

Báo cáo Final Lab, Toán cho AI: Contrastive Language-Image Pre-Training (CLIP)



Nhóm 6 - Math4AI, AI23@HCMUS

Sinh viên thực hiện:

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

—
AI23@HCMUS, VNUHCM

Giáo viên hướng dẫn:

TS. Cấn Trần Thành Trung
ThS. Nguyễn Ngọc Toàn
ThS. Trần Hà Sơn

—
FIT@HCMUS, VNUHCM

Ngày 23 tháng 6 năm 2025



fit@hcmus 

HCMUS **fit@hcmus**

1. Giới thiệu

CLIP là một bước đột phá trong AI đa phương thức (multi-modal AI), cho phép liên kết hình ảnh và ngôn ngữ.

CLIP (Contrastive Language–Image Pretraining) là mô hình thị giác–ngôn ngữ do OpenAI phát triển, nhằm học biểu diễn chung cho ảnh và văn bản. Nhờ phương pháp huấn luyện contrastive learning trên dữ liệu lớn, CLIP có khả năng thực hiện nhiều tác vụ mà không cần huấn luyện lại — gọi là zero-shot learning.

2. Nền tảng xây dựng mô hình CLIP

- Dữ liệu: WIT
- Kiến trúc mô hình: Dual-Encoder

2.1.1 Đặt vấn đề: Hạn chế của bộ dữ liệu truyền thống

Bối cảnh

Các mô hình thị giác máy tính hàng đầu thường được huấn luyện trên các bộ dữ liệu được gán nhãn thủ công (ví dụ: ImageNet, MS-COCO).

Hạn chế đáng kể:

- Quy mô và chi phí hạn chế
- Phạm vi khái niệm cố định
- Thiếu ngữ cảnh và sắc thái

2.1.2 Động lực từ NLP và câu hỏi lớn

Thành công của NLP:

- Mô hình như GPT-3, BERT đã cách mạng hóa NLP nhờ huấn luyện trên khối lượng văn bản khổng lồ từ internet.
- Học từ dữ liệu "thô", không gán nhãn, bằng giám sát ngôn ngữ tự nhiên.
- Khả năng chuyển giao zero-shot vượt trội.

Liệu chúng ta có thể áp dụng triết lý tương tự để đạt được bước đột phá trong thị giác máy tính?

⇒ Bộ dữ liệu WIT sinh ra để trả lời câu hỏi này.

2.1.3 Bộ dữ liệu WIT (WebImageText)

WIT được xây dựng để trở thành nền tảng cho việc học biểu diễn hình ảnh từ ngôn ngữ tự nhiên.

Quy mô

- Bao gồm 400 triệu cặp (hình ảnh, văn bản) được thu thập từ internet.
- Lớn hơn 300 lần so với ImageNet và 4000 lần so với MS-COCO.

Lợi ích

- *Tiếp xúc đa dạng hơn*: Bao phủ một lượng lớn các khái niệm, đối tượng, hành động, phong cách nghệ thuật, và ngữ cảnh khác nhau.
- *Giảm thiểu overfitting*: Lượng dữ liệu đồ sộ buộc mô hình phải học các khái niệm tổng quát hơn.
- *Học hỏi sâu sắc hơn*: Khai thác tối đa tiềm năng của kiến trúc mạng sâu.

2.1.3 Bộ dữ liệu WIT (WebImageText)

Bản chất giám sát: Đây là điểm đặc biệt và cốt lõi nhất của WIT, sử dụng văn bản mô tả tự nhiên đi kèm với hình ảnh (**Natural Language Supervision**).

Lợi ích

- Tính đa dạng và khả năng khái quát hóa của khái niệm → Chìa khóa cho *Zero-Shot Transfer*.
- Tiết kiệm chi phí
- Chấp nhận "nhiều" tự nhiên vì dữ liệu từ internet không được gán nhãn thủ công một cách hoàn hảo.

