

ĐẠI HỌC QUỐC GIA TPHCM

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

AI23 (23TNT1), FIT@HCMUS-VNUHCM

Final Project

Đề tài: CLIP (Contrastive Language-Image Pretraining)

Môn học: Phương pháp Toán cho Trí tuệ nhân tạo

Sinh viên thực hiện:

Nguyễn Đình Hà Dương (23122002)
Nguyễn Lê Hoàng Trung (23122004)
Đinh Đức Tài (23122013)
Hoàng Minh Trung (23122014)

Giáo viên hướng dẫn:

TS. Cấn Trần Thành Trung
ThS. Nguyễn Ngọc Toàn

Ngày 26 tháng 6 năm 2025



Mục lục

1 Giới thiệu	2
2 Lý do lựa chọn	3
3 Nền tảng xây dựng mô hình CLIP	4
3.1 Bộ dữ liệu WIT (WebImageText)	4
3.1.1 Bối cảnh và thách thức của các bộ dữ liệu truyền thống	4
3.1.2 Giải pháp Natural Language Supervision	4
3.2 Kiến trúc mô hình	6
3.2.1 Image Encoder	6
3.2.2 Text Encoder	11
3.3 Tiền huấn luyện với Contrastive Learning	13
3.3.1 Tại sao lại là Contrastive Learning?	13
3.3.2 Cơ chế hoạt động của Contrastive learning trong CLIP	14
3.4 Đặc điểm của Multi-modal Embedding Space	18
4 So sánh các biến thể của CLIP	19
5 Ưu điểm, thách thức	19
6 Ứng dụng của CLIP	21
6.1 Ứng dụng CLIP trong truy vấn Ảnh - Văn bản	21
6.1.1 Bộ dữ liệu COCO 2017	21
6.1.2 Thực hiện Truy vấn Ảnh - Văn bản với CLIP	21
6.1.3 Kết quả trên tập COCO Validation	22
6.1.4 Ví dụ minh họa kết quả truy vấn	22
6.2 Ứng dụng của CLIP để phân biệt khuôn mặt, từ đó phát triển tác vụ nhận diện khuôn mặt người.	23
6.2.1 Mục tiêu	23
6.2.2 Dữ liệu	23
6.2.3 Xây dựng mô hình	23
6.2.4 Huấn luyện	24
6.2.5 So sánh và đánh giá trong tác vụ nhận diện khuôn mặt	26
7 Các mô hình tương tự CLIP	28
7.1 ALIGN	28
7.2 BLIP	29
7.3 So sánh CLIP, ALIGN và BLIP	31
8 Kết luận, đánh giá	32
8.1 Kết luận	32
8.2 Đánh giá	32
A Phụ lục	34

1 Giới thiệu

Dây là bài báo cáo cho **Final project: CLIP**, môn Phương pháp toán cho Trí tuệ nhân tạo, lớp Trí tuệ nhân tạo Khóa 2023 (23TNT1), Khoa Công nghệ thông tin, Trường Đại học Khoa học tự nhiên - Đại học Quốc gia TP.HCM. Trong bài báo cáo này, chúng tôi sẽ tóm tắt, giải thích về **CLIP** - một mô hình đa phương thức thu hẹp khoảng cách giữa thị giác máy tính và xử lý ngôn ngữ tự nhiên. Chúng tôi cũng nghiên cứu một số ứng dụng, phát triển của CLIP và đánh giá về mô hình này.

Báo cáo được thực hiện bởi nhóm 6¹, gồm các thành viên:

- Nguyễn Đình Hà Dương (23122002)
- Nguyễn Lê Hoàng Trung (23122004)
- Đinh Đức Tài (23122013)
- Hoàng Minh Trung (23122014)

Đường dẫn repository Github của báo cáo: <https://github.com/ductai05/Math-For-AI> [1]

Đường dẫn demo Youtube: <https://www.youtube.com/watch?v=1G227RKnv-k> [2]

Bảng phân công nhiệm vụ cho từng thành viên:

Họ và tên	MSSV	Nhiệm vụ
Nguyễn Đình Hà Dương	23122002	- Ứng dụng CLIP trong truy vấn ảnh-văn bản. Thử nghiệm ViT. - Ứng dụng nâng cao CLIP vào phân biệt, nhận diện khuôn mặt.
Nguyễn Lê Hoàng Trung	23122004	- Các kĩ thuật huấn luyện CLIP. Chỉnh sửa video thuyết trình. - So sánh biến thể, ưu nhược điểm CLIP.
Đinh Đức Tài	23122013	- Trình bày các mô hình tương tự CLIP. Review report. - Demo code CLIP, BLIP, ALIGN và so sánh.
Hoàng Minh Trung	23122014	- Các kĩ thuật nền tảng của CLIP. - Tìm hiểu, trình bày WIT, ResNet, ViT, Transformer.

Các thư viện và công nghệ sử dụng:

- Numpy, Pandas: thư viện Python để xử lý số học, thao tác và xử lý dữ liệu.
- transformers, torch, PIL: Các thư viện AI và xử lý hình ảnh.
- Jupyter Notebook (qua jupyter, ipykernel): Môi trường làm việc tương tác cho phép kết hợp mã thực thi, Markdown, công thức toán học và trực quan hóa.
- Git, Github, Visual Studio Code: Quản lý dự án, lưu, chia sẻ và soạn thảo mã nguồn.

¹Nhóm 6, môn Phương pháp toán cho Trí tuệ nhân tạo, lớp Trí tuệ nhân tạo Khóa 2023 (HCMUS, VNUHCM)

2 Lý do lựa chọn

Trong hai hướng dự án được cho trước, chúng tôi lựa chọn **Hướng 2: Nghiên cứu một bài báo khoa học** với lý do sau:

- Nâng cao hiểu biết về các mô hình cơ bản, quan trọng trong lĩnh vực Machine learning / Deep learning / AI.
- Hiểu cách ứng dụng của toán học và kĩ thuật lập trình trong nghiên cứu và thực tế.
- Tăng khả năng tự nghiên cứu và áp dụng tri thức mới vào thực tiễn.
- Nâng cao tư duy phản biện, phân tích, tổng hợp.

CLIP [3] (Contrastive Language-Image Pre-Training) là một mạng nơ-ron được huấn luyện trên nhiều cặp dữ liệu (ảnh, văn bản). Mô hình này có thể được hướng dẫn bằng ngôn ngữ tự nhiên để dự đoán đoạn văn bản phù hợp nhất với một hình ảnh, mà không cần được tối ưu hóa trực tiếp cho nhiệm vụ đó — tương tự như khả năng “zero-shot” (không cần huấn luyện lại) của GPT-2 và GPT-3.

CLIP đạt hiệu suất tương đương với ResNet-50 gốc trên bộ dữ liệu ImageNet theo cách “zero-shot”, mà không cần sử dụng bất kỳ ví dụ có gán nhãn nào trong 1,28 triệu ảnh ban đầu, từ đó vượt qua một số thách thức lớn trong lĩnh vực thị giác máy tính.

Paper [3], code [4] và blog [5] của CLIP được trích dẫn dưới phần tài liệu.

Lý do lựa chọn bài báo:

- CLIP là một mô hình cơ sở (foundation model) quan trọng trong lĩnh vực học đa phương thức, kết nối giữa thị giác máy tính và xử lý ngôn ngữ tự nhiên.
- CLIP có ứng dụng đa dạng: truy vấn hình ảnh bằng văn bản; phân loại hình ảnh zero-shot và fewshot; phân tích nội dung video.
- CLIP được xây dựng dựa trên các kiến thức nền tảng toán học như đại số tuyến tính, xác suất thống kê. Hiểu rõ CLIP sẽ nâng cao kiến thức về toán học.

Kế hoạch thực hiện:

1. Phân tích các nền tảng của CLIP: WIT, Contrastive learning, Vision Transformer/Transformer, Embedding Vector Space,...
2. Phân tích dữ liệu, cách huấn luyện CLIP, so sánh các mô hình.
3. Ứng dụng của CLIP: truy vấn ảnh dựa trên văn bản đầu vào (text-to-image).
4. Phân tích các mô hình VLMs tương tự CLIP: ALIGN, BLIP, ...

3 Nền tảng xây dựng mô hình CLIP

3.1 Bộ dữ liệu WIT (WebImageText)

3.1.1 Bối cảnh và thách thức của các bộ dữ liệu truyền thống

Trong nhiều năm, các mô hình thị giác máy tính hàng đầu đã được huấn luyện trên các bộ dữ liệu được gán nhãn thủ công như ImageNet (phân loại đối tượng), MS-COCO (phát hiện đối tượng và chú thích), hay OpenImages. Mặc dù những bộ dữ liệu này đã thúc đẩy sự tiến bộ vượt bậc, chúng tồn tại những hạn chế đáng kể:

- **Quy mô và chi phí hạn chế:** Việc gán nhãn thủ công hàng triệu hình ảnh là một quá trình tốn kém và mất thời gian. Điều này giới hạn quy mô của bộ dữ liệu, khiến chúng không thể bao phủ toàn bộ sự đa dạng của thế giới thị giác.
- **Phạm vi khái niệm cố định:** Các bộ dữ liệu này thường tập trung vào một tập hợp các danh mục cố định (ví dụ: 1000 lớp trên ImageNet-1K). Điều này khiến mô hình học được một bộ kỹ năng rất cụ thể và gặp khó khăn khi tổng quát hóa sang các khái niệm hoặc đối tượng chưa từng thấy trong quá trình huấn luyện. Chúng thiếu khả năng **open set recognition** (nhận diện khái niệm tổng quát).
- **Thiếu ngữ cảnh và sắc thái:** Một nhãn đơn lẻ như "mèo" không truyền tải được nhiều thông tin như một chú thích mô tả chi tiết: "một con mèo Xiêm đang nằm trên ghế sofa dưới trời nắng". Sự thiếu hụt ngữ cảnh này giới hạn độ sâu của biểu diễn mà mô hình có thể học được.

Trong khi đó, lĩnh vực xử lý ngôn ngữ tự nhiên đã chứng kiến một cuộc cách mạng nhờ các mô hình được huấn luyện trên khối lượng văn bản khổng lồ từ internet (ví dụ: GPT-3, BERT). Câu hỏi đặt ra là: liệu chúng ta có thể áp dụng triết lý tương tự - sử dụng dữ liệu "thô" từ web với sự giám sát bằng ngôn ngữ tự nhiên - để đạt được bước đột phá tương tự trong thị giác máy tính? Bộ dữ liệu WIT ra đời để trả lời câu hỏi này.

3.1.2 Giải pháp Natural Language Supervision

WIT được xây dựng để trở thành nền tảng cho việc học biểu diễn hình ảnh từ ngôn ngữ tự nhiên. Nó khác biệt rõ rệt so với các bộ dữ liệu truyền thống ở ba khía cạnh chính: quy mô, bản chất giám sát và chiến lược xây dựng.

Quy mô vượt trội WIT bao gồm **400 triệu cặp (hình ảnh, văn bản)** được thu thập từ internet. Con số này lớn hơn gấp nhiều lần so với các bộ dữ liệu học sâu thị giác tiêu chuẩn:

- So với ImageNet (khoảng 1.28 triệu hình ảnh huấn luyện), WIT lớn hơn khoảng 300 lần về số lượng cặp.
- So với MS-COCO (khoảng 118.000 hình ảnh huấn luyện), WIT lớn hơn khoảng 4000 lần.

Sự "khổng lồ" về quy mô này là một yếu tố mang tính quyết định. Nó cho phép mô hình:

- **Tiếp xúc đa dạng hơn:** Gặp gỡ hàng tỷ đối tượng, ngữ cảnh, phong cách hình ảnh, và mô tả văn bản khác nhau. Điều này giúp mô hình không chỉ học các đặc trưng cấp thấp mà còn cả các tính chất, khái niệm trừu tượng, phức tạp.
- **Giảm thiểu overfitting:** Với một lượng dữ liệu đồ sộ, khả năng mô hình "ghi nhớ" các ví dụ cụ thể hoặc các mối tương quan giả mạo trong dữ liệu huấn luyện giảm đi đáng kể, buộc nó phải học các khái niệm tổng quát hơn.
- **Học hỏi sâu sắc hơn:** Khỏi lượng dữ liệu lớn cho phép huấn luyện các mô hình có năng lực lớn (ví dụ: hàng tỷ tham số) một cách hiệu quả, khai thác tối đa tiềm năng của kiến trúc mạng sâu.

Natural Language Supervision Đây là điểm đặc biệt và cốt lõi nhất của WIT. Thay vì sử dụng các nhãn phân loại đơn lẻ (như "chó", "mèo"), WIT sử dụng **văn bản mô tả tự nhiên** đi kèm với hình ảnh (ví dụ: chú thích, mô tả, văn bản từ các trang web).

- **Tính đa dạng và khả năng khái quát hóa của khái niệm:**

- Ngôn ngữ tự nhiên có khả năng mô tả **vô số khái niệm** - từ đối tượng, hành động, thuộc tính, đến ngữ cảnh và mối quan hệ - mà không bị giới hạn bởi một danh sách cố định. Một hình ảnh có thể được mô tả cụ thể bằng "một con chó Husky đang bơi trong hồ", thay vì chỉ là "chó".
- Khả năng này là chìa khóa cho *Zero-Shot Transfer* của CLIP. Khi mô hình học cách liên kết hình ảnh với các mô tả ngôn ngữ đa dạng, nó có thể nhận diện các khái niệm mới (chưa từng thấy trong quá trình huấn luyện) chỉ bằng cách mô tả chúng bằng văn bản.

