt (bảng hiệu, tài liệu, hóa đơn,...). ViTextVQA cung cấp gần 30.000 câu hỏi tương ứng với 10.000 hình ảnh, giúp mở ra hướng nghiên cứu cho VQA tiếng Việt.

Bên cạnh đó, một số mô hình thử nghiệm như **LaVy** và **Vista** được xây dựng như benchmark cho Visual Reasoning đa ngôn ngữ. Các mô hình này sử dụng kiến trúc encoder ảnh như VinVL hoặc ViT, kết hợp với các mô hình ngôn ngữ mBERT hoặc PhoBERT để tạo pipeline suy luận thị giác-ngôn ngữ cho tiếng Việt. Tuy nhiên, các nghiên cứu này vẫn còn giới hạn về độ chính xác và chưa tận dụng được các mô hình encoder hình ảnh mạnh mẽ hơn như InternViT hoặc CLIP-ViT.

## Các kiến trúc trích xuất đặc trưng ảnh

Việc lựa chọn kiến trúc trích xuất đặc trưng ảnh (image feature extractor) đóng vai trò then chốt trong hiệu quả của mô hình Text VQA. Qua các nghiên cứu trước đây, có ba dòng kiến trúc chính đã và đang được ứng dụng phổ biến:

### Các kiến trúc CNN truyền thống

Các mô hình như **ResNet** và **EfficientNet** từng là backbone tiêu chuẩn cho nhiều hệ thống thị giác. Chúng có khả năng trích xuất đặc trưng không gian tốt nhưng lại thiếu khả năng biểu diễn ngữ nghĩa sâu và quan hệ toàn cục giữa các vùng ảnh – yếu tố quan trọng trong bài toán Text VQA. Vì lý do đó, các mô hình CNN dần được thay thế bởi kiến trúc Transformer.

### Các mô hình Vision Transformer (ViT)

Kiến trúc **ViT (Vision Transformer)** là bước ngoặt lớn khi lần đầu tiên đưa cơ chế attention thuần túy vào xử lý ảnh. ViT chia ảnh thành các patch và xử lý tương tự như chuỗi từ trong NLP. Các biến thể như **ViT-B/16** hay **Swin Transformer** cho thấy hiệu quả vượt trội trong các nhiệm vụ thị giác. ViT giúp mô hình hiểu được toàn cục nội dung hình ảnh, đặc biệt hữu ích khi ảnh chứa các vùng văn bản nhỏ và rải rác.

### Các kiến trúc lớn tiền huấn luyện đa nhiệm (Foundation Vision Models)

Gần đây, các mô hình quy mô cực lớn như **InternViT** đã được huấn luyện trước trên hàng trăm triệu ảnh với đa dạng nhiệm vụ như phân loại, detection, segmentation,... InternViT-300M-448px, một biến thể lớn của InternViT, có khả năng trích xuất đặc trưng giàu ngữ nghĩa và tổng quát, cho phép áp dụng hiệu quả trong các nhiệm vụ downstream như Text VQA mà không cần huấn luyện lại toàn bộ mô hình.

Các mô hình như **CLIP-ViT** – huấn luyện bằng ghép cặp hình ảnh và văn bản – cũng là lựa chọn tiềm năng nhờ khả năng ánh xạ hình và chữ vào cùng một không gian ngữ nghĩa.

## Hướng nghiên cứu đề xuất

Dựa trên các nghiên cứu kể trên, hướng tiếp cận của đề tài này là áp dụng **mô hình InternViT-300M-448px** làm encoder hình ảnh nhằm tận dụng khả năng trích xuất đặc trưng mạnh mẽ của mô hình Transformer học sâu. Các đặc trưng ảnh sau đó sẽ được kết hợp với token hóa câu hỏi để đưa vào một mô hình sinh câu trả lời (decoder). Mô hình được huấn luyện trên **ViTextVQA** – bộ dữ liệu gốc được xử lý lại phù hợp cho pipeline huấn luyện sequence-to-sequence và đánh giá trên tập dữ liệu mà nhóm thu thập được.

Cách tiếp cận này kỳ vọng sẽ đạt hiệu suất cao hơn so với các mô hình hiện tại nhờ vào kiến trúc encoder mạnh và khả năng hiểu tốt nội dung thị giác kết hợp văn bản.

# **CƠ SỞ LÝ THUYẾT**

## Visual Question Answering (VQA)

Visual Question Answering (VQA) là một lĩnh vực thuộc trí tuệ nhân tạo, kết hợp giữa xử lý ảnh (thị giác máy tính) và xử lý ngôn ngữ tự nhiên. Mục tiêu của bài toán là xây dựng một hệ thống có thể trả lời câu hỏi bằng văn bản, dựa trên nội dung trong một hình ảnh.

Để làm được điều này, hệ thống cần hiểu được nội dung của ảnh (ví dụ như các đối tượng trong ảnh, vị trí, mối quan hệ giữa các đối tượng) và đồng thời hiểu được ý nghĩa của câu hỏi đầu vào. Sau đó, hệ thống phải kết hợp thông tin từ cả hai nguồn để đưa ra câu trả lời hợp lý.

Một hệ thống VQA thường bao gồm ba thành phần chính:

* Trích xuất đặc trưng ảnh: Sử dụng các mô hình học sâu như CNN hoặc Transformer để lấy thông tin từ ảnh đầu vào.
* Mã hóa câu hỏi: Sử dụng các mô hình như LSTM hoặc các biến thể của Transformer để biểu diễn câu hỏi dưới dạng vector.
* Kết hợp và suy luận: Áp dụng các kỹ thuật hợp nhất thông tin từ ảnh và câu hỏi để suy luận ra câu trả lời. Text VQA (Text-based Visual Question Answering)

## Text-based Visual Question Answering (Text VQA)

TextVQA là một biến thể đặc biệt của VQA, trong đó hình ảnh chứa văn bản là yếu tố quan trọng để trả lời câu hỏi. Khác với VQA thông thường – nơi hệ thống có thể chỉ cần nhận diện vật thể – thì trong TextVQA, hệ thống cần đọc và hiểu được văn bản nằm trong ảnh, sau đó kết hợp với câu hỏi để trả lời.