2.1.3 Bộ dữ liệu WIT (WebImageText)

Chiến lược xây dựng

- Tìm kiếm cặp hình ảnh-văn bản dựa trên 500.000 truy vấn chọn lọc (từ Wikipedia, WordNet, v.v.) để đảm bảo tính bao quát.
- Đảm bảo rằng bộ dữ liệu bao phủ một **phạm vi rất rộng** các khái niệm, đối tượng, và tình huống khác nhau.

2.2 Kiến trúc mô hình

CLIP bao gồm hai bộ mã hóa chính (dual-encoder) hoạt động độc lập để xử lý hai loại dữ liệu:

- **Image Encoder:** Chuyển đổi hình ảnh thành vector.
- **Text Encoder:** Chuyển đổi văn bản thành vector.

Mục tiêu: Dựa cả hai vector này vào cùng một không gian nhúng đa phương thức.

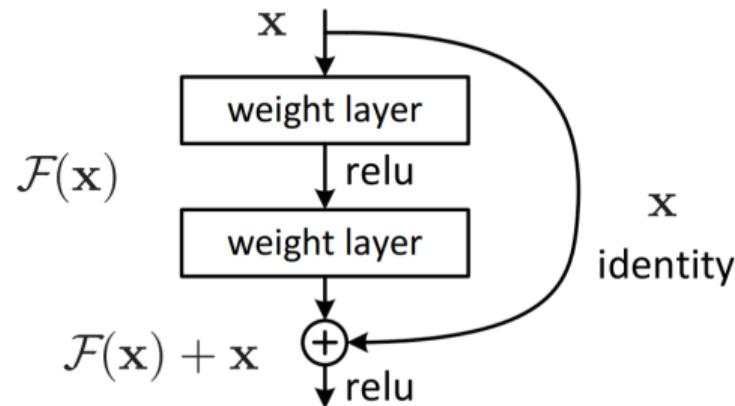
2.3 Image Encoder

CLIP thử nghiệm 2 dòng kiến trúc chính:

- ResNet (Residual Networks)
- Vision Transformer (ViT)

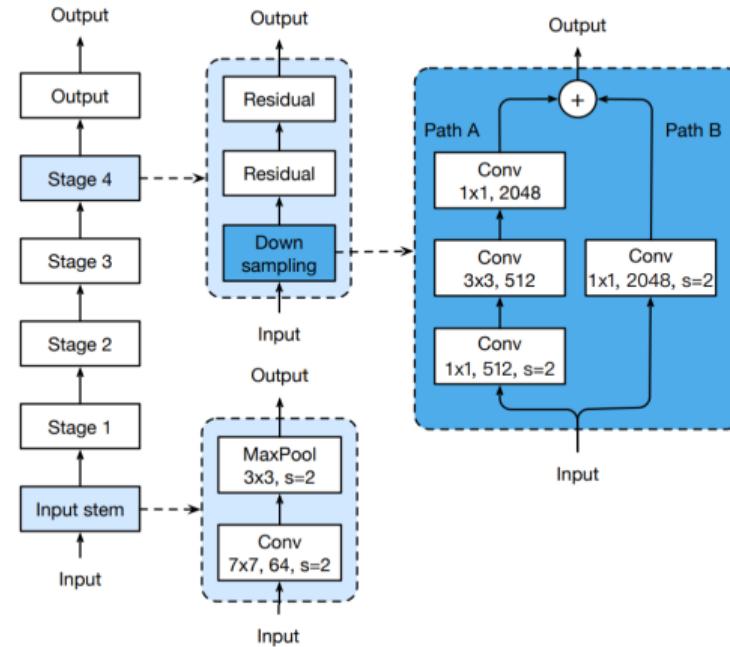
2.3.1 ResNet

- Kiến trúc CNN đột phá, nổi tiếng với kỹ thuật skip connections.