- **Tiết kiệm chi phí:**

- Dữ liệu WIT được thu thập tự động từ internet, loại bỏ nhu cầu về quá trình gán nhãn thủ công tốn kém. Điều này giúp giảm chi phí đáng kể và cho phép quy mô bộ dữ liệu tăng lên theo khả năng thu thập dữ liệu web.
- Quá trình này mô phỏng cách con người học: chúng ta học về thế giới xung quanh thông qua các giác quan và sự mô tả bằng ngôn ngữ, chứ không phải chỉ qua các nhãn cố định.

- **Chấp nhận "nhiều" tự nhiên:**

- Dữ liệu từ internet thường chứa "nhiều" và không được chọn lọc một cách hoàn hảo. Các cặp hình ảnh-văn bản có thể không luôn khớp chính xác (ví dụ: chú thích không liên quan trực tiếp đến nội dung chính của ảnh).
- Tuy nhiên, các nhà nghiên cứu nhận thấy rằng việc học từ dữ liệu nhiều này thực sự có lợi. Nó buộc mô hình phải học các biểu diễn mạnh mẽ hơn, ít bị ảnh hưởng bởi các mối tương quan giả mạo, và tổng quát hóa tốt hơn ra các phân phối dữ liệu thực tế (thường cũng không hoàn hảo). Điều này giúp CLIP có tính mạnh mẽ cao hơn đối với các dịch chuyển phân phối tự nhiên.

Chiến lược xây dựng bộ dữ liệu Mặc dù dữ liệu WIT được thu thập từ internet một cách tự động, quá trình này không hoàn toàn ngẫu nhiên mà được thực hiện một cách có chiến lược để đảm bảo tính bao quát. Các cặp hình ảnh-văn bản được tìm kiếm dựa trên một danh sách **500.000 truy vấn**. Danh sách này được xây dựng từ:

- Tất cả các từ xuất hiện ít nhất 100 lần trong phiên bản tiếng Anh của Wikipedia.
- Các bigram (cặp từ) có thông tin tương hỗ điểm cao (high pointwise mutual information).
- Tên của tất cả các bài viết Wikipedia trên một ngưỡng khối lượng tìm kiếm nhất định.
- Tất cả các tập hợp từ WordNet (WordNet synsets) chưa có trong danh sách truy vấn.

Chiến lược này đảm bảo rằng bộ dữ liệu WIT bao phủ một **phạm vi rất rộng** các khái niệm, đối tượng, và tình huống mà mọi người quan tâm và mô tả trên web, từ đó cung cấp một nền tảng kiến thức thị giác-ngôn ngữ phong phú cho mô hình.

3.2 Kiến trúc mô hình

Kiến trúc mô hình giúp CLIP học cách liên kết hình ảnh và văn bản trong một không gian nhúng chung. Thay vì phát triển một kiến trúc mới, họ đã **tận dụng và điều chỉnh các kiến trúc mạng có sẵn đạt hiệu quả cao**. Về cơ bản, kiến trúc CLIP bao gồm hai bộ mã hóa (encoders) chính hoạt động độc lập để xử lý hai loại dữ liệu khác nhau:

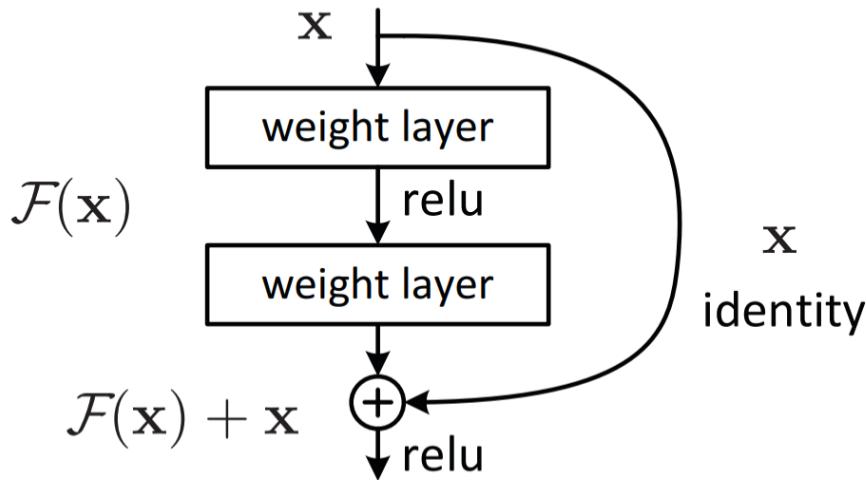
- **Bộ mã hóa hình ảnh (Image Encoder):** Chuyển đổi một hình ảnh thành một vector biểu diễn (embedding) trong không gian đa phương thức chung.
- **Bộ mã hóa văn bản (Text Encoder):** Chuyển đổi một đoạn văn bản (chú thích) thành một vector biểu diễn (embedding) trong cùng không gian đa phương thức đó.

Sau đó, các embedding vector từ cả hai bộ mã hóa này sẽ được so sánh thông qua một phép nhân vô hướng để đo lường độ tương đồng, làm cơ sở cho [Contrastive learning](#) của CLIP.

3.2.1 Image Encoder

Bộ mã hóa hình ảnh chịu trách nhiệm trích xuất các đặc trưng từ dữ liệu đầu vào. CLIP đã thử nghiệm hai dòng kiến trúc chính: **ResNet** và **Vision Transformer**.

Dòng kiến trúc ResNet (Residual Networks) ResNet (He et al., 2016a) là một kiến trúc CNN mang tính đột phá, nổi tiếng với việc sử dụng kỹ thuật *skip connections* giúp huấn luyện các mạng rất sâu:

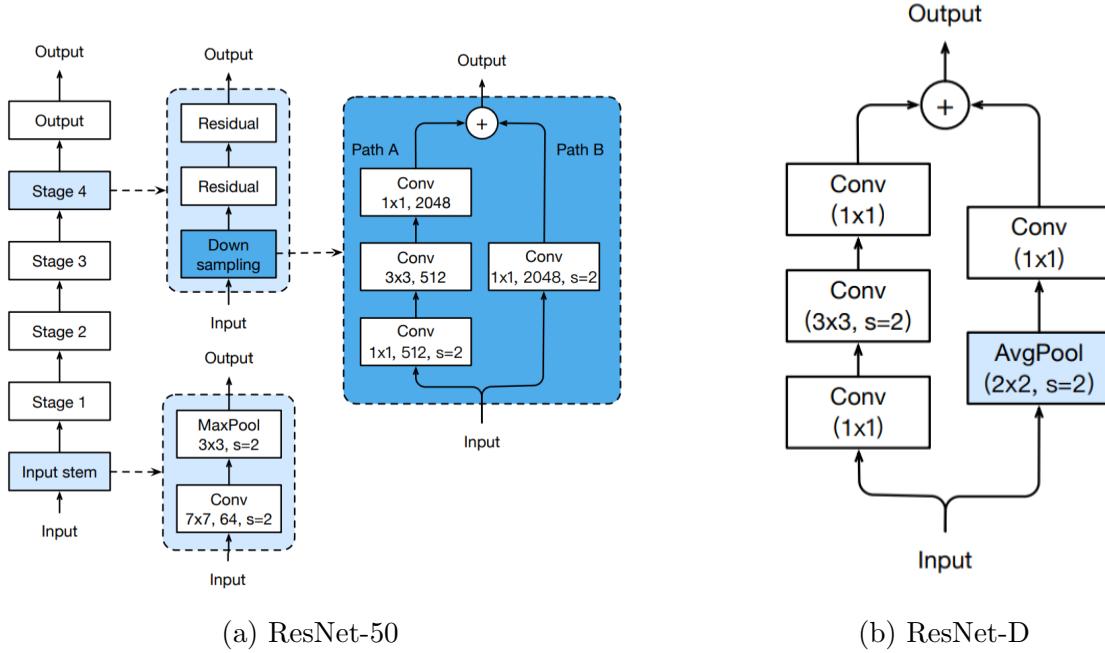


Hình 1: Kiến trúc một residual block trong ResNet. Đầu vào x được truyền trực tiếp qua một đường tắt (identity shortcut) và cộng với đầu ra $\mathcal{F}(x)$ của các lớp học trọng số. Skip connection giúp duy trì thông tin gốc, hỗ trợ truyền gradient hiệu quả và là yếu tố then chốt giúp huấn luyện mạng sâu trở nên khả thi. Khắc phục được hiện tượng *exploding/vanishing gradient*.

CLIP sử dụng ResNet-50 và Resnet-101 làm kiến trúc cơ sở và áp dụng một số cải tiến quan trọng:

1. **ResNet-D (He et al., 2019):** Họ đề xuất một số "mẹo" để cải thiện hiệu suất của ResNet. Trong số đó, các thay đổi chính của "ResNet-D" liên quan đến:

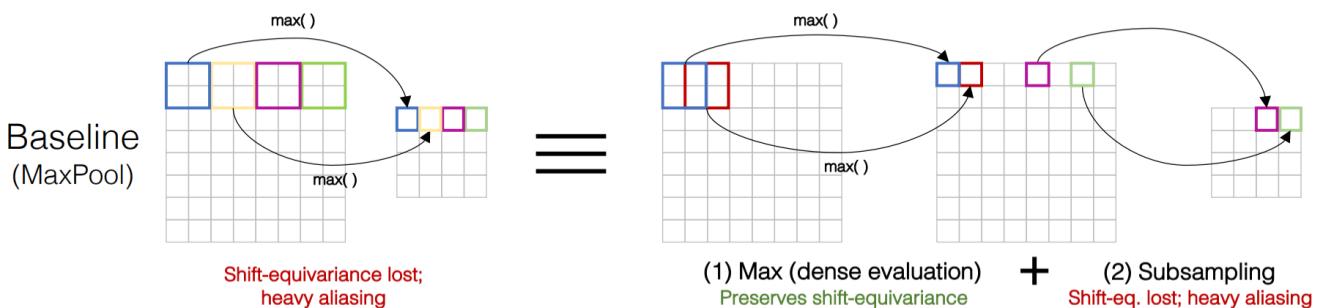
- **Đưa stride 2 từ block Conv đầu tiên lên block Conv thứ hai:** Trong ResNet gốc, block đầu tiên của mỗi giai đoạn thường sử dụng một lớp tích chập 1×1 với stride 2 để giảm kích thước. ResNet-D thay đổi nó thành một lớp tích chập 3×3 với stride 2 ở block Conv thứ hai, giúp bảo toàn thông tin tốt hơn.
- **Pooling thay vì Conv stride:** Ở nhánh shortcut connection (path B) của các block ResNet, thay vì sử dụng tích chập 1×1 với stride 2 để khớp kích thước khi downsampling, ResNet-D sử dụng lớp average pooling rồi mới đến tích chập 1×1 , giúp tránh mất thông tin và các vấn đề răng cưa (aliasing).



Hình 2: So sánh sự thay đổi của ResNet-D lên block downsampling của ResNet-50

2. **Antialiased Rect-2 Blur Pooling (Zhang, 2019):** Kỹ thuật này tích hợp các bộ lọc làm mờ (blur filters) vào các lớp downsampling.

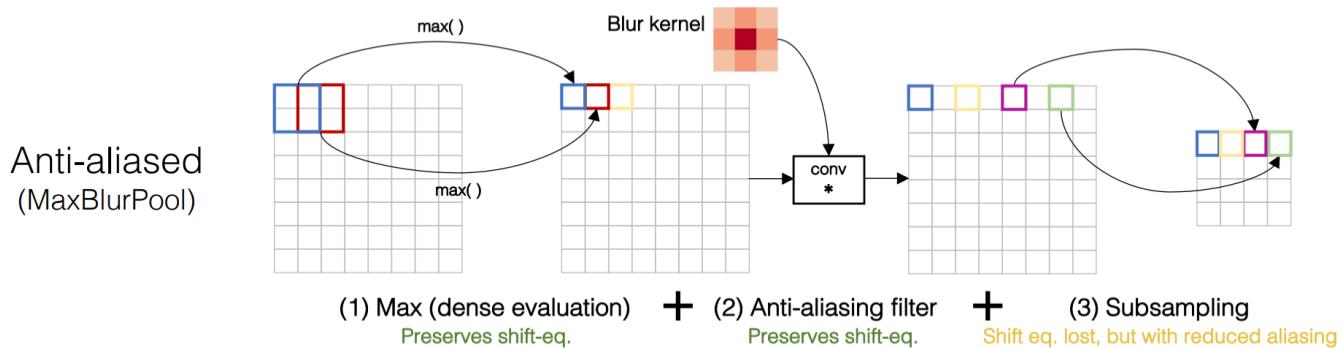
- *Vấn đề răng cưa (Aliasing):* Khi một hình ảnh hoặc bản đồ đặc trưng được downsampling, nếu không được xử lý cẩn thận, hiện tượng aliasing (hiện tượng méo mó tín hiệu do lấy mẫu dưới mức) có thể xảy ra. Hiện tượng aliasing có thể khiến mạng neural học các đặc trưng sai lệch hoặc kém bền vững. Mô hình có thể trở nên nhạy cảm với những thay đổi nhỏ trong vị trí của đối tượng, vì những thay đổi đó có thể tạo ra các "mẫu aliasing" khác nhau, làm cho mạng khó khai quát hóa.



Hình 3: Minh họa Max Pooling truyền thống, cho thấy sự mất mát của shift-equivariance và hiện tượng aliasing.

- *Giải pháp Blur Pooling:* Bằng cách áp dụng một bộ lọc làm mờ nhỏ trước khi downsampling, Blur Pooling giúp làm mịn các tín hiệu tần số cao có thể gây ra aliasing, làm cho mạng tích chập giữ được tính shift-equivariance (ít nhạy cảm hơn với các dịch chuyển).

nhỏ). Giữ lại nhiều thông tin hữu ích hơn và làm cho biểu diễn học được mạnh mẽ hơn đối với các biến thể nhỏ trong dữ liệu đầu vào.