Ví dụ, với một ảnh chụp bảng hiệu và câu hỏi “Tên cửa hàng là gì?”, hệ thống cần đọc được chữ trên bảng hiệu và đưa ra câu trả lời chính xác. Điều này khiến TextVQA có thêm một thách thức lớn: trích xuất và hiểu văn bản trong ảnh, hay còn gọi là OCR (Optical Character Recognition).

Quá trình xử lý bài toán TextVQA thường gồm các bước:

* Tiền xử lý và nhận diện văn bản trong ảnh (OCR).
* Trích xuất đặc trưng của ảnh và câu hỏi.
* Hợp nhất thông tin từ hình ảnh, văn bản trích xuất và câu hỏi để đưa ra câu trả lời

## Nhận diện văn bản trong ảnh (OCR – Optical Character Recognition)

OCR là kỹ thuật chuyển đổi văn bản trong ảnh thành chuỗi ký tự có thể xử lý được bằng máy tính. Trong bối cảnh tiếng Việt, OCR gặp thách thức do sự đa dạng của phông chữ, dấu thanh và bố cục văn bản phức tạp. Các thư viện như **Tesseract**, **PaddleOCR** hay **VietOCR** được sử dụng phổ biến nhờ khả năng hỗ trợ tiếng Việt tốt. OCR đóng vai trò quan trọng trong Text VQA vì nhiều thông tin cần thiết nằm trong phần văn bản của ảnh.

## InternViT

InternViT‑300M‑448px là phiên bản “rút gọn” của InternViT‑6B, được distill xuống 304 triệu tham số để tăng tốc suy diễn nhưng vẫn giữ sức mạnh của mô hình gốc. Nó chấp nhận ảnh đầu vào 448 × 448 pixel và hỗ trợ ghép tile linh hoạt – 1‑12 tile khi huấn luyện, mở rộng đến 40 tile khi suy diễn – giúp xử lý cảnh có độ phân giải cao mà không tốn nhiều bộ nhớ. Nhờ huấn luyện trên tập dữ liệu khổng lồ kết hợp các bộ OCR tiếng Anh và tiếng Trung, InternViT‑300M thể hiện khả năng nhận dạng văn bản tốt, độ bền cao trước nhiễu, và đặc biệt phù hợp làm backbone cho các mô hình ngôn ngữ‑hình ảnh (MLLM) nguồn mở.

## Các thành phần kỹ thuật liên quan

### Convolutional Neural Network (CNN)

CNN là mạng nơ-ron tích chập thường được dùng để xử lý ảnh. Mạng này có khả năng trích xuất các đặc trưng quan trọng trong ảnh như cạnh, hình dạng và kết cấu. Các mô hình phổ biến như ResNet, EfficientNet, hoặc ViT (Vision Transformer) thường được sử dụng trong các hệ thống VQA hiện đại.

### LSTM (Long Short-Term Memory)

LSTM là một dạng mạng hồi tiếp (RNN) được cải tiến, có khả năng ghi nhớ các thông tin dài hạn trong chuỗi dữ liệu. Trong bài toán VQA, LSTM thường được sử dụng để mã hóa câu hỏi đầu vào, giúp mô hình hiểu ngữ nghĩa của từng từ và toàn bộ câu.

### Transformer

Transformer là kiến trúc mạng đã thay đổi hoàn toàn cách xử lý dữ liệu tuần tự. Khác với RNN, Transformer sử dụng cơ chế attention để xác định mối liên hệ giữa các phần tử trong chuỗi. Transformer hiện được áp dụng rộng rãi trong cả xử lý ngôn ngữ và thị giác máy tính (ví dụ như ViT cho ảnh, BERT hay GPT cho văn bản).

### Attention Mechanism

Attention giúp mô hình tập trung vào những phần thông tin quan trọng hơn trong ảnh hoặc văn bản. Trong VQA, attention thường được dùng để xác định vùng ảnh hoặc từ ngữ nào có liên quan nhiều nhất đến câu hỏi đang được xét.

## Các phương pháp đánh giá mô hình

Để đánh giá độ chính xác và mức độ phù hợp của mô hình trong việc sinh câu trả lời, nhóm chúng em sử dụng các chỉ số phổ biến như Accuracy và BLEU score. Các chỉ số này giúp xác định mức độ chính xác và phù hợp của mô hình trong việc tạo ra câu trả lời cho các câu hỏi dựa trên hình ảnh chứa văn bản.

### BLEU (Bilingual Evaluation Understudy)

BLEU là chỉ số đánh giá độ tương đồng n-gram giữa câu trả lời của mô hình và một hoặc nhiều câu trả lời tham chiếu. BLEU thường được sử dụng trong các tác vụ sinh câu (text generation) và dịch máy (machine translation).

BLEU tính đến độ dài và thứ tự của các cụm từ, tuy nhiên nó có thể không phản ánh chính xác ý nghĩa khi câu trả lời đúng có nhiều cách diễn đạt.

Trong đó:

* p\_n: Độ chính xác của n-gram, p\_n = (số n-gram khớp) / (tổng số n-gram trong câu trả lời)
* w\_n: Trọng số cho mỗi n-gram, thường w\_n = 1/N (với N = 4)

### ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) là một tập hợp các chỉ số đánh giá hiệu quả của hệ thống sinh văn bản, đặc biệt phổ biến trong các bài toán tóm tắt, sinh mô tả ảnh, và hỏi đáp tự động. Khác với BLEU (dựa trên độ chính xác - precision), ROUGE tập trung vào độ bao phủ (recall), đo lường tỷ lệ từ hoặc cụm từ (n-gram) trong câu trả lời tham chiếu được mô hình tái hiện.