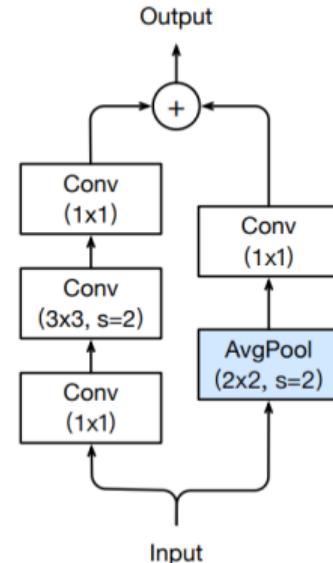
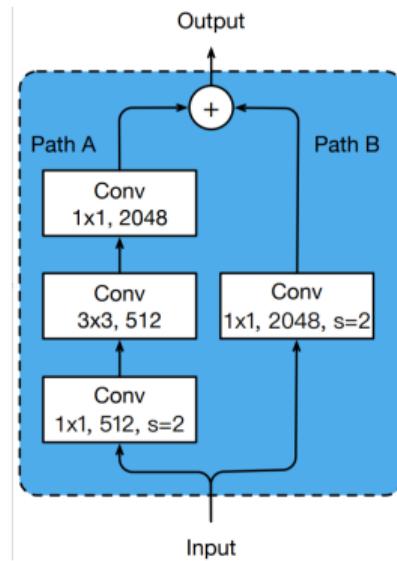


- CLIP sử dụng ResNet-50 và ResNet-101 làm cơ sở.
- Áp dụng một số cải tiến quan trọng.