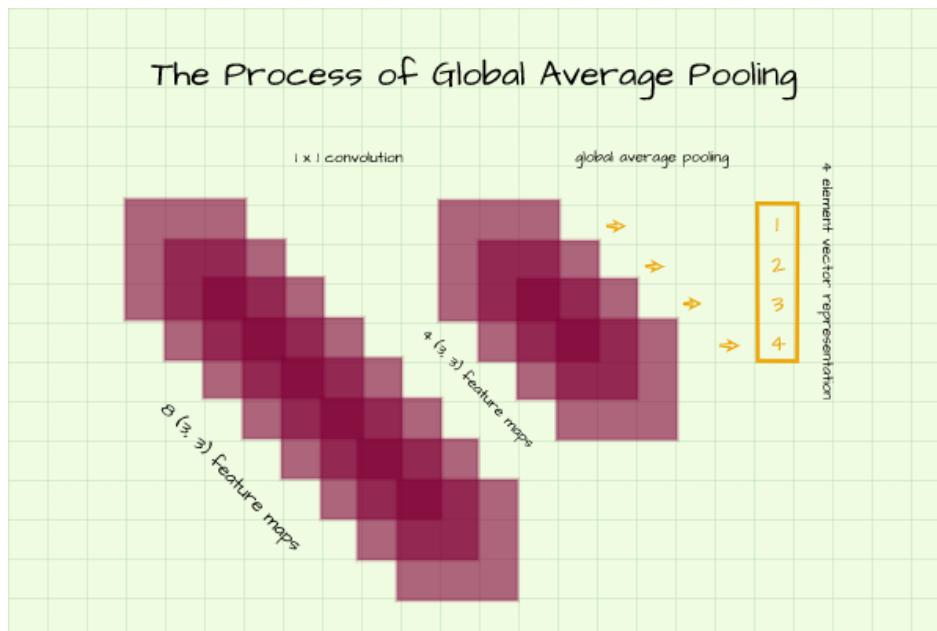


Hình 4: MaxBlurPool (Anti-aliased), giảm răng cưa bằng cách áp dụng bộ lọc làm mờ trước khi subsampling.

Như minh họa, việc mất tính shift-equivariance là không thể tránh khỏi do tính chất của subsampling nhưng bằng cách áp dụng bộ lọc làm mờ trước khi subsampling, hiện tượng aliasing đã được giảm đi đáng kể.

3. Thay thế Global Average Pooling bằng Attention Pooling:

Global Average Pooling (GAP): Thường được sử dụng ở cuối mạng CNN để tổng hợp các đặc trưng không gian thành một vector duy nhất bằng cách tính trung bình. Mặc dù đơn giản và hiệu quả trong việc giảm chiều dữ liệu, nó coi mọi thông tin trong feature map đều có mức độ quan trọng như nhau. Điều này có thể khiến mô hình bỏ qua các thông tin quan trọng hoặc tập trung vào các vùng không liên quan trong hình ảnh khi tổng hợp biểu diễn cuối cùng.



Hình 5: Mỗi feature map sẽ được lấy trung bình để output ra một số duy nhất.

Attention Pooling của CLIP: CLIP thay thế GAP bằng một cơ chế pooling dựa trên attention. Điều này cho phép mô hình học cách "chú ý" có chọn lọc đến các vùng quan trọng nhất của hình ảnh. Đây là một lớp multi-head QKV attention (Query, Key, Value) kiểu "Transformer":

- **Query (Truy vấn):** Trong trường hợp này, vector truy vấn (query) được điều kiện hóa bởi một đặc trưng được tổng hợp bằng Global Average Pooling (GAP) từ feature map đầu vào. Nghĩa là, một bản tóm tắt ban đầu của hình ảnh (từ GAP) sẽ được dùng để "hỏi" các vùng khác của hình ảnh.
- **Key (Khóa) và Value (Giá trị):** Các vector Key và Value được tạo ra từ mỗi vị trí/diểm ảnh trên bản đồ đặc trưng cuối cùng của mạng CNN (trước lớp pooling).

Cơ chế này sẽ tính toán "điểm chú ý" giữa vector Query và từng vector Key từ bản đồ đặc trưng. Các điểm chú ý này sau đó được chuẩn hóa thành trọng số, cho biết mức độ "quan trọng" của từng vùng trong hình ảnh. Cuối cùng, một vector biểu diễn duy nhất cho toàn bộ hình ảnh được tạo ra bằng cách lấy tổng có trọng số của các vector Value, với trọng số là các điểm chú ý đã tính.

Lợi ích: Cơ chế attention cho phép mô hình tập trung vào các vùng quan trọng nhất của hình ảnh khi tạo ra biểu diễn cuối cùng. Thay vì trung bình đơn giản, nó "ưu tiên" các thông tin có giá trị cao. Điều này đặc biệt có lợi cho các nhiệm vụ đa phương thức của CLIP. Khi một hình ảnh chứa nhiều đối tượng hoặc bối cảnh phức tạp.

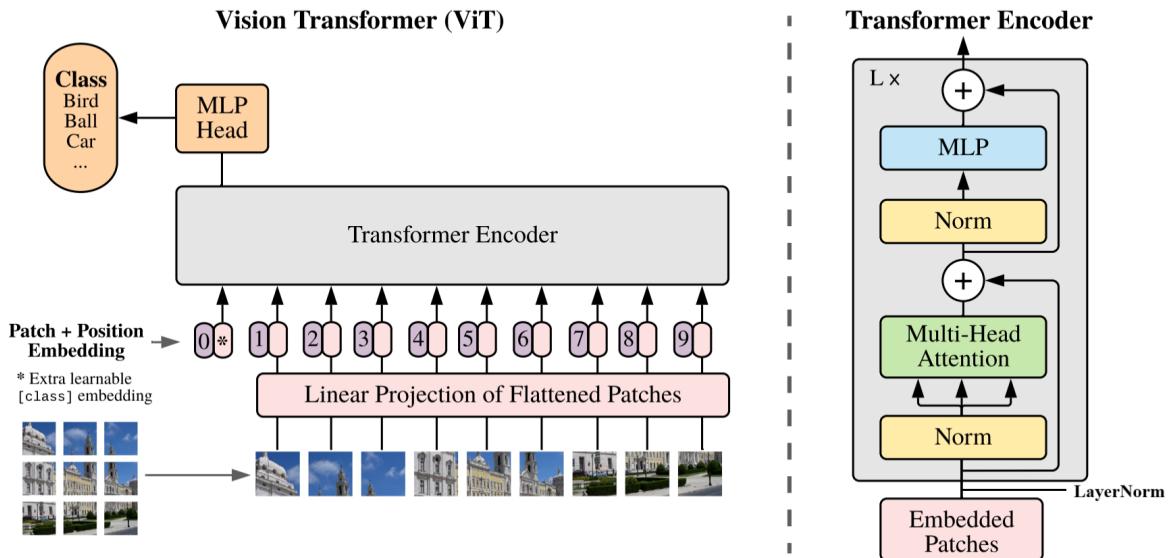
4. Chiến lược mở rộng mô hình (Model Scaling):

- *Kinh nghiệm từ EfficientNet (Tan & Le, 2019):* Thay vì chỉ mở rộng một chiều (chiều rộng, chiều sâu hoặc độ phân giải đầu vào), CLIP áp dụng một cách tiếp cận tương tự EfficientNet: phân bổ thêm năng lực tính toán để mở rộng **đồng thời cả chiều rộng, chiều sâu và độ phân giải đầu vào** của mô hình. Cách tiếp cận này được chứng minh là hiệu quả nhất để cải thiện hiệu suất với một lượng tính toán nhất định.
- *Các phiên bản ResNet của CLIP:* Từ ResNet-50 cơ sở, CLIP huấn luyện các mô hình lớn hơn, được ký hiệu là RN50x4, RN50x16 và RN50x64. Các ký hiệu này cho thấy mô hình lớn hơn tương ứng 4 lần, 16 lần và 64 lần về số lượng channel so với ResNet-50 ban đầu.

Dòng kiến trúc Vision Transformer (ViT) Vision Transformer (Dosovitskiy et al., 2020) là một kiến trúc tương đối mới tại thời điểm CLIP ra đời, chứng minh rằng Transformer (vốn được thiết kế cho NLP) cũng có thể đạt hiệu suất SOTA trong thị giác máy tính, đặc biệt khi được huấn luyện trên dữ liệu lớn. CLIP cũng khám phá và sử dụng ViT làm bộ mã hóa hình ảnh của mình.

- **CLIP tuân thủ khắt khe với việc triển khai ViT gốc. Có các sửa đổi nhỏ:**

- *Thêm lớp Layer Normalization:* Một sửa đổi nhỏ là việc thêm một lớp layer normalization bổ sung vào trước các khối Transformer, sau khi kết hợp các patch và position embeddings. Điều này có thể giúp ổn định quá trình huấn luyện và cải thiện hiệu suất.



Hình 6: Thêm LayerNorm sau Embedded Patches trước khi vào Encoder

- Áp dụng một lược đồ khởi tạo (initialization scheme) khác một chút.
- **Hiệu quả tính toán vượt trội:** Các nghiên cứu sau này (cũng như CLIP) đã chỉ ra rằng ViT thường hiệu quả tính toán hơn CNN (như ResNet) khi được huấn luyện trên các bộ dữ liệu rất lớn. Điều này cho phép CLIP đạt được hiệu suất tổng thể cao hơn trong cùng một ngân sách tính toán.
- **Các phiên bản ViT của CLIP:** Bao gồm ViT-B/32, ViT-B/16 và ViT-L/14.
- **Huấn luyện ở độ phân giải cao hơn (ViT-L/14@336px):** Để tăng cường hiệu suất (tương tự kỹ thuật FixRes), phiên bản ViT-L/14 lớn nhất còn được huấn luyện thêm một epoch ở độ phân giải hình ảnh cao hơn (336x336 pixel) sau khi đã huấn luyện ở độ phân giải tiêu chuẩn (224x224 pixel).

3.2.2 Text Encoder

Bộ mã hóa văn bản của CLIP chịu trách nhiệm biến đổi các đoạn văn bản (chú thích) thành các vector biểu diễn có ngữ nghĩa. CLIP sử dụng kiến trúc Transformer làm nền tảng cho bộ mã hóa văn bản, với một số điều chỉnh cụ thể.

Kiến trúc Transformer cơ sở Được xây dựng dựa trên kiến trúc Transformer gốc (Vaswani et al., 2017) và kết hợp các cải tiến đáng kể từ mô hình GPT-2 (Radford et al., 2019) của OpenAI. Điều này ngũ ý rằng nó là một biến thể của Transformer chỉ có decoder, được thiết kế để xử lý chuỗi văn bản.

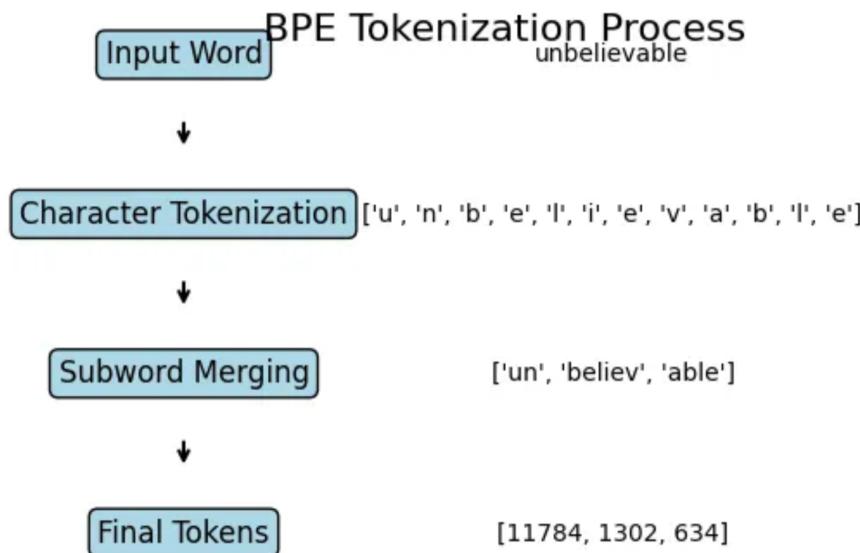
- **Thông số cấu hình điển hình:**

- *Số lớp và chiều rộng:* Phiên bản cơ sở thường là mô hình 12 lớp với chiều rộng 512 (512 chiều của các embedding bên trong các khối Transformer).

- *Attention Heads*: Sử dụng 8 head attention, cho phép mô hình tập trung vào các phần khác nhau của chuỗi đầu vào và học các mối quan hệ đa dạng giữa các từ cùng một lúc, từ đó thu thập được các khía cạnh thông tin phong phú hơn.

- **Mã hóa văn bản (Text Tokenization):**

- *Lowercase*: Đầu tiên, tất cả văn bản đầu vào được chuyển đổi sang chữ thường. Bước này giúp giảm độ phức tạp của từ vựng và làm cho mô hình ít nhạy cảm hơn với sự khác biệt về cách viết hoa/thường, coi "Dog" và "dog" là cùng một từ.
- *Byte Pair Encoding (BPE)*: Văn bản đầu vào được mã hóa thành các "tokens" bằng thuật toán BPE (Sennrich et al., 2015). BPE là một phương pháp nén dữ liệu hiệu quả, kết hợp các ký tự hoặc chuỗi ký tự phổ biến thành các tokens lớn hơn dựa theo từ điển, giúp xử lý các từ hiếm gặp (không nằm trong từ điển) và giảm kích thước từ vựng.