Công thức cơ bản của ROUGE-N được tính như sau:

Trong đó:

* Số n-gram khớp: Số lượng n-gram (thường từ 1-gram đến 4-gram) xuất hiện đồng thời trong câu trả lời của mô hình và câu trả lời tham chiếu.
* Tổng số n-gram trong câu trả lời tham chiếu: Tổng số n-gram có trong câu trả lời tham chiếu.

Ngoài ROUGE-N, các biến thể khác như ROUGE-L (dựa trên chuỗi con chung dài nhất - Longest Common Subsequence) và ROUGE-S (dựa trên skip-bigram) cũng được sử dụng để đánh giá tính liên tục và ngữ nghĩa của văn bản.

# **MÔ HÌNH ĐỀ XUẤT**

## Mô hình dựa trên InternVit và LSTM

Trong các nghiên cứu gần đây về hệ thống trả lời câu hỏi trực quan (Visual Question Answering – VQA), đặc biệt là **Text VQA** – nơi mà câu hỏi liên quan trực tiếp đến nội dung văn bản trong hình ảnh – việc sử dụng các mô hình thị giác mạnh mẽ để trích xuất đặc trưng ảnh đóng vai trò vô cùng quan trọng. Mô hình thị giác tiên tiến như **Vision Transformer (ViT)** đã chứng minh hiệu quả vượt trội trong việc nắm bắt ngữ cảnh toàn cục của hình ảnh, đặc biệt là với ảnh chứa nhiều văn bản phân bố không đồng đều. Trên cơ sở đó, **nhóm chúng em đề xuất mô hình Text VQA cho tiếng Việt** với các thành phần chính như sau:

### Khối trích xuất đặc trưng thị giác (Vision Backbone)

* + Sử dụng **InternViT-300M** (ảnh vào 448 px) đã được tiền-huấn luyện để sinh ra chuỗi đặc trưng không gian của ảnh.
  + Một lớp **Linear projection** nén chuỗi đặc trưng này thành **Image Feature Vector** cố định, phù hợp kích thước khi kết hợp với đặc trưng câu hỏi.

### Khối mã hoá câu hỏi (Question Encoder)

* + Câu hỏi tiếng Việt được token-hoá và ánh xạ qua **Question Encoder Embedding**.
  + Một lớp LSTM theo dõi ngữ cảnh tuần tự và xuất **Question Feature Vector** biểu diễn toàn bộ câu hỏi.

### Khối kết hợp đặc trưng (Feature Fusion)

* + **Ghép nối (Concatenate)** hai vector đặc trưng ảnh và câu hỏi để tạo nên biểu diễn ngữ nghĩa chung, giữ lại cả thông tin thị giác lẫn ngôn ngữ.

### Bộ giải mã câu trả lời (Decoder)

* + Một **LSTM** được khởi tạo bằng vector hợp nhất, sinh tuần tự các token đáp án.
  + Lớp **Linear + Softmax** chuyển hidden state thành phân phối xác suất trên từ vựng, cho ra đáp án cuối cùng

### Kiến trúc mô hình

A group of colorful rectangular shapes

AI-generated content may be incorrect.

1. *Kiến trúc mô hình dựa trên InternViT và LSTM*

## Mô hình dựa trên InternViT, PhoBert và cơ chế fusion–decoder generative

1. **Image Encoder – InternViT**
   * Đầu vào ảnh: Ảnh RGB được resize và chuẩn-hoá về kích thước 3 × 448 × 448. Tensor đầu vào lúc này có dạng B × 3 × 448 × 448.
   * Patch Embedding: Khối Patch-Embedding cắt ảnh thành 196 patch 16 × 16 px, thêm 1 token [CLS] ⇒ tổng 197 token. Mỗi patch (3 × 16 × 16) được làm phẳng rồi qua một lớp Linear để trở thành vector nhúng 768 chiều. Kết quả: B × 197 × 768.
   * InternViT Transformer Blocks: Chuỗi 197 token chạy qua N khối self-attention của mô hình InternViT-300M-448px. Trong fine-tune, chúng em chỉ “mở khóa” (trainable) block cuối, các khối còn lại giữ nguyên trọng số tiền-huấn luyện nhằm tiết kiệm tài nguyên nhưng vẫn khai thác được ngữ cảnh toàn cục của ảnh.
   * Linear Projection + Positional Encoding: Đầu ra từ InternViT (vẫn B × 197 × 768) đi qua một lớp Linear projection (nếu cần) để đồng bộ hidden\_dim, rồi được cộng Positional Encoding (learned hoặc sinusoidal) nhằm duy trì thông tin vị trí patch trong chuỗi embedding.
   * Đầu ra khối thị giác: Thu được Image Feature Sequence kích thước B × 197 × 768. Đây là biểu diễn không gian giàu ngữ cảnh của toàn ảnh và sẽ được chuyển sang FusionBlock để kết hợp với đặc trưng câu hỏi trong giai đoạn tiếp theo.
2. **Text Encoder (PhoBert)**
   * Đầu vào câu hỏi: Câu hỏi tiếng Việt được token-hoá và padding/truncation về độ dài cố định MAX\_LEN. Tensor đầu vào có dạng B × L (với L ≤ MAX\_LEN), gồm chỉ số từ vựng và mặt nạ attention.
   * PhoBERT Embedding & Transformer Layers: Chuỗi token đi qua bộ nhúng PhoBERT để nhận vector ẩn 768 chiều, sau đó chạy qua 12 lớp Transformer của PhoBERT-base. Vì PhoBERT đã pre-train trên corpora tiếng Việt lớn, toàn bộ trọng số được đóng băng (freeze) nhằm giữ tri thức ngôn ngữ và giảm chi phí huấn luyện.
   * Projection + Positional Encoding: Đầu ra PhoBERT (B × L × 768) được đưa qua Linear projection (nếu cần) để đồng bộ với hidden\_dim của mô hình tổng thể. Tiếp theo, cộng thêm Positional Encoding (learned) để mô hình vẫn nhận biết vị trí tuần tự của các token trong câu hỏi.
   * Đầu ra khối Text Encoder: Thu được Question Feature Sequence có kích thước B × L × hidden\_dim. Chuỗi đặc trưng này mang thông tin ngữ nghĩa đầy đủ của câu hỏi và sẽ được chuyển vào FusionBlock để kết hợp với đặc trưng ảnh.
3. **Khối FusionBlock - Kết hợp thị giác & ngôn ngữ**
   * Ghép đặc trưng ban đầu: Chuỗi đặc trưng câu hỏi txt\_feats (B × N\_txt × D) và chuỗi đặc trưng ảnh img\_feats (B × N\_img × D) được nối dọc trục thời gian thành một dãy duy nhất [txt; img] có kích thước B × (N\_txt + N\_img) × D. Việc ghép này bảo toàn toàn bộ thông tin của hai miền.
   * Self-Attention toàn cục: Dãy kết hợp trên được đưa qua một lớp Multi-Head Self-Attention. Cơ chế attention ở bước này cho phép mỗi token (cả text lẫn image) “nhìn” toàn bộ chuỗi, giúp mô hình mô hình hóa quan hệ chung - ví dụ một từ có thể chú ý tới patch ảnh liên quan và ngược lại.
   * Cross-Attention có điều hướng: Sau khi self-attn, dãy được tách lại thành hai nhánh. Phần text được dùng làm truy vấn (Q), trong khi phần image đóng vai trò khóa–giá trị (K,V). Lớp Cross-Attention (Q = txt, K/V = img) giúp mỗi từ trong câu hỏi tập trung chính xác vào các vùng ảnh (patch) hữu ích để trả lời, thay vì phải tìm trong toàn bộ chuỗi.
   * Feed-Forward & LayerNorm: Kết quả cross-attention của nhánh text đi qua mạng FFN 2 lớp (hidden → 4×hidden → hidden) và LayerNorm để tăng khả năng biểu diễn và ổn định gradient. Phía image giữ nguyên (chỉ chịu ảnh hưởng của self-attn).
   * Đầu ra & Mask: Cuối cùng, hai nhánh được nối lại tạo thành memory sequence kích thước B × (N\_txt + N\_img) × D, kèm một memory\_key\_padding\_mask (mask pad của text, image không mask). Memory này sẽ được Transformer Decoder sử dụng để sinh đáp án.
4. **Khối Generative Decoder – sinh đáp án**
   * Đầu vào - Answer Input IDs