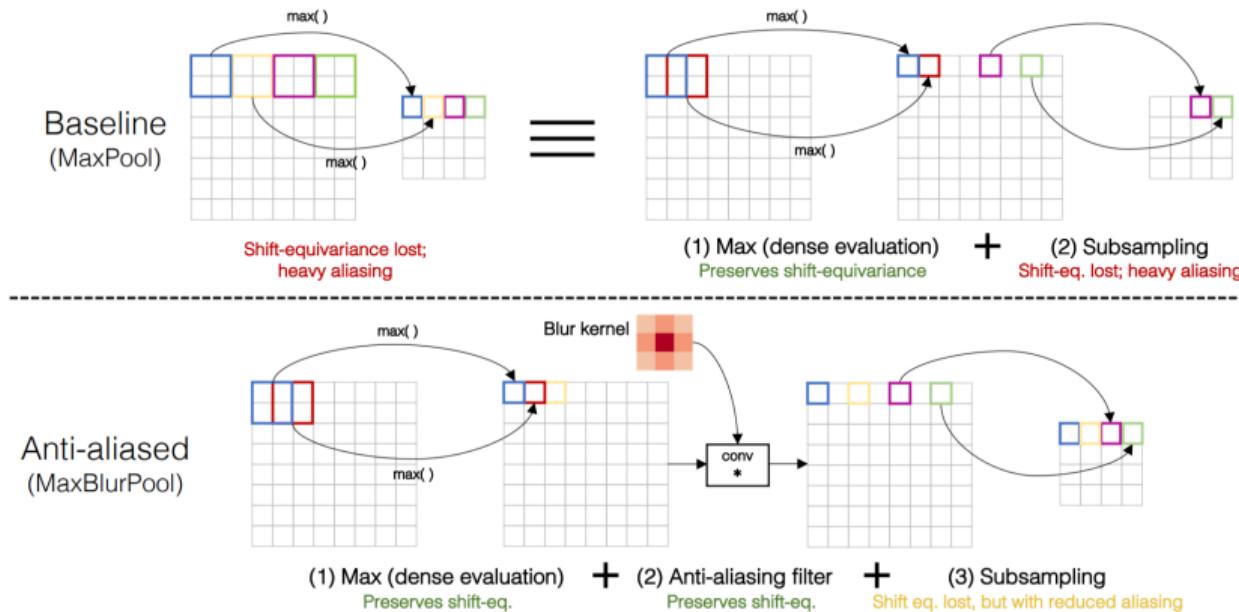
ResNet-50



ResNet-D



Anti-aliasing



Making Convolutional Networks Shift-Invariant Again, Richard Zhang

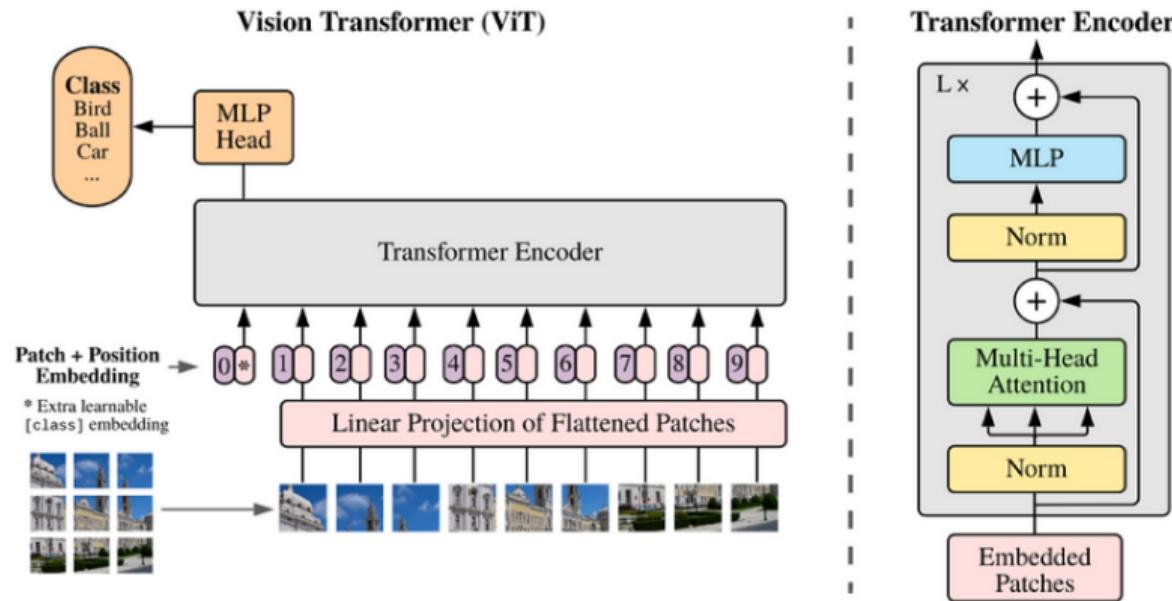
Attention Pooling

Thay thế **Global Average Pooling** bằng **Attention Pooling**: Cho phép mô hình tập trung vào các vùng quan trọng nhất của hình ảnh khi tạo biểu diễn cuối cùng, thay vì chỉ lấy trung bình đơn giản.

2.3.2 Vision Transformer

- Kiến trúc mới hơn, dựa trên Transformer của NLP, hiệu quả vượt trội trên dữ liệu lớn.
- **Các sửa đổi nhỏ:** Thêm lớp Layer Normalization trước các khối Transformer, áp dụng lược đồ khởi tạo khác một chút.
- **Kỹ thuật huấn luyện:** Huấn luyện phiên bản ViT-L/14 lớn nhất thêm một epoch ở độ phân giải cao hơn (336x336 pixel) để tăng cường hiệu suất