Hình 7: Các từ phổ biến sẽ được giữ nguyên, trong khi từ ít gặp sẽ được chia nhỏ thành các subwords phổ biến nhất.

- *Kích thước từ vựng*: Sử dụng kích thước từ vựng là 49.152.
- *Token [SOS] và [EOS]*: Mỗi trình tự văn bản được bao bọc bởi các token đặc biệt: [SOS] (Start of Sequence) ở đầu văn bản và [EOS] (End of Sequence) ở cuối văn bản. Việc sử dụng các token này giúp mô hình xác định ranh giới của văn bản đầu vào.
- *Giới hạn độ dài trình tự*: Vì hiệu quả tính toán, mỗi trình tự token hóa được giới hạn độ dài tối đa là 76 tokens. Các chuỗi dài hơn sẽ bị cắt bớt, và các chuỗi ngắn hơn sẽ được đệm để đạt đủ độ dài này.
- **Masked Self-Attention**: Một cơ chế attention trong decoder Transformer đảm bảo rằng khi mô hình xử lý một token ở vị trí t , nó chỉ có thể chú ý đến các token ở vị trí 1 đến t , và không thể "nhìn thấy" các token ở vị trí $t+1$ trở đi (các token tương lai), đảm bảo rằng việc dự đoán từ tiếp theo trong một chuỗi chỉ dựa trên ngữ cảnh đã biết (các từ trước đó).

Tuy nhiên, **CLIP không được huấn luyện để sinh văn bản**. Nhiệm vụ huấn luyện chính của CLIP là học đối lập giữa hình ảnh và văn bản, không phải là dự đoán từ tiếp theo. Sau khi mã hóa, các chuỗi đầu vào sẽ đi qua nhiều lớp Transformer. Mỗi lớp Transformer sẽ xử lý chuỗi bằng cách cơ chế Masked Self-Attention. Nhờ cơ chế này, ở mỗi lớp, mỗi token trong chuỗi đều có một vector biểu diễn ngữ cảnh riêng. Vector này không chỉ thể hiện bản thân token đó mà còn mã hóa thông tin về mối quan hệ của nó với các token trước đó trong chuỗi.

- **Trích xuất đặc trưng:**

Đây là bước then chốt. Thay vì chỉ quan tâm đến logits của từ tiếp theo, CLIP tận dụng các vector biểu diễn ngữ cảnh này.

- Biểu diễn đặc trưng của văn bản được lấy từ **biểu diễn của token [EOS] ở lớp Transformer cuối cùng (lớp cao nhất)**. Token [EOS] thường được coi là một điểm đặc biệt nơi toàn bộ ngữ cảnh của chuỗi đã được tổng hợp. Vector biểu diễn của [EOS] ở lớp cuối cùng thường được coi là một vector biểu diễn toàn diện và ngữ nghĩa của toàn bộ chuỗi văn bản đã được xử lý. Nó đã "nhìn thấy" tất cả các token trước nó và tổng hợp thông tin từ chúng.
- *Layer Normalization và Linear Projection*: Sau khi trích xuất, vector đặc trưng này được chuẩn hóa lớp và sau đó được chiếu tuyến tính vào một không gian nhúng chung. Đây là không gian mà các vector hình ảnh cũng được chiếu vào, cho phép tính toán độ tương đồng giữa văn bản và hình ảnh.

Chiến lược mở rộng quy mô: CLIP nhận thấy rằng hiệu suất tổng thể của nó không quá nhạy cảm với việc tăng số lượng tham số của bộ mã hóa văn bản. Do đó, khi mở rộng quy mô tổng thể của mô hình, họ chủ yếu tập trung vào việc mở rộng **chiều rộng** của bộ mã hóa văn bản một cách tương ứng với bộ mã hóa hình ảnh (ResNet hoặc ViT), nhưng **không tăng chiều sâu** (số lớp Transformer) một cách đáng kể. Điều này giúp tối ưu hóa hiệu quả tính toán trong khi vẫn đạt được hiệu suất mạnh mẽ.

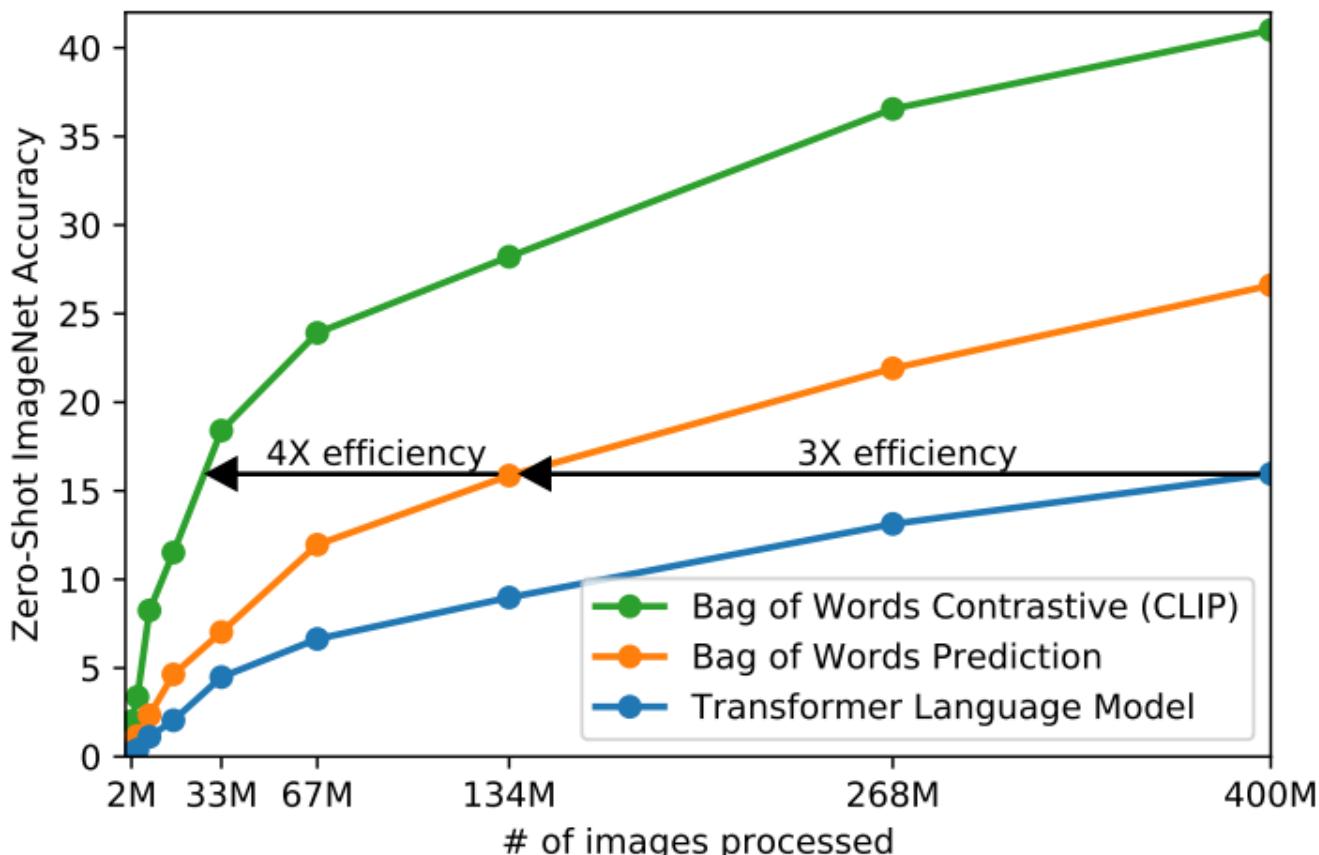
3.3 Tiền huấn luyện với Contrastive Learning

Đây là yếu tố then chốt tạo nên sự khác biệt về hiệu suất và khả năng mở rộng của CLIP so với các phương pháp trước đó trong việc học biểu diễn thị giác từ ngôn ngữ tự nhiên.

3.3.1 Tại sao lại là Contrastive Learning?

Ban đầu, một số phương pháp học biểu diễn hình ảnh từ văn bản (như VirTex) đã cố gắng xây dựng mô hình generative, nghĩa là mô hình sẽ **dự đoán chính xác chú thích** của một hình ảnh. Tuy nhiên, phương pháp này gặp phải một số thách thức lớn:

- **Độ phức tạp và tính hiệu quả:** Việc dự đoán từng từ chính xác trong một câu chú thích là một nhiệm vụ rất phức tạp đối với mô hình. Ngôn ngữ có sự đa dạng lớn, nhiều cách diễn đạt, và nhiều đáng kể trong dữ liệu web (chú thích có thể không luôn hoàn hảo hoặc liên quan trực tiếp đến hình ảnh). Điều này khiến việc huấn luyện mô hình generative trở nên kém hiệu quả về mặt tính toán và khó mở rộng quy mô. Các tác giả cũng đã chuyển sang thử nghiệm một baseline đơn giản hơn: dự đoán một "túi từ" của văn bản (tức là chỉ quan tâm từ nào xuất hiện, không quan tâm thứ tự hay ngữ pháp).



Hình 8: Trong một so sánh về hiệu năng zero-shot transfer, nhóm nghiên cứu chỉ ra rằng mô hình Transformer dựa trên generative học chậm hơn 3 lần so với baseline dự đoán bag-of-words và thậm chí là chậm hơn 12 lần so với Contrastive learning của CLIP.

- **Tập trung sai mục tiêu:** Mục tiêu cuối cùng là học được biểu diễn hình ảnh hữu ích cho nhiều tác vụ khác nhau, chứ không phải chỉ để tạo ra chú thích hoàn hảo. Đôi khi, việc dự đoán chú thích chính xác có thể khiến mô hình quá tập trung vào các chi tiết ngôn ngữ thay vì mối quan hệ ngữ nghĩa cốt lõi giữa hình ảnh và văn bản.

Nhận thấy những hạn chế này, nhóm nghiên cứu đã chuyển sang một nhiệm vụ proxy (proxy task) đơn giản hơn nhưng hiệu quả hơn nhiều: **Contrastive learning**. Ý tưởng này được lấy cảm hứng từ các nghiên cứu gần đây về học biểu diễn đối lập trong các lĩnh vực khác, cho thấy chúng có thể học được biểu diễn tốt hơn so với các mục tiêu dự đoán tương đương (Tian et al., 2019; Chen et al., 2020a).

3.3.2 Cơ chế hoạt động của Contrastive learning trong CLIP

1. Xây dựng Batch đầu vào:

- Trong mỗi bước huấn luyện, một batch gồm N cặp (hình ảnh, văn bản) thực tế được thu thập từ bộ dữ liệu WIT. Ví dụ: $(\text{Image}_1, \text{Text}_1)$, $(\text{Image}_2, \text{Text}_2)$, ..., $(\text{Image}_N, \text{Text}_N)$.
- Đây là N cặp "đúng" (positive pairs) được biết là có liên quan đến nhau.

2. Tạo các biểu diễn Embeddings: Các bộ mã hóa ảnh và văn bản sẽ nhiệm vụ trích xuất các đặc trưng riêng biệt từ đầu vào của chúng dưới dạng các vector biểu diễn cấp cao. Mục đích là chuyển đổi các đặc trưng này vào một không gian chung, được gọi là **Multi-modal Embedding Space**. Bước này được thực hiện thông qua **linear projection layers**.

Bước này hoạt động như một "cầu nối" giữa hai phương thức, cho phép chúng được so sánh và tương tác một cách có ý nghĩa.

Chức năng của các Linear projection layer

- **Ánh xạ vào không gian chung:** Mỗi bộ mã hóa (hình ảnh và văn bản) sẽ tạo ra một vector đặc trưng riêng biệt (ví dụ: I_f cho hình ảnh và T_f cho văn bản). Các vector này có thể có số chiều khác nhau và nằm trong các không gian đặc trưng riêng của từng phương thức. Để có thể so sánh trực tiếp chúng, CLIP sử dụng một lớp chiếu tuyến tính riêng biệt cho mỗi phương thức: W_i cho hình ảnh và W_t cho văn bản.
 - Cụ thể, đặc trưng hình ảnh I_f sẽ được biến đổi thành $I_e = I_f \cdot W_i$, và đặc trưng văn bản T_f sẽ được biến đổi thành $T_e = T_f \cdot W_t$.
 - Mục tiêu là I_e và T_e sẽ có cùng số chiều và nằm trong cùng một không gian nhúng đa phương thức.
- **Đầu ra tuyến tính:** Khác với một số mô hình học biểu diễn đối lập khác (ví dụ: SimCLR) thường sử dụng một "projection head" phi tuyến tính (gồm nhiều lớp MLP với hàm kích hoạt phi tuyến tính) để ánh xạ các đặc trưng từ bộ mã hóa vào không gian nhúng, CLIP đã lựa chọn một lớp chiếu **tuyến tính** đơn giản.
 - Điều đáng ngạc nhiên là nhóm nghiên cứu CLIP nhận thấy rằng việc sử dụng lớp chiếu tuyến tính không gây ra sự khác biệt đáng kể về hiệu quả huấn luyện so với lớp chiếu phi tuyến tính.
 - Điều này gợi ý rằng các biểu diễn đặc trưng mà các bộ mã hóa hình ảnh và văn bản học được (trước khi chiếu) đã có chất lượng rất cao và đủ mạnh mẽ để có thể được ánh xạ tuyến tính vào không gian chung mà vẫn giữ được thông tin quan trọng. Sự đơn giản này cũng góp phần vào hiệu quả và tính dễ mở rộng của mô hình.