* Ở giai đoạn huấn luyện: chuỗi đáp án thật đã dịch trái 1 bước (BOS + ground-truth, bỏ token cuối).
* Ở suy luận: chuỗi khởi đầu chỉ chứa token <s> (BOS).

=> Tensor: B × T.

* + Nhúng & Positional Encoding
* Mỗi ID được ánh xạ qua Embedding layer (vocab\_size × hidden\_dim = 768).
* Cộng Positional Encoding (learned) để gìn giữ thứ tự.

=> Kết quả: B × T × 768.

* + Các lớp Transformer Decoder
* Self-attention có mặt nạ nhân quả: bảo đảm mỗi bước chỉ “nhìn” các token trước nó.
* Cross-attention với memory sequence đến từ khối Fusion Block, cho phép mỗi bước sinh tham chiếu cả thông tin ảnh lẫn câu hỏi.
* Áp dụng pad-mask (tgt\_key\_padding\_mask) và mask của memory (memory\_key\_padding\_mask) để bỏ qua vị trí padding.
* Đầu ra: B × T × 768.
  + Chiếu tuyến tính → Vocab logits
* Hidden-state sau lớp cuối được đưa qua Linear (768 → vocab\_size) sinh ma trận logits.
* Softmax (áp dụng ngoài hàm loss hoặc trong khối Sequence Generation) chuyển logits thành xác suất token.
  + Đầu ra khối Generative Decoder:
* Logits (B × T × vocab\_size): Dùng cross-entropy với đáp án thật để tính loss.
* Suy luận: Logits được truyền sang khối Sequence Generation (Greedy / Beam / Top-k) để lấy chuỗi token đáp án.

1. **Khối Sequence Generation**
   * Khởi tạo: Sau khi nhận được memory sequence từ Decoder, quá trình sinh bắt đầu với token <s> (BOS); hidden state đầu tiên của LSTM/Transformer decoder đã mang thông tin hợp nhất giữa ảnh và câu hỏi.
   * Chiến lược tìm kiếm

* Greedy search: Chọn token có xác suất cao nhất tại mỗi bước; nhanh nhưng dễ đơn điệu.
* Beam search: Giữ song song k chuỗi ứng viên tốt nhất, cộng log-probability; cân bằng giữa chất lượng và tốc độ (mặc định beam\_size = 5).
* Top-k / Top-p sampling: Lấy ngẫu nhiên trong k token xác suất cao nhất (hoặc tập xác suất tích lũy p); tăng đa dạng câu trả lời.
* Trong mã, nếu top-k sampling sinh chuỗi rỗng, thuật toán tự động fallback → beam search.
  + Tiêu chí dừng
* Sinh lặp cho tới khi gặp </s> (EOS) và ít nhất 2 token không đặc biệt đã được tạo, hoặc đạt giới hạn max\_len (mặc định 32/64).
  + Hậu xử lý
* Chuỗi token ID được tokenizer giải mã thành văn bản tiếng Việt, loại bỏ token đặc biệt và padding – đây chính là Output Answer Tokens.

1. **Kiến trúc mô hình**

A diagram of a computer

AI-generated content may be incorrect.