L2 Normalization Sau khi các vector đặc trưng được chiếu tuyến tính vào không gian chung (I_e và T_e), một bước quan trọng tiếp theo là **chuẩn hóa L2** chúng: $I_e = \text{L2_normalize}(I_e)$ và $T_e = \text{L2_normalize}(T_e)$.

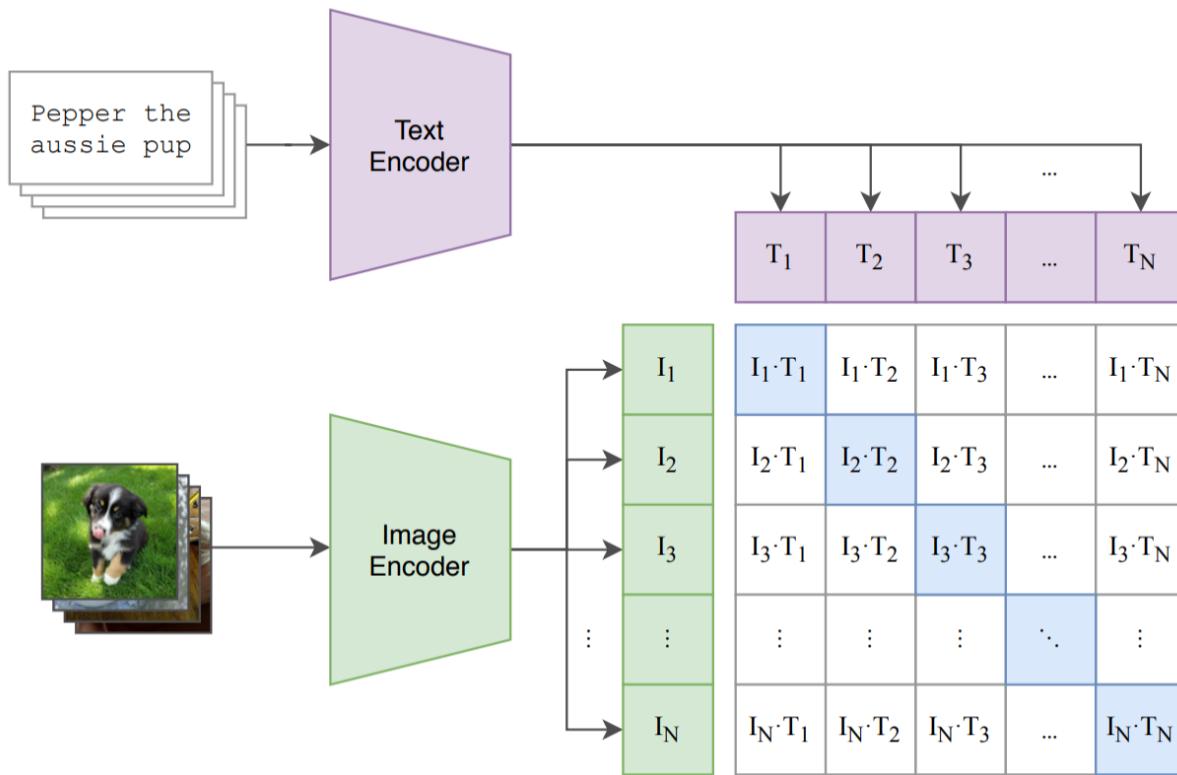
- **Đảm bảo độ dài vector là 1:** Chuẩn hóa L2 (chia mỗi vector cho độ dài Euclid của chính nó) đảm bảo rằng tất cả các vector nhúng trong không gian chung đều có độ dài bằng 1.
- **Tầm quan trọng cho Contrastive Loss:** Độ tương đồng cosine là thước đo tiêu chuẩn để đánh giá "độ gần" ngữ nghĩa trong không gian nhúng của CLIP. Việc chuẩn hóa L2 đảm bảo rằng sự tương đồng này chỉ phụ thuộc vào **góc** giữa các vector (hướng của chúng), chứ không phải vào độ dài hoặc độ lớn. Điều này rất quan trọng cho Contrastive learning, nơi mô hình cố gắng kéo các cặp liên quan lại gần nhau và đẩy các cặp không liên quan ra xa.

3. Tính toán Similarity Matrix: Với N embedding vector hình ảnh và N embedding vector văn bản từ bước trên, chúng ta có thể tạo ra một ma trận độ tương đồng S có kích thước $N \times N$.

- Mỗi phần tử $S_{i,j}$ của ma trận này được tính bằng **cosine similarity** giữa vector nhúng hình ảnh I_i và vector nhúng văn bản T_j theo công thức tổng quát sau:

$$\cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$$

Do các vector đã được chuẩn hóa L2, $S_{i,j}$ đơn giản là phép nhân vô hướng $I_i \cdot T_j^T$.



- Ma trận này chứa:

- Các phần tử trên đường chéo chính $S_{i,i}$ là độ tương đồng giữa các cặp hình ảnh-văn bản **đúng** (positive pairs).
- Các phần tử ngoài đường chéo chính $S_{i,j}$ (với $i \neq j$) là độ tương đồng giữa các cặp hình ảnh-văn bản **sai** (negative pairs), được tạo ra bằng cách ghép ngẫu nhiên hình ảnh từ một cặp đúng với văn bản từ một cặp sai khác trong cùng batch.

4. Hàm mất mát và Tối ưu hóa: CLIP sử dụng một phiên bản của mục tiêu học đối lập được gọi là **N-pair Contrastive Loss** (Sohn, 2016) hoặc **InfoNCE Loss** (Oord et al., 2018), đã được điều chỉnh cho nhiệm vụ đa phương thức hình ảnh-văn bản (Zhang et al., 2020), được gọi là **symmetric cross-entropy loss**. Mục tiêu là tối đa hóa độ tương đồng của N cặp đúng trong khi giảm thiểu độ tương đồng của $N^2 - N$ cặp sai.

- *Cụ thể:*

- Mô hình xem xét mỗi hàng của ma trận S như một tập hợp các "logits" (đầu ra thô, chưa qua xử lý) để phân loại hình ảnh I_i với N đoạn văn bản có thể có.
- Đồng thời, mô hình xem xét mỗi cột của ma trận S như một tập hợp các "logits" để phân loại văn bản T_j với N hình ảnh có thể có.
- Hàm mất mát tổng cộng là trung bình của hàm mất mát cross-entropy từ phía hình ảnh (phân loại văn bản cho ảnh) và từ phía văn bản (phân loại ảnh cho văn bản).

- **Temperature Parameter t :**

- Tham số t được sử dụng để điều khiển dải giá trị của các logits trước khi đưa vào hàm softmax (logits được nhân với $\exp(t)$).
- Một giá trị t lớn hơn sẽ làm cho phân phối xác suất sau softmax "sắc nét" hơn, tập trung nhiều hơn vào các cặp có độ tương đồng cao nhất.
- Điều đặc biệt của CLIP là tham số t này không phải là một siêu tham số cố định mà được **tối ưu hóa trực tiếp** trong quá trình huấn luyện (được tham số hóa theo log để tránh các giá trị quá lớn gây mất ổn định huấn luyện). Điều này giúp mô hình tự điều chỉnh độ "sắc nét" của sự phân biệt giữa các cặp đúng và sai.

Mã giả:

```
# image_encoder - ResNet or Vision Transformer
# text_encoder - CBOW or Text Transformer
# I[n, h, w, c] - minibatch of aligned images
# T[n, l]         - minibatch of aligned texts
# W_i[d_i, d_e] - learned proj of image to embed
# W_t[d_t, d_e] - learned proj of text to embed
# t              - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T)  #[n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss   = (loss_i + loss_t)/2
```

Hình 9: Mã giả các bước tiến hành CLIP

3.4 Đặc điểm của Multi-modal Embedding Space

Đây chính là sản phẩm cuối cùng của quá trình ánh xạ, và là nơi sự "hiểu biết" đa phương thức của CLIP thực sự phát huy tác dụng. Mô hình phải học được sự khác biệt thực sự về khái niệm, chứ không chỉ dựa vào các đặc trưng thấp.

- **Một không gian thống nhất:** Multi-modal Embedding Space là không gian vector cao chiều, nơi các biểu diễn của hình ảnh và văn bản có thể được so sánh trực tiếp.

- **Phản ánh mối quan hệ ngữ nghĩa:**

- Nếu một hình ảnh và một đoạn văn bản mô tả cùng một khái niệm hoặc có mối quan hệ ngữ nghĩa chặt chẽ (ví dụ: một bức ảnh về một chú mèo và chú thích "a cat napping on the sofa"), các embedding vector tương ứng của chúng (I_e và T_e) sẽ nằm **rất gần nhau** trong không gian này, với độ tương đồng cosine cao.
- Ngược lại, nếu một hình ảnh và một đoạn văn bản không có mối quan hệ (ví dụ: một bức ảnh về một chú chó và chú thích "a blue car"), các vector nhung của chúng sẽ nằm **xa nhau**, với độ tương đồng cosine thấp.

Điều này minh họa một cách mạnh mẽ rằng việc học để ánh xạ hai phương thức vào một không gian ngữ nghĩa thống nhất là chìa khóa để CLIP có thể tổng quát hóa sang các tác vụ mới chỉ bằng cách sử dụng ngôn ngữ.

- **Khả năng "Open-set" của không gian:** Vì không gian này được học từ dữ liệu ngôn ngữ tự nhiên đa dạng (WIT), nó không bị giới hạn bởi một tập hợp các lớp cố định. Điều này có nghĩa là các khái niệm chưa từng được "nhìn thấy" trong hình ảnh huấn luyện vẫn có thể được hiểu nếu chúng có mô tả bằng ngôn ngữ tương ứng.

4 So sánh các biến thể của CLIP

CLIP được huấn luyện với nhiều kiến trúc Image Encoder và kích thước khác nhau, dẫn đến các biến thể với hiệu suất và chi phí tính toán khác nhau.

		Food101	CIFAR10	CIFAR100	Birdsnap	SUN397	StanfordCars	FGVC Aircraft	VOC2007	DTD	Oxford Pets	Caltech101	Flowers102	MINIST	FER2013	STL10	EuroSAT	RESISC45	GTSRB	KITTI	Country211	PCam	UCF101	Kinetics700	CLEVR	HatefulMemes	Rendered SST2	ImageNet
CLIP-ResNet	RN50	81.1	75.6	41.6	32.6	59.6	55.8	19.3	82.1	41.7	85.4	82.1	65.9	66.6	42.2	94.3	41.1	54.2	35.2	42.2	16.1	57.6	63.6	43.5	20.3	59.7	56.9	59.6
	RN101	83.9	81.0	49.0	37.2	59.9	62.3	19.5	82.4	43.9	86.2	85.1	65.7	59.3	45.6	96.7	33.1	58.5	38.3	33.3	16.9	55.2	62.2	46.7	28.1	61.1	64.2	62.2
	RN50x4	86.8	79.2	48.9	41.6	62.7	67.9	24.6	83.0	49.3	88.1	86.0	68.0	75.2	51.1	96.4	35.0	59.2	35.7	26.0	20.2	57.5	65.5	49.0	17.0	58.3	66.6	65.8
	RN50x16	90.5	82.2	54.2	45.9	65.0	72.3	30.3	82.9	52.8	89.7	87.6	71.9	80.0	56.0	97.8	40.3	64.4	39.6	33.9	24.0	62.5	68.7	53.4	17.6	58.9	67.6	70.5
	RN50x64	91.8	86.8	61.3	48.9	66.9	76.0	35.6	83.8	53.4	93.4	90.6	77.3	90.8	61.0	98.3	59.4	69.7	47.9	33.2	29.6	65.0	74.1	56.8	27.5	62.1	70.7	73.6
CLIP-ViT	B/32	84.4	91.3	65.1	37.8	63.2	59.4	21.2	83.1	44.5	87.0	87.9	66.7	51.9	47.3	97.2	49.4	60.3	32.2	39.4	17.8	58.4	64.5	47.8	24.8	57.6	59.6	63.2
	B/16	89.2	91.6	68.7	39.1	65.2	65.6	27.1	83.9	46.0	88.9	89.3	70.4	56.0	52.7	98.2	54.1	65.5	43.3	44.0	23.3	48.1	69.8	52.4	23.4	61.7	59.8	68.6
	L/14	92.9	96.2	77.9	48.3	67.7	77.3	36.1	84.1	55.3	93.5	92.6	78.7	87.2	57.5	99.3	59.9	71.6	50.3	23.1	32.7	58.8	76.2	60.3	24.3	63.3	64.0	75.3
	L/14-336px	93.8	95.7	77.5	49.5	68.4	78.8	37.2	84.3	55.7	93.5	92.8	78.3	88.3	57.7	99.4	59.6	71.7	52.3	21.9	34.9	63.0	76.9	61.3	24.8	63.3	67.9	76.2

Hình 10: Bảng so sánh hiệu suất Zero-shot của các biến thể CLIP trên 17 tập dữ liệu

Dựa vào bảng so sánh trên, ta có các nhận xét:

- **Scaling Laws:** Hiệu suất của CLIP tăng lên đáng kể khi tăng kích thước mô hình (cả Image Encoder và Text Encoder) và lượng dữ liệu huấn luyện.
- **ViT vs. ResNet:** Các mô hình Vision Transformer (ViT) thường cho hiệu suất tốt hơn so với các mô hình ResNet có cùng lượng tính toán, đặc biệt là ở quy mô lớn.
- **Zero-Shot Power:** Ngay cả các biến thể nhỏ hơn của CLIP cũng cho thấy khả năng zero-shot ấn tượng trên nhiều bộ dữ liệu và tác vụ khác nhau, vượt xa các phương pháp trước đó.
- **Tính toán và bộ nhớ:** Các mô hình lớn hơn (như ViT-L/14) đòi hỏi tài nguyên tính toán và bộ nhớ GPU đáng kể cho cả huấn luyện và suy luận.