1. *Kiến trúc mô hình dựa trên InternViT, PhoBert và cơ chế fusion–decoder generative*

## Mô hình finetune dựa trên VinTern

* + Mô hình Vintern-1B-v2 là một mô hình đa phương thức tối ưu cho tiếng Việt, kết hợp InternViT-300M-448px (vision) và Qwen2-0.5B-Instruct (language), được fine-tune trên hơn 3 triệu cặp hình ảnh-câu hỏi-câu trả lời từ các bộ dữ liệu như Viet-Doc-VQA, tập trung vào bài toán Text VQA trong giáo dục. Với 1 tỷ tham số và độ dài ngữ cảnh 4096, mô hình hiệu quả và dễ dàng fine-tune trên T4 GPU. Dưới đây là 3 khối chính của mô hình:

1. **Image Encoder – InternViT**
   * **Đầu vào**: Ảnh RGB được chuẩn hóa về kích thước 448 × 448 pixel.
   * **Patch Embedding**: Ảnh được chia thành 196 patch (16 × 16 pixel) và 1 token [CLS], tổng 197 token. Mỗi patch được ánh xạ thành vector 768 chiều qua lớp Linear.
   * **Transformer Blocks**: Chuỗi token được xử lý qua các khối self-attention của InternViT-300M-448px. Trong fine-tune, chỉ khối cuối được huấn luyện để tiết kiệm tài nguyên, tận dụng khả năng nhận diện văn bản từ dữ liệu tiền huấn luyện.
   * **Đầu ra**: Chuỗi đặc trưng hình ảnh, biểu diễn ngữ cảnh của trang sách giáo khoa, sẵn sàng kết hợp với đặc trưng câu hỏi.
2. **Text Encoder – Qwen2-0.5B-Instruct**
   * **Đầu vào**: Câu hỏi tiếng Việt được token hóa, giới hạn độ dài tối đa 100 token.
   * **Embedding & Transformer**: Token được ánh xạ thành vector 768 chiều, xử lý qua các lớp Transformer của Qwen2-0.5B-Instruct. Trọng số được đóng băng để giữ tri thức tiếng Việt từ quá trình instruction-tuning.
   * **Đầu ra**: Chuỗi đặc trưng câu hỏi, chứa thông tin ngữ nghĩa, sẵn sàng cho khối Fusion.
3. **Fusion & Generative Decoder**
   * **Fusion**

* Đặc trưng ảnh và câu hỏi được kết hợp qua MLP projector và xử lý bằng Multi-Head Self-Attention (4 head) để tương tác toàn cục. Cross-Attention (text làm truy vấn, ảnh làm khóa/giá trị) giúp câu hỏi tập trung vào vùng ảnh liên quan trong sách giáo khoa.
* Kết quả tạo chuỗi đặc trưng tổng hợp (memory sequence).
  + **Generative Decoder**
* Đầu vào: Chuỗi đáp án (huấn luyện: bắt đầu bằng BOS và dịch trái; suy luận: chỉ chứa <s>).
* Token được nhúng, cộng Positional Encoding, và xử lý qua Transformer Decoder của Qwen2-0.5B-Instruct, sử dụng self-attention (mặt nạ nhân quả) và cross-attention với memory sequence để sinh token.
* Chiếu tuyến tính: Hidden state được ánh xạ thành logits, áp dụng Softmax để lấy xác suất token.
* Sequence Generation: Sử dụng beam search (beam\_size=5) để sinh câu trả lời, dừng khi gặp </s> hoặc đạt độ dài tối đa (32 token). Chuỗi token được giải mã thành văn bản tiếng Việt, loại bỏ token đặc biệt.

1. **Kiến trúc mô hình**

A diagram of a diagram

AI-generated content may be incorrect.

1. *Kiến trúc mô hình dựa trên InternViT, PhoBert và cơ chế fusion–decoder generative*

# **THỰC NGHIỆM**

## Môi trường thực nghiệm

Để đảm bảo quá trình thực nghiệm diễn ra hiệu quả và tận dụng tốt khả năng tính toán, dự án được triển khai trên nền tảng Google Colab với cấu hình phần cứng và phần mềm như sau:

* **Cấu hình phần cứng:**
  + Nền tảng: Google Colab, Kaggle
  + Bộ xử lý đồ họa (GPU): NVIDIA Tesla T4
* **Cấu hình phần mềm:**
  + Hệ điều hành: Môi trường mặc định của Colab (Ubuntu-based Linux)
  + Ngôn ngữ lập trình: Python 3.10
  + Thư viện chính sử dụng:
    - TensorFlow/Keras: Huấn luyện mô hình học sâu.
    - NumPy và Pandas: Xử lý dữ liệu và phân tích.
    - Matplotlib/Seaborn: Trực quan hóa dữ liệu.
    - scikit-learn: Cung cấp các công cụ bổ trợ như chia tách dữ liệu, tính toán các chỉ số đánh giá mô hình.
* **Thiết lập môi trường:** Toàn bộ các thí nghiệm được thực hiện trên Google Colab và kagge, với các môi trường đã được cấu hình sẵn, giúp tiết kiệm thời gian cài đặt và tối ưu hóa việc sử dụng tài nguyên.Các thông số cấu hình bổ sung bao gồm:
  + Bộ nhớ RAM: 12.7 GB.
  + Dung lượng lưu trữ tạm thời: 112.6 GB.
  + Thời gian thực nghiệm: Khoảng 15 phút cho mỗi lần huấn luyện mô hình với khoảng 8.000 mẫu dữ liệu.
* **Lý do chọn Google Colab và kagge:**
  + Miễn phí và mạnh mẽ: Cung cấp GPU Tesla T4 miễn phí, giúp tăng tốc việc huấn luyện mô hình học sâu.
  + Dễ sử dụng: Môi trường trực tuyến dễ thao tác, không cần cài đặt phần mềm phức tạp. Các thư viện phổ biến như TensorFlow và scikit-learn có sẵn.
  + Cộng tác dễ dàng: Có thể chia sẻ tài liệu và làm việc nhóm trực tuyến qua Google Drive.
  + Quản lý tài nguyên hiệu quả: Cung cấp bộ nhớ và lưu trữ tạm thời đủ để huấn luyện mô hình mà không gặp phải vấn đề tài nguyên.
  + Hỗ trợ thư viện khoa học dữ liệu: Các thư viện như NumPy, Pandas, Matplotlib giúp xử lý dữ liệu và trực quan hóa hiệu quả.