5 Ưu điểm, thách thức

Ưu điểm nổi bật

- **Khả năng Zero-Shot mạnh mẽ:** Đây là đóng góp quan trọng nhất, cho phép áp dụng vào nhiều tác vụ thị giác mà không cần huấn luyện lại hoặc fine-tuning.
- **Học từ dữ liệu web tự nhiên:** Giảm sự phụ thuộc vào các bộ dữ liệu được gán nhãn thủ công tốn kém.
- **Tính linh hoạt cao:** Dễ dàng thích ứng với các tác vụ mới chỉ bằng cách thay đổi mô tả văn bản (prompt engineering).
- **Mô hình nền tảng (Foundation Model):** Có thể được sử dụng làm cơ sở cho nhiều ứng dụng downstream phức tạp hơn (ví dụ: tạo ảnh từ văn bản, VQA).
- **Robustness:** Hiệu suất tốt trên nhiều loại dữ liệu và phân phối khác nhau so với các mô hình chỉ học trên tập dữ liệu cố định.

Thách thức, hạn chế:

- **Khó khăn với tác vụ chi tiết:** CLIP có thể gặp khó khăn với các tác vụ đòi hỏi sự hiểu biết rất chi tiết, ví dụ như đếm số lượng đối tượng nhỏ, nhận diện chữ rất nhỏ (fine-grained OCR), hoặc các mối quan hệ không gian phức tạp.
- **Chi phí tính toán:** Huấn luyện CLIP đòi hỏi tài nguyên tính toán rất lớn. Các mô hình lớn nhất cũng tốn kém khi suy luận.
- **Độ nhạy với Prompt Engineering:** Hiệu suất có thể thay đổi đáng kể tùy thuộc vào cách diễn đạt câu truy vấn văn bản.
- **Dữ liệu "nhiều" và thiên kiến (Bias):** Vì học từ dữ liệu web, CLIP có thể kế thừa các thiên kiến xã hội tiềm ẩn trong dữ liệu đó.
- **Không phải là "All-in-one":** Mặc dù mạnh mẽ, CLIP không phải lúc nào cũng là lựa chọn tốt nhất cho mọi tác vụ thị giác. Các mô hình chuyên biệt được huấn luyện có giám sát vẫn có thể vượt trội trong các lĩnh vực hẹp cụ thể.
- **Khả năng trừu tượng hóa hạn chế:** Gặp khó khăn khi khái quát hóa với các khái niệm hoàn toàn mới hoặc trừu tượng mà không có sự tương đồng trong dữ liệu huấn luyện

6 Ứng dụng của CLIP

6.1 Ứng dụng CLIP trong truy vấn Ảnh - Văn bản

Mô hình CLIP (Contrastive Language-Image Pretraining) do OpenAI phát triển là một bước tiến quan trọng trong việc xây dựng hệ thống truy vấn đa phương thức, cho phép tìm kiếm chéo giữa ảnh và văn bản. Với CLIP, người dùng có thể nhập truy vấn văn bản để tìm ảnh phù hợp hoặc nhập ảnh để truy xuất mô tả văn bản gần nhất.

Ví dụ: Truy vấn văn bản “người đàn ông đeo kính” có thể giúp hệ thống tìm ra các hình ảnh tương ứng trong tập dữ liệu. CLIP được ứng dụng rộng rãi trong các lĩnh vực như tìm kiếm hình ảnh, gợi ý nội dung, kiểm duyệt nội dung tự động, và nhiều tác vụ liên quan đến AI thị giác.

6.1.1 Bộ dữ liệu COCO 2017

Nguồn dữ liệu: <https://www.kaggle.com/datasets/awsaf49/coco-2017-dataset/data>

Bộ dữ liệu COCO (Common Objects in Context) được sử dụng phổ biến trong huấn luyện và đánh giá các mô hình học sâu trong thị giác máy tính, bao gồm cả mô hình CLIP. COCO cung cấp ảnh gắn nhãn kèm theo mô tả ngôn ngữ tự nhiên (caption), rất phù hợp cho bài toán truy vấn ảnh-văn bản.

- **Train2017:** Khoảng 118.000 ảnh gắn nhãn dùng để huấn luyện.
- **Val2017:** 5.000 ảnh dùng để đánh giá mô hình.
- **Test2017:** 41.000 ảnh (không có nhãn công khai).
- **Caption:** Mỗi ảnh đi kèm trung bình 5 mô tả ngắn do con người viết.
- **Định dạng:** Ảnh ‘.jpg’ và nhãn định dạng ‘.json’ theo chuẩn COCO.

COCO là bộ dữ liệu tiêu chuẩn được nhiều mô hình như CLIP, BLIP, ViLT, Flamingo sử dụng trong huấn luyện và đánh giá.

6.1.2 Thực hiện Truy vấn Ảnh - Văn bản với CLIP

Do việc huấn luyện mô hình từ đầu trên bộ dữ liệu lớn tốn kém tài nguyên (GPU, thời gian), ta sử dụng mô hình CLIP ViT-B/32 đã được huấn luyện sẵn bởi OpenAI. Các bước thực hiện như sau:

- **Bước 1:** Nạp mô hình CLIP ViT-B/32 từ thư viện `clip` của OpenAI.
- **Bước 2:** Tiền xử lý ảnh và văn bản đầu vào (resize, chuẩn hóa, token hóa).
- **Bước 3:** Trích xuất vector đặc trưng (embedding) từ cả ảnh và văn bản.
- **Bước 4:** Tính độ tương đồng cosine giữa các vector để xác định cặp gần nhất.
- **Bước 5:** Truy xuất ảnh hoặc văn bản tương ứng có độ tương đồng cao nhất.

6.1.3 Kết quả trên tập COCO Validation

Mô hình CLIP ViT-B/32 được đánh giá trên tập val2017 của COCO với các chỉ số truy xuất gồm:

- Recall@1: 49.92%
- Recall@5: 74.94%
- Recall@10: 83.24%

Nhận xét:

- CLIP cho kết quả tốt ở Recall@10 (trên 83%), cho thấy khả năng bao phủ cao.
- Recall@1 còn hạn chế (~ 50%) – mô hình có thể nhầm lẫn khi các caption có ngữ nghĩa gần nhau.
- Phù hợp với các ứng dụng cần truy xuất nhiều kết quả để chọn lọc (top-5, top-10).

6.1.4 Ví dụ minh họa kết quả truy vấn

Hình dưới đây minh họa một truy vấn văn bản và kết quả truy xuất ảnh tương ứng.



Hình 11: *Ảnh minh họa kết quả truy vấn từ văn bản*

Truy vấn văn bản:

- "A boy in birthday hat holding a tennis racket"
- "A boy swinging a tennis racket at a ball on a court."

Caption gốc của ảnh:

- "A boy in birthday hat holding a tennis racket"
- "A young boy in a birthday hat holds a tennis racquet"

6.2 Ứng dụng của CLIP để phân biệt khuôn mặt, từ đó phát triển tác vụ nhận diện khuôn mặt người.

CLIP (Contrastive Language–Image Pretraining) là mô hình được huấn luyện trên hàng trăm triệu cặp dữ liệu văn bản và hình ảnh để học mối liên hệ giữa chúng. Mặc dù không được thiết kế chuyên biệt cho nhận diện khuôn mặt, CLIP có khả năng biểu diễn hình ảnh mạnh mẽ, cho phép ứng dụng hiệu quả trong các bài toán phân loại và nhận diện khuôn mặt.

6.2.1 Mục tiêu

- Mã hóa ảnh khuôn mặt thành vector đặc trưng (embedding).
- Dự đoán và phân biệt khuôn mặt bằng cách so sánh độ tương đồng giữa các embedding.

6.2.2 Dữ liệu

Nguồn dữ liệu: <https://www.kaggle.com/datasets/stoicstatic/face-recognition-dataset>

Bộ dữ liệu được xây dựng dựa trên tập **Labeled Faces in the Wild** với ảnh JPEG của những người nổi tiếng, mỗi ảnh có kích thước 250x250. Mỗi thư mục trong dataset đại diện cho một người nổi tiếng, chứa từ 2 đến 50 ảnh khuôn mặt.



Hình 12: Các khuôn mặt trong tập dữ liệu nhận diện.

Xử lý dữ liệu

1. Áp dụng kỹ thuật tăng cường dữ liệu (augmentation) để cân bằng số lượng ảnh giữa các lớp.
2. Chia dữ liệu thành tập huấn luyện (Train) và kiểm tra (Test).
3. Tạo Dataloader cho việc huấn luyện theo cặp ảnh.

6.2.3 Xây dựng mô hình

Chúng tôi sử dụng kiến trúc mô hình ViT-B/32 của CLIP để rút trích đặc trưng khuôn mặt. Chúng tôi đã mô phỏng lại các lớp trong ViT-B/32 để huấn luyện nhận diện khuôn mặt bằng Contrastive Learning.

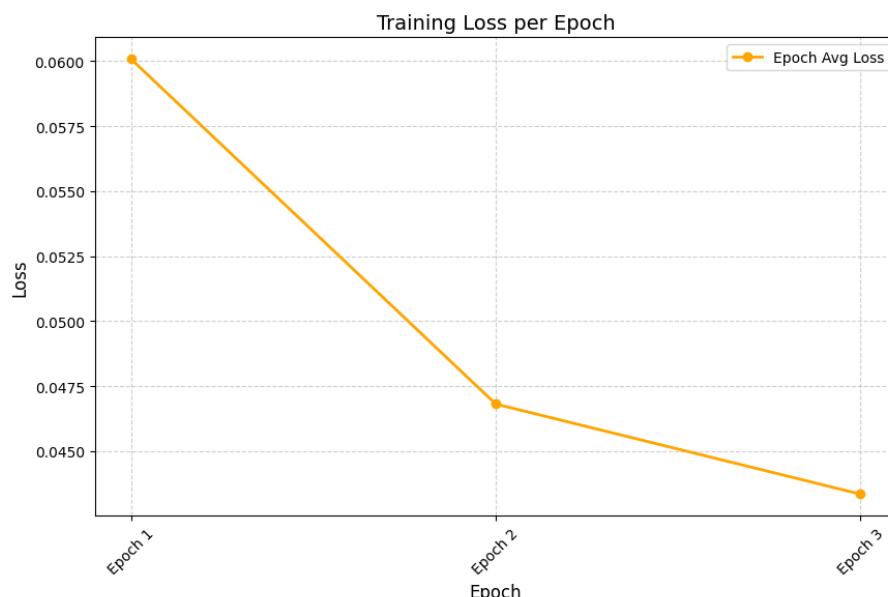
Kiến trúc mô hình ViT-B/32 (Sử dụng lại lớp Vision Transformer)

- **Patch Embedding:** Ảnh RGB được chia thành các patch kích thước 32×32 và chuyển thành vector.
- **LayerNorm:** Chuẩn hóa đầu vào cho Transformer.
- **Transformer Encoder:** Gồm 12 khối ResidualAttentionBlock:
 - Mỗi block có multi-head attention với đầu vào/ra 768 chiều.
 - Chuẩn hóa (\ln_1 , \ln_2).
 - MLP: Linear($768 \rightarrow 3072$) \rightarrow QuickGELU \rightarrow Linear($3072 \rightarrow 768$).
- **Output LayerNorm:** Chuẩn hóa lần cuối để lấy embedding có đầu ra 768.

6.2.4 Huấn luyện

- Hàm mất mát: **Contrastive Loss**.
- Thuật toán tối ưu: **Adam**, với tốc độ học ban đầu là $1e^{-3}$.
- Số epoch: **3**. (Do mô hình khá phức tạp và huấn luyện rất tốn GPU nên chúng tôi chỉ mô phỏng huấn luyện qua 3 epoch để thấy được sự khác biệt.)
- Dữ liệu được cung cấp dưới dạng các cặp ảnh:
 - Cặp dương tính: Hai ảnh cùng người.
 - Cặp âm tính: Hai ảnh khác người.

Kết quả huấn luyện cho thấy loss giảm ổn định sau mỗi epoch, cho thấy mô hình học được biểu diễn đặc trưng khuôn mặt hiệu quả.



Hình 13: Biểu đồ loss theo từng epoch trong quá trình huấn luyện.