## Tập dữ liệu

### Dữ liệu huấn luyện

Trong dự án này, nhóm sử dụng bộ dữ liệu ViTextVQA, lấy từ nền tảng Hugging Face ([dataset](https://huggingface.co/datasets/minhquan6203/ViTextVQA)). Bộ dữ liệu bao gồm 16.762 ảnh và 50.342 cặp câu hỏi – trả lời (QA), được biên soạn thủ công. Nội dung ảnh rất đa dạng, từ biển hiệu đường phố, bảng quảng cáo đến hóa đơn và tài liệu số hoá. Mỗi câu hỏi trong tập dữ liệu yêu cầu mô hình phải kết hợp khả năng nhận dạng ký tự (OCR) với hiểu ngữ nghĩa để đưa ra câu trả lời ngắn gọn và chính xác. Điều này cho phép đánh giá chính xác năng lực đọc hiểu thị giác của mô hình, thay vì chỉ dừng lại ở việc phát hiện văn bản.

### Dữ liệu kiểm thử

Nhằm kiểm tra khả năng tổng quát hóa của mô hình trong các tình huống thực tế, nhóm đã tự xây dựng một tập dữ liệu kiểm thử riêng, gồm 50 ảnh và 100 cặp câu hỏi - trả lời. Dữ liệu được thu thập từ hai nguồn chính: hình ảnh tìm kiếm từ Google và ảnh chụp thủ công tại khu vực sinh sống.

Sau quá trình thu thập, nhóm tiến hành tiền xử lý dữ liệu nhằm loại bỏ các ảnh mờ, ảnh không chứa văn bản hoặc có chất lượng không đảm bảo. Tiếp đó, các câu hỏi - trả lời được tạo thủ công dựa trên nội dung văn bản xuất hiện trong ảnh. Việc gán nhãn được thực hiện cẩn thận để đảm bảo các câu hỏi mang tính gợi mở, yêu cầu mô hình phải hiểu và suy luận thay vì chỉ nhận dạng đơn thuần. Tập dữ liệu này được sử dụng làm cơ sở để đánh giá hiệu quả của mô hình trên dữ liệu ngoài tập huấn luyện, phản ánh mức độ phù hợp với ứng dụng thực tế.

### Xử lý dữ liệu

1. **Tiền xử lý ảnh**

Nhóm sử dụng thư viện torchvision.transforms để chuẩn hóa ảnh đầu vào trước khi đưa vào mô hình. Các bước xử lý gồm:

* Resize: Đưa tất cả ảnh về kích thước cố định 224×224 để đảm bảo đồng nhất đầu vào.
* Chuyển sang tensor: Ảnh được chuyển từ định dạng PIL sang tensor để sử dụng trong PyTorch.

1. **Tiền xử lý văn bản**

Để đảm bảo dữ liệu đầu vào sạch và nhất quán, nhóm đã thực hiện tiền xử lý văn bản cho cả câu hỏi và câu trả lời. Các bước xử lý chính gồm:

* Chuyển chữ thường: Tất cả văn bản được đưa về chữ thường để tránh phân biệt chữ hoa – thường.
* Xóa ký tự không cần thiết: Loại bỏ các ký tự như dấu ngoặc, ký hiệu HTML (vd: &nbsp;), emoji và các ký tự đặc biệt không liên quan như “✓”, “•”, “★”,...
* Chuẩn hóa khoảng trắng: Xóa khoảng trắng thừa ở đầu, cuối và giữa câu.
* Chuẩn hóa dấu tiếng Việt: Sử dụng hàm text\_normalize để đảm bảo dấu được đặt đúng vị trí (vd: “oà” → “òa”).

Tách từ: Dùng word\_tokenize để tách câu thành các từ đơn, phục vụ cho huấn luyện mô hình.

1. **Tăng cường dữ liệu ảnh**

Để tăng tính đa dạng cho tập huấn luyện và giảm overfitting, nhóm áp dụng một số kỹ thuật tăng cường dữ liệu ảnh như:

* Xoay ảnh ngẫu nhiên ±30 độ (RandomRotation(30))
* Lật ảnh ngang, điều chỉnh màu sắc, cắt ảnh ngẫu nhiên

Các phép biến đổi được thực hiện bằng torchvision.transforms trong quá trình nạp dữ liệu, giúp mô hình tiếp xúc với nhiều dạng ảnh khác nhau dù chỉ huấn luyện trên tập dữ liệu ban đầu.

1. **Tăng cường văn bản**

Nhóm nghiên cứu đã tận dụng sức mạnh của công cụ Gemini để nâng cao chất lượng và sự phong phú của tập hợp câu hỏi, qua đó đảm bảo rằng dữ liệu đầu vào đạt được mức độ đa dạng tối ưu, đáp ứng yêu cầu của các ứng dụng xử lý ngôn ngữ tự nhiên hoặc học máy. Bằng cách sử dụng Gemini, nhóm đã thành công trong việc sinh ra một khối lượng dữ liệu khổng lồ, cụ thể là 25.000 dòng dữ liệu mới, bao gồm các câu hỏi được tạo tự động với sự biến đổi cấu trúc câu và ngữ cảnh nhưng vẫn giữ nguyên nghĩa của câu hỏi. Để lọc ra những dữ liệu thực sự phù hợp, nhóm đã áp dụng một ngưỡng độ tương đồng được thiết lập ở mức 0.5, nhằm đảm bảo rằng các câu hỏi mới vừa giữ được sự liên quan với tập dữ liệu gốc, vừa mang lại giá trị bổ sung. Kết quả, từ quá trình này, nhóm đã thu được hơn 3.000 dòng dữ liệu chất lượng cao, đáp ứng tiêu chí đã đề ra và sẵn sàng để sử dụng trong việc huấn luyện hoặc đánh giá mô hình.