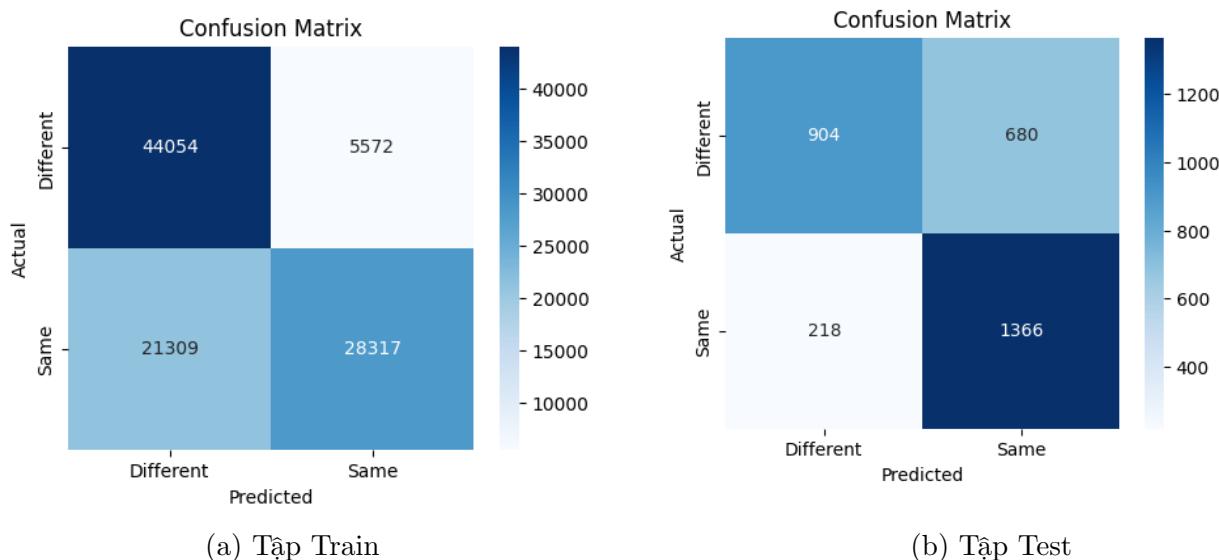
Nhận xét Mô hình hiện tại mới chỉ được huấn luyện trong **3 epoch** với dữ liệu **giới hạn**, dẫn đến chất lượng embedding chưa thực sự ổn định. Cụ thể:

- Các vector đặc trưng (embedding) của ảnh **cùng một người** chưa hoàn toàn hội tụ gần nhau.
- Các ảnh **khác người** đôi khi chưa được tách biệt rõ ràng trong không gian vector.

Các chỉ số đánh giá mô hình:

Chỉ số	Tập Train	Tập Test
Accuracy	72.92%	71.65%
Precision	83.56%	66.76%
Recall	57.06%	86.24%
F1-score	67.81%	75.26%

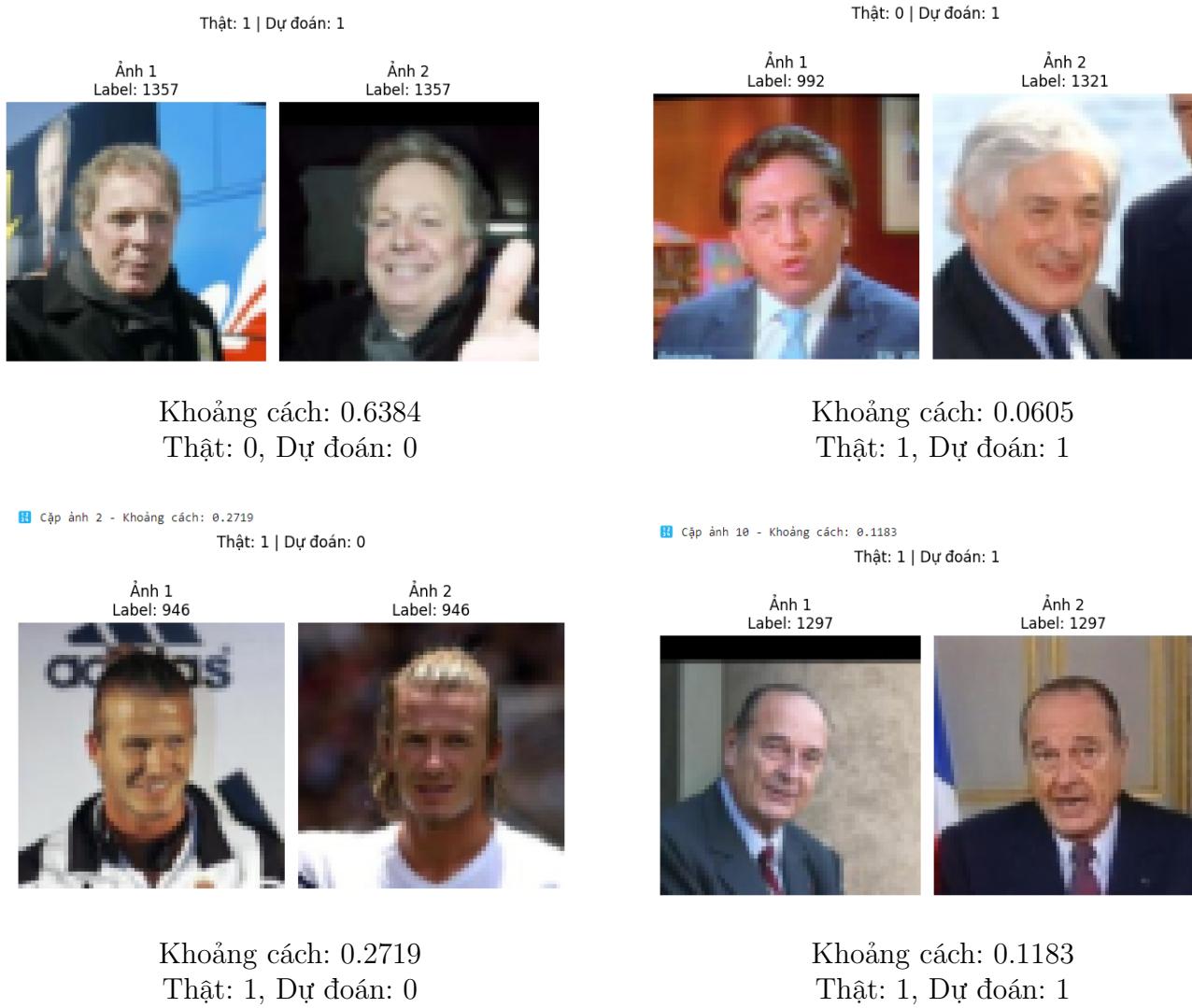
Confusion Matrix:



Hình 14: Ma trận nhầm lẫn trên tập train và test

Mô hình đạt Accuracy 72.92% (train) và 71.65% (test), cho thấy độ ổn định tương đối giữa hai tập dữ liệu. Precision giảm từ 83.56% xuống 66.76%, phản ánh sự xuất hiện của nhiều dự đoán dương tính sai trên tập test. Ngược lại, Recall tăng mạnh từ 57.06% lên 86.24%, cho thấy mô hình nhận diện hiệu quả các cặp ảnh cùng người. F1-score cũng được cải thiện, từ 67.81% lên 75.26%, thể hiện sự cân bằng tốt hơn giữa precision và recall.

Trên tập kiểm thử, mô hình bước đầu phân biệt được ảnh khuôn mặt cùng và khác người, nhưng vẫn cần tinh chỉnh thêm để tối ưu hóa hiệu suất.

Ví dụ minh họa trên tập test:

Hình 15: Một số dự đoán minh họa trên tập kiểm thử

Kết luận:

Việc huấn luyện ngắn hạn đã giúp mô hình bước đầu học được biểu diễn khái quát về khuôn mặt. Tuy nhiên, để cải thiện hiệu năng, cần:

- Tăng số lượng epoch huấn luyện.
- Bổ sung thêm dữ liệu đa dạng hơn.

6.2.5 So sánh và đánh giá trong tác vụ nhận diện khuôn mặt

Mặc dù CLIP không được thiết kế chuyên biệt cho nhiệm vụ nhận diện khuôn mặt, nhưng nó có khả năng trích xuất đặc trưng từ ảnh và so sánh với các mẫu đã biết để tìm ra ảnh có đặc trưng gần nhất. Nếu ảnh gần nhất tương ứng với một khuôn mặt đã được gán nhãn, mô hình sẽ dự đoán đó là khuôn mặt tương ứng.

Dể đánh giá hiệu quả của CLIP trong bài toán nhận diện khuôn mặt, chúng tôi tiến hành so sánh với bốn mô hình CNN tiêu biểu khác.

Mô hình	Accuracy	Precision	Recall	F1
CLIP (ViT-B/32)	0.73	0.6982	0.7311	0.7022
ResNet18	0.2151	0.1743	0.2151	0.1672
ResNet50	0.2169	0.1493	0.2169	0.1483
EfficientNet-B0	0.2218	0.1658	0.2218	0.1621
MobileNetV2	0.2602	0.1916	0.2602	0.1923

Nhận xét: Trong thiết lập thực nghiệm này, mô hình **CLIP (ViT-B/32 pretrained)** được sử dụng như một bộ trích xuất đặc trưng (feature extractor) cho ảnh đầu vào. Sau khi trích xuất embedding, phương pháp **FAISS k-NN** được áp dụng để tìm embedding gần nhất trong tập huấn luyện, từ đó suy ra nhãn dự đoán. Kết quả cho thấy chiến lược này mang lại độ chính xác và độ khai quát hóa vượt trội so với các mô hình CNN truyền thống huấn luyện lại như ResNet, MobileNet hay EfficientNet.

Điểm mạnh nổi bật của cách tiếp cận này nằm ở việc tận dụng embedding mạnh mẽ của CLIP, vốn đã được huấn luyện trên tập dữ liệu quy mô lớn và đa dạng về ngữ nghĩa. Việc sử dụng **FAISS k-NN** giúp thay thế hoàn toàn quá trình huấn luyện mô hình phân loại, đồng thời giảm rủi ro overfitting trong các tập dữ liệu có số lượng lớp lớn hoặc phân bố không đều.

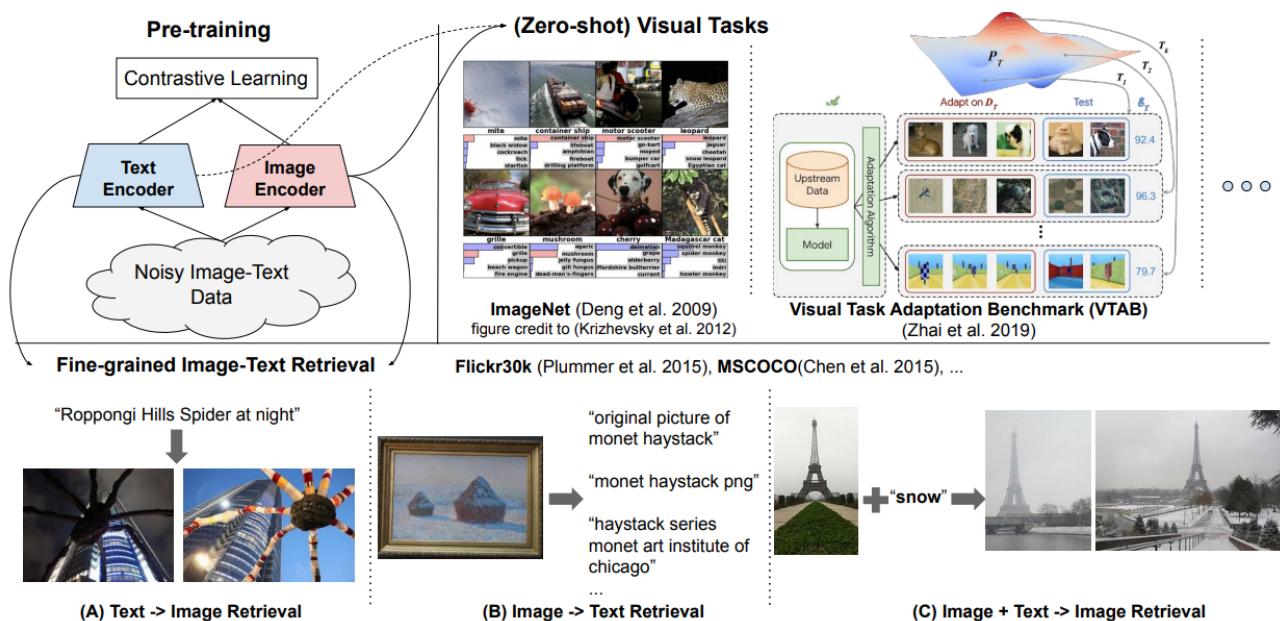
Trong khi các mô hình CNN chỉ huấn luyện phần classifier cuối thường gặp khó khăn với số lớp lớn hoặc dữ liệu ít, mô hình CLIP kết hợp FAISS hoạt động như một hệ thống truy hồi embedding hiệu quả, cho phép nhận diện chính xác dựa trên khoảng cách đặc trưng mà không cần gradient descent.

Kết luận: Việc sử dụng **CLIP + FAISS** là một chiến lược rất thực tiễn và hiệu quả trong các ứng dụng nhận diện hình ảnh, đặc biệt trong các bài toán nhiều lớp, ít dữ liệu hoặc không muốn huấn luyện thêm. Chiến lược này tận dụng tối đa sức mạnh của mô hình pretrained mà vẫn đạt hiệu suất rất tốt trong phân loại.

7 Các mô hình tương tự CLIP

7.1 ALIGN

Trong phương pháp **ALIGN** được giới thiệu trong bài báo **Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision** [6], các biểu diễn (representations) hình ảnh và ngôn ngữ được huấn luyện đồng thời từ dữ liệu nhiễu (noisy) là các cặp ảnh và văn bản thay thế (alt-text). Các bộ mã hóa hình ảnh (image encoder) và văn bản (text encoder) được học thông qua hàm mất mát tương phản (contrastive loss), được định dạng dưới dạng softmax chuẩn hóa. Hàm mất mát này có tác dụng đẩy các embedding của cặp ảnh-văn bản khớp nhau lại gần nhau hơn, đồng thời đẩy chúng ra xa các embedding của những cặp ảnh-văn bản không khớp.



Hình 16: Tóm tắt phương pháp ALIGN.

Mục tiêu chính: Mục tiêu của ALIGN là học các biểu diễn (representations) mạnh mẽ cho cả thị giác (vision) và ngôn ngữ-thị giác (vision-language) bằng cách mở rộng quy mô dữ liệu huấn luyện lên mức cực lớn, nhưng với một phương pháp thu thập dữ liệu đơn giản, ít tốn kém. Giả thuyết cốt lõi của họ là **quy mô của dữ liệu có thể bù đắp cho sự nhiễu** của nó.