## Ứng dụng thực nghiệm

### Quy trình huấn luyện mô hình

Mô hình được triển khai trên nền tảng Kaggle, sử dụng GPU NVIDIA Tesla P100 để tăng tốc quá trình tính toán. Quy trình huấn luyện bao gồm các bước:

* Tiền xử lý và tăng cường dữ liệu từ tập ban đầu.
* Xây dựng mô hình.
* Huấn luyện mô hình trên tập dữ liệu đã được chuẩn bị.

### Cấu hình huấn luyện

1. **Mô hình dựa trên InternVit và LSTM**

Mô hình Text VQA được huấn luyện với các tham số sau:

* **Số epoch**: 2 epoch (theo notebook, EPOCHS = 2).
* **Batch size**: 16 cho tập huấn luyện và 1 cho tập kiểm thử (theo notebook, TRAIN\_BATCH\_SIZE = 16, VAL\_BATCH\_SIZE = 1).
* **Tối ưu hóa (Optimizer)**: AdamW với learning rate mặc định (theo notebook, sử dụng AdamW từ torch.optim).
* **Hàm mất mát (Loss function)**: Sử dụng Cross-Entropy Loss, phù hợp với bài toán sinh câu trả lời dạng văn bản.
* **Metrics đánh giá**: BLEU, ROUGE-1, ROUGE-L (theo notebook, sử dụng sentence\_bleu và rouge\_scorer).

1. **Mô hình dựa trên InternViT, PhoBert và cơ chế fusion–decoder generative**

Mô hình Text VQA được huấn luyện với các tham số sau:

* **Số epoch**: 3 epoch (theo notebook, EPOCHS = 3).
* **Batch size**: 16 cho tập huấn luyện và 16 cho tập kiểm thử (theo notebook, batch\_size=16 trong DataLoader cho cả train\_loader và val\_loader).
* **Hàm mất mát (Loss function):** Cross-Entropy Loss, bỏ qua padding token (theo notebook, nn.CrossEntropyLoss(ignore\_index=tokenizer.pad\_token\_id)).
* **Metrics đánh giá:** BLEU, ROUGE-1, ROUGE-L (theo notebook, sử dụng corpus\_bleu và rouge\_scorer với rouge1, rougeL).
* **Tỷ lệ warmup:** 10% tổng số bước huấn luyện (theo notebook, WARMUP\_RATIO = 0.1).
* **Bước tích lũy gradient:** 64 bước (theo notebook, ACCUMULATION\_STEPS = 64).
* **Kích thước ảnh đầu vào:** 448 × 448 pixel (theo notebook, IMG\_SIZE = 448).
* **Độ dài tối đa chuỗi văn bản:** 80 token (theo notebook, MAX\_LEN = 80).
* **Kích thước beam search:** 3 (theo notebook, BEAM\_SIZE = 3).

1. **Mô hình finetune dựa trên VinTern**

Mô hình Text VQA được huấn luyện với các tham số sau:

* **Số epoch**: 1 (theo --num\_train\_epochs 1).
* **Weight decay**: 0.01 (theo --weight\_decay 0.01).
* **Tỷ lệ warmup**: 3% (theo --warmup\_ratio 0.03).
* **Loại scheduler**: Cosine (theo --lr\_scheduler\_type "cosine").
* **Bước logging**: 10 (theo --logging\_steps 10).
* **Bước lưu checkpoint**: 500 (theo --save\_steps 500).
* **Giới hạn số checkpoint lưu**: 2 (theo --save\_total\_limit 2).

## Kết quả thực nghiệm

1. **Mô hình dựa trên InternVit và LSTM**

Mô hình được huấn luyện trên tập dữ liệu ViTextVQA với 32,251 mẫu huấn luyện và đánh giá trên tập kiểm thử gồm 8,063 mẫu. Quá trình huấn luyện được thực hiện trên thiết bị GPU. Với 20 epoch, mỗi epoch mất khoảng 20 phút, tổng thời gian huấn luyện ước tính khoảng 7 giờ 30 phút.

## So sánh kết quả giữa các mô hình

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mô hình | BLEU | ROUGE-1 | ROUGE-2 | ROUGE-L |
| InternViT + LSTM | 0.0036 | 0.1622 | 0.0690 | 0.1311 |
| InternViT, PhoBert và cơ chế fusion–decoder generative | 0.0013 | 0.3877 | 0.0466 | 0.2195 |
| 5CD-AI/Vintern-1B-v2 | 0.1288 | 0.3247 | 0.2380 | 0.3216 |
| 5CD-AI/Vintern-1B-v2 fine-tune | 0.4471 | 0.6976 | 0.5735 | 0.6954 |

# **KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## Kết luận

Trong nghiên cứu này, nhóm chúng em đã đề xuất và phát triển một hệ thống hỏi-đáp giáo dục dựa trên bài toán Text-based Visual Question Answering (Text VQA), tập trung vào việc xử lý các trang sách giáo khoa tiếng Việt. Hệ thống được xây dựng bằng cách kết hợp các mô hình học sâu tiên tiến như InternViT, PhoBERT và VinTern, tận dụng khả năng trích xuất đặc trưng hình ảnh mạnh mẽ và xử lý ngôn ngữ tự nhiên hiệu quả cho tiếng Việt.

Quá trình thực nghiệm trên bộ dữ liệu ViTextVQA và kiểm thử trên tập dữ liệu test, bao gồm 16.762 ảnh và 50.342 cặp câu hỏi – trả lời của tập kiểm tra cùng với 50 ảnh và 100 cặp câu hỏi - trả lời của tập test. Cụ thể, mô hình VinTern-1B-v2 sau khi được tinh chỉnh (fine-tune) đạt các chỉ số đánh giá cao, với BLEU là 0.0689, ROUGE-1 là 0.6558, ROUGE-2 là 0.4745 và ROUGE-L là 0.5349. Những kết quả này khẳng định khả năng trả lời chính xác và phù hợp của hệ thống trong bối cảnh giáo dục, đặc biệt khi xử lý văn bản và hình ảnh từ sách giáo khoa.

Hệ thống không chỉ góp phần nâng cao hiệu quả học tập thông qua việc tự động hóa truy xuất thông tin mà còn mở ra tiềm năng ứng dụng thực tiễn trong các công cụ hỗ trợ giáo dục thông minh. Đề tài cũng giúp nhóm chúng em củng cố kiến thức về xử lý ngôn ngữ tự nhiên và thị giác máy tính, đồng thời tạo nền tảng cho các nghiên cứu tiếp theo trong lĩnh vực trí tuệ nhân tạo hỗ trợ ngôn ngữ tiếng Việt.

## Hướng phát triển

Dựa trên các kết quả đạt được, nhóm em đề xuất một số hướng nghiên cứu và cải tiến trong tương lai như sau:

1. **Tăng cường dữ liệu và đa dạng hóa ngữ cảnh:**
   * Thu thập thêm dữ liệu từ các nguồn tài liệu giáo dục khác nhau, như bài giảng, tài liệu tham khảo hoặc bài kiểm tra, để mô hình có thể xử lý nhiều loại câu hỏi và ngữ cảnh đa dạng hơn.
   * Áp dụng các kỹ thuật tăng cường dữ liệu tiên tiến, chẳng hạn như sử dụng mô hình ngôn ngữ lớn để tạo câu hỏi tự động hoặc dùng GAN để sinh hình ảnh tài liệu giả lập, nhằm làm phong phú dữ liệu huấn luyện.
2. **Mở rộng ứng dụng sang các lĩnh vực khác:**
   * Áp dụng hệ thống Text VQA vào các lĩnh vực ngoài giáo dục, như xử lý tài liệu hành chính, hóa đơn hoặc bảng hiệu tiếng Việt, để đáp ứng nhu cầu thực tiễn trong nhiều ngành nghề.
   * Phát triển các phiên bản mô hình hỗ trợ đa ngôn ngữ, kết hợp tiếng Việt với các ngôn ngữ khác như tiếng Anh, nhằm phục vụ người học và giáo viên trong môi trường quốc tế.
3. **Tích hợp vào các nền tảng giáo dục:**
   * Xây dựng ứng dụng di động hoặc nền tảng web tích hợp hệ thống hỏi-đáp, cho phép học sinh và giáo viên sử dụng trực tiếp trên thiết bị cá nhân, hỗ trợ học tập mọi lúc, mọi nơi.
   * Tối ưu hóa giao diện người dùng để đảm bảo tính thân thiện và dễ sử dụng, đồng thời bổ sung tính năng phản hồi để cải thiện chất lượng câu trả lời theo thời gian.
4. **Cải thiện hiệu suất mô hình:**
   * Khám phá các kiến trúc mạng mới, như Vision Transformer thế hệ tiếp theo hoặc các mô hình tự giám sát (self-supervised learning), để nâng cao khả năng trích xuất đặc trưng và hiểu ngữ nghĩa.
   * Tối ưu hóa siêu tham số và áp dụng kỹ thuật tinh chỉnh chuyên sâu để cải thiện độ chính xác và khả năng tổng quát hóa của mô hình trên các bộ dữ liệu mới.
5. **Giảm chi phí tính toán:**
   * Tối ưu hóa mô hình để hoạt động hiệu quả trên các thiết bị có tài nguyên hạn chế, như điện thoại thông minh hoặc máy tính bảng, nhằm mở rộng khả năng tiếp cận ở những khu vực có hạ tầng công nghệ hạn chế.
   * Nghiên cứu các kỹ thuật nén mô hình, như lượng tử hóa (quantization) hoặc tỉa cành (pruning), để giảm kích thước mô hình mà vẫn duy trì hiệu suất.

**TÀI LIỆU THAM KHẢO**

[1] A. Singh et al., “TextVQA: Visual Question Answering on Textual Content in Images,” in *Proc. CVPR*, 2019. [Online]. Available: <https://arxiv.org/abs/1904.08920>  
[2] Y. Wang et al., “OCR-VQA: Visual Question Answering by Reading Text in Images,” in *Proc. ICDAR*, 2021.   
[3] J. Li et al., “BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models,” *arXiv preprint*, 2023. [Online]. Available: <https://arxiv.org/abs/2301.12597>  
[4] T. Q. Nguyen et al., “ViTextVQA: A Large-Scale Visual Question Answering Dataset for Evaluating Vietnamese Text Comprehension in Images,” *arXiv preprint*, 2023.  
[5] H. T. Nguyen et al., “LaVy: A Vietnamese Multimodal Dataset and Baseline for Visual-Linguistic Reasoning,” in *Proc. VLSP Workshop*, 2022.  
[6] K. Nguyen et al., “Vista: A Vision-and-Language Benchmark Suite for Vietnamese,” *arXiv preprint*, 2022.   
[7] S. Wang et al., “InternImage: Exploring Large-Scale Vision Foundation Models with Deformable Convolutions,” in *CVPR*, 2023.

[8] K. T. Doan *et al.*, “Vintern-1B: An Efficient Multimodal Large Language Model for Vietnamese,” *arXiv.org*, 2024. https://arxiv.org/abs/2408.12480 (accessed May 18, 2025).

[9] A. Agrawal *et al.*, “VQA: Visual Question Answering,” *International Journal of Computer Vision*, vol. 123, no. 1, pp. 4–31, Nov. 2016, doi: https://doi.org/10.1007/s11263-016-0966-6.

[10] H. Tang, “Vision Question Answering System Based on Roberta and Vit Model,” *IEEE Xplore*, Oct. 01, 2022. https://ieeexplore.ieee.org/document/10009711 (accessed Aug. 15, 2023).

[11] N. D. Huynh, M. R. Bouadjenek, S. Aryal, I. Razzak, and H. Hacid, “Visual question answering: from early developments to recent advances -- a survey,” *arXiv.org*, 2025. https://arxiv.org/abs/2501.03939

[12] Devika Patadia, Shivam Kejriwal, R. Shah, and Neha Katre, “Review of VQA : Datasets and Approaches,” pp. 1–6, Dec. 2021, doi: https://doi.org/10.1109/icac353642.2021.9697283.