Kỹ thuật chính:

- **Dữ liệu (Data):** Đây là điểm đột phá và khác biệt nhất của ALIGN.
 - **Nguồn:** Họ sử dụng một tập dữ liệu khổng lồ gồm hơn 1.8 tỷ cặp ảnh và alt-text (văn bản thay thế cho ảnh) được thu thập từ web.
 - **Đặc điểm:** Dữ liệu này rất "nhiễu" (noisy). Các alt-text thường không phải là mô tả hoàn hảo, có thể chứa thông tin không liên quan, tên file, hoặc chỉ là một vài từ khóa.

- **Quy trình lọc:** Thay vì áp dụng các bước lọc và xử lý phức tạp, tốn kém như các bộ dữ liệu được giám sát kỹ lưỡng (ví dụ: Conceptual Captions), ALIGN chỉ áp dụng các bộ lọc rất đơn giản dựa trên tần suất (frequency-based filtering). Ví dụ: loại bỏ các ảnh/văn bản khiêu dâm, ảnh có kích thước quá nhỏ, hoặc các alt-text quá phổ biến (như "ảnh"), quá ngắn hoặc quá dài.

- **Kiến trúc mô hình (Model architecture):**

- ALIGN sử dụng một kiến trúc **bộ mã hóa kép (dual-encoder)**.
- **Image Encoder:** Một mạng CNN, cụ thể là **EfficientNet**.
- **Text Encoder:** Một mạng Transformer, cụ thể là **BERT**.
- Kiến trúc này lấy một ảnh và một đoạn văn bản, mã hóa chúng một cách độc lập để tạo ra hai vector embedding.

- **Hàm mục tiêu (Loss function):**

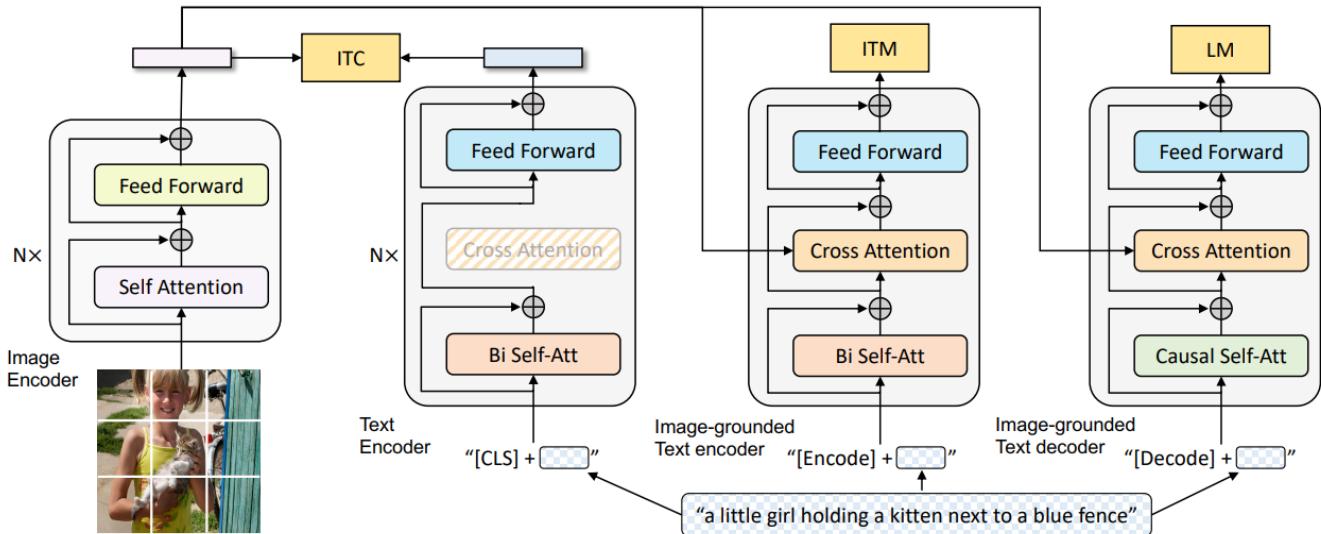
- Cả Image encoder và Text encoder được huấn luyện bằng một **hàm mất mát tương phản (contrastive loss)**, cụ thể là normalized softmax loss (còn gọi là InfoNCE).
- **Cách hoạt động:** Trong một batch dữ liệu, với mỗi cặp (ảnh, văn bản) đúng, mô hình sẽ cố gắng kéo vector embedding của chúng lại gần nhau trong không gian biểu diễn chung. Đồng thời, nó sẽ đẩy vector embedding của cặp đó ra xa khỏi tất cả các vector embedding của các ảnh và văn bản khác trong batch (được coi là các cặp "âm" - negative pairs). Quá trình này giúp "căn chỉnh" (align) không gian biểu diễn của ảnh và văn bản.

7.2 BLIP

BLIP được giới thiệu trong bài báo **BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation** [7]. Các mô hình tiền huấn luyện thị giác-ngôn ngữ (Vision-Language Pre-training, viết tắt là VLP) đã nâng cao hiệu suất cho nhiều tác vụ thị giác-ngôn ngữ. Tuy nhiên, hầu hết VLP chỉ vượt trội ở một trong hai tác vụ: **hiểu** (understanding) hoặc **sinh** (generation). BLIP là mô hình có thể làm tốt ở cả hai nhiệm vụ này. BLIP còn có phiên bản cải tiến là BLIP-2 [8].

Mục tiêu chính:

- **Về mô hình:** Các mô hình hiện có thường chỉ xuất sắc ở một trong hai loại tác vụ: hoặc là các tác vụ **hiểu** (understanding) như truy xuất ảnh, phân loại (dựa trên kiến trúc encoder), hoặc là các tác vụ **sinh** (generation) như tạo chú thích ảnh (dựa trên kiến trúc encoder-decoder). BLIP muốn tạo ra một mô hình **thống nhất (unified)** có thể làm tốt cả hai.
- **Về dữ liệu:** Các mô hình như ALIGN và CLIP đạt được thành công bằng cách tăng quy mô dữ liệu với các cặp (ảnh, alt-text) nhiều từ web. BLIP cho rằng đây chưa phải là cách tối ưu và đề xuất một phương pháp **chủ động cải thiện chất lượng dữ liệu** thay vì chỉ chấp nhận sự nhiễu loạn của nó.



Hình 17: Tổng quan mô hình BLIP

Kỹ thuật chính: BLIP giới thiệu hai đóng góp chính - về dữ liệu và về kiến trúc mô hình.

- **CapFilt (Captioning and Filtering) - Tự cải thiện dữ liệu:** Đây là ý tưởng đột phá nhất của BLIP, một quy trình "bootstrapping" để làm sạch và làm giàu dữ liệu.
 1. **Huấn luyện mô hình ban đầu:** Đầu tiên, họ huấn luyện một mô hình BLIP cơ bản trên 14 triệu cặp ảnh-văn bản nhiễu từ web.
 2. **Tạo hai mô-đun chuyên dụng:** Từ mô hình đã huấn luyện, họ tinh chỉnh (fine-tune) để tạo ra hai mô-đun:
 - **Captioner** (Bộ tạo chú thích): Một bộ giải mã (decoder) có khả năng tạo ra các chú thích mới (synthetic captions) cho các ảnh trên web.
 - **Filter** (Bộ lọc): Một bộ mã hóa (encoder) học cách xác định xem một cặp (ảnh, văn bản) có khớp nhau hay không.
 3. **Cải thiện dữ liệu:**
 - **Sinh chú thích mới:** Dùng Captioner để tạo một chú thích mới cho mỗi ảnh từ web.
 - **Lọc nhiễu:** Dùng Filter để loại bỏ những cặp (ảnh, văn bản) không khớp. Quá trình lọc này được áp dụng cho cả chú thích gốc từ web và chú thích mới được tạo ra.
 4. **Huấn luyện mô hình cuối cùng:** Họ kết hợp tập dữ liệu đã được làm sạch và làm giàu này với các bộ dữ liệu chất lượng cao có sẵn (như COCO) để huấn luyện một mô hình BLIP mới từ đầu.

Kết quả là họ có một tập dữ liệu huấn luyện lớn, vừa sạch hơn, vừa đa dạng hơn về mặt ngữ nghĩa.

- **MED (Multimodal Mixture of Encoder-Decoder) - Kiến trúc thống nhất:** Để phục vụ cả tác vụ hiểu và sinh, BLIP đề xuất một kiến trúc đa năng có thể hoạt động ở ba chế độ:

1. **Encoder đơn phương thức (Unimodal Encoder)**: Hoạt động như ALIGN/CLIP. Mã hóa ảnh và văn bản một cách riêng biệt, sau đó dùng contrastive loss (ITC) để căn chỉnh chúng. Dùng cho tác vụ truy xuất.
2. **Bộ mã hóa văn bản dựa trên ảnh (Image-grounded Text Encoder)**: Thêm các lớp chú ý chéo (cross-attention) để kết hợp thông tin hình ảnh vào biểu diễn văn bản. Dùng hàm loss image-text matching (ITM) để học sự tương hợp ở mức độ chi tiết hơn (xác định cặp ảnh-văn bản là "positive" hay "negative"). Dùng cho tác vụ hiểu sâu hơn.
3. **Bộ giải mã văn bản dựa trên ảnh (Image-grounded Text Decoder)**: Sử dụng các lớp tự chú ý nhân quả (causal self-attention) để sinh văn bản dựa trên ảnh đầu vào. Dùng hàm loss Language Modeling (LM). Dùng cho tác vụ sinh văn bản như tạo chủ thích, trả lời câu hỏi.

Kiến trúc này rất thông minh ở chỗ nó chia sẻ phần lớn các tham số giữa ba chế độ, giúp việc huấn luyện hiệu quả.

7.3 So sánh CLIP, ALIGN và BLIP

Tính năng	ALIGN (Google)	CLIP (OpenAI)	BLIP (Salesforce)
Triết lý	Quy mô dữ liệu bù đắp sự nhiễu	Dặt nền móng cho các tác vụ zero-shot.	Linh hoạt, hiểu sâu ảnh-văn bản.
Kiến trúc	EfficientNet - BERT.	Resnet/ViT-Transformer	MED
Dữ liệu	1.8 tỷ cặp ảnh-văn bản thô từ web.	400 triệu cặp ảnh-văn bản (WIT).	Tự tạo và lọc dữ liệu (CapFilt).
Thế mạnh	- NLP đa dạng. - Truy vấn rộng	- Phân loại Zero-shot. - Retrieval image.	- Image captioning. - VQA.
Ứng dụng	Hệ thống tìm kiếm hình ảnh bằng ngôn ngữ tự nhiên, phức tạp.	Tác vụ phân loại nhanh mà không cần fine-tuning.	Ứng dụng đòi hỏi sự hiểu biết chi tiết hình ảnh-văn bản.

8 Kết luận, đánh giá

8.1 Kết luận

CLIP (Contrastive Language-Image Pre-Training):

- **Khả năng:** Hiểu nội dung hình ảnh và văn bản theo cách liên kết chúng lại với nhau.
- **Kiến trúc:** Dual-Encoder (ViT/ResNet, Transformer), huấn luyện bằng Contrastive Learning, dựa trên dữ liệu WIT.
- **Ứng dụng:** Phân loại (Zero-shot); truy vấn ảnh (bằng văn bản); làm nền tảng cho caption ranking, VQA.
- **Hạn chế:** Không hỗ trợ sinh văn bản.

8.2 Đánh giá

Nhóm 6 đã hoàn thành tốt các nhiệm vụ đề ra. Bài báo cáo cùng video thuyết trình, mã nguồn đã giải quyết được những vấn đề sau:

- Các nền tảng và nguyên lý cốt lõi của CLIP.
- Ưu, nhược điểm của CLIP và so sánh với các mô hình tương tự.
- Ứng dụng của CLIP trong các tác vụ khác nhau.
- Thuyết trình và demo về CLIP.

Tài liệu

- [1] Ha Duong, Hoang Trung, Duc Tai, and Minh Trung. Math for AI - MTH00056, AI23@HCMUS. <https://github.com/ductai05/Math-For-AI/>, 2025.
- [2] Ha Duong, Hoang Trung, Duc Tai, and Minh Trung. [MML - 23TNT1 - Nhóm 6] Dồ án cuối kì: CLIP. <https://www.youtube.com/watch?v=1G227RKnv-k>, 2025.
- [3] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.
- [4] OpenAI. CLIP: Contrastive Language-Image Pre-training (Software Repository). <https://github.com/openai/CLIP>, 2021. Accessed: 2025-05-20.
- [5] OpenAI. CLIP: Connecting text and images. <https://openai.com/index/clip/>, 2021. Accessed: 2025-05-20.
- [6] Chao Jia, Yafei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision, 2021.
- [7] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022.
- [8] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023.

A Phụ lục



Hình 18: Hình ảnh một con mèo nằm giữa một cuốn sách và một cái laptop, trên một cái bàn nhỏ.

Văn bản cho trước	CLIP	ALIGN
'a photo of a cat'	0.0228	0.0018
'a photo of a dog'	0.0020	0.0000
'cat lying between laptop computer and book on small desk'	0.9752	0.9982

Bảng 1: Kết quả so sánh độ tương đồng ảnh-văn bản bởi CLIP và ALIGN

Tác vụ	BLIP
Tạo chú thích ảnh (Image Captioning)	a cat is sleeping on a desk with a laptop
How many cats are there? (VQA)	1
What is the cat's color? (VQA)	white

Bảng 2: Kết quả của BLIP cho 2 tác vụ: VQA và Image Captioning