Vision Transformer



2.4 Text Encoder

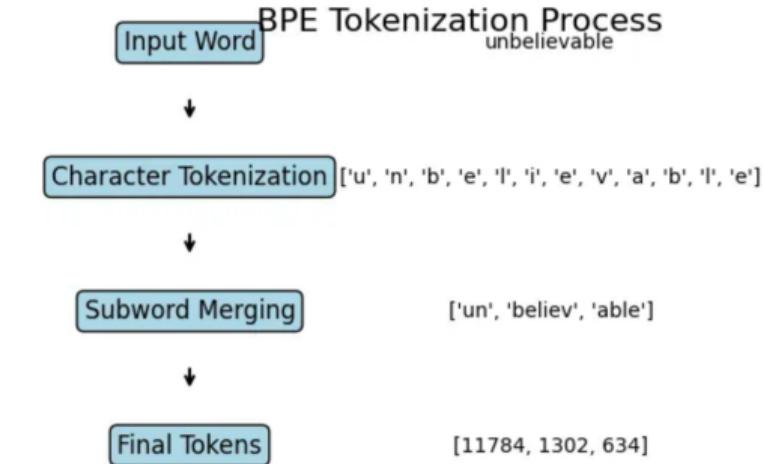
Sử dụng kiến trúc Transformer làm nền tảng cho bộ mã hóa văn bản, với một số điều chỉnh cụ thể.

2.4.1 Kiến trúc nền tảng

- Transformer chỉ có decoder, dựa trên Transformer gốc và cải tiến từ GPT-2.
- Cấu hình điển hình:** 12 lớp, chiều rộng 512, 8 attention heads.

2.4.2 Text Tokenization

- Sử dụng thuật toán **Byte Pair Encoding (BPE)** để mã hóa từ.
- Sử dụng từ điển với 49152 từ vựng.



2.4.3 Masked Self-Attention

Một cơ chế attention trong decoder Transformer đảm bảo rằng việc dự đoán từ tiếp theo trong một chuỗi chỉ dựa trên các từ trước đó.

Tuy nhiên, CLIP không được huấn luyện để sinh văn bản, thay vào đó, CLIP học ghép cặp hình ảnh-văn bản lại với nhau.

Cơ chế **Masked Self-Attention** được tận dụng để, 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.

2.5 Trích xuất đặc trưng

Trong số các vector biểu diễn ngữ cảnh sau khi qua lớp decoder, CLIP chọn **biểu diễn của token [EOS] ở lớp Transformer cuối cùng (lớp cao nhất)** do nó đã "nhìn thấy" tất cả các token trước nó và tổng hợp thông tin từ chúng.

Sau khi trích xuất, vector đặc trưng này được đưa qua LayerNorm và sau đó được chiếu tuyến tính vào một không gian 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.

2.6 Mở rộng quy mô mô hình

CLIP cho 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.

Nhận thấy điều này, họ quyết định mở rộng quy mô tổng thể của mô hình để gia tăng hiệu suất. 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).

3. Tiền huấn luyện với Contrastive Learning

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.

3.1 Đặt vấn đề

Ban đầu, một số phương pháp học biểu diễn hình ảnh từ văn bản đã cố gắng xây dựng mô hình **dự đoán chính xác chủ thích** của một hình ảnh.

Tuy nhiên, phương pháp này gặp phải một số thách thức lớn:

- Quá phức tạp và kém hiệu quả
- Tập trung sai mục tiêu

3.2 Giải pháp

- Proxy task - **Contrastive learning**
- Dự đoán cặp (hình ảnh, văn bản) nào là "đúng" trong một nhóm các lựa chọn.

3.3 Huấn luyện CLIP

1. Tiền xử lý dữ liệu
2. Chuẩn bị Batch dữ liệu và mã hóa
3. Mã hóa song song
4. Tính toán ma trận tương đồng Cosine
5. Hàm mất mát
6. Tối ưu hóa

3.3 Huấn luyện CLIP

1. Tiền xử lý dữ liệu:

- **Tiền xử lý ảnh:** Resize, crop ngẫu nhiên, chuẩn hóa theo ImageNet.
- **Tiền xử lý văn bản:** Token hóa bằng BPE, padding cho đồng đều độ dài.

2. Chuẩn bị Batch dữ liệu và mã hóa:

- Một batch gồm N cặp (hình ảnh, văn bản) thực tế từ bộ dữ liệu WIT.
- Ví dụ: $(I_1, T_1), (I_2, T_2), \dots, (I_N, T_N)$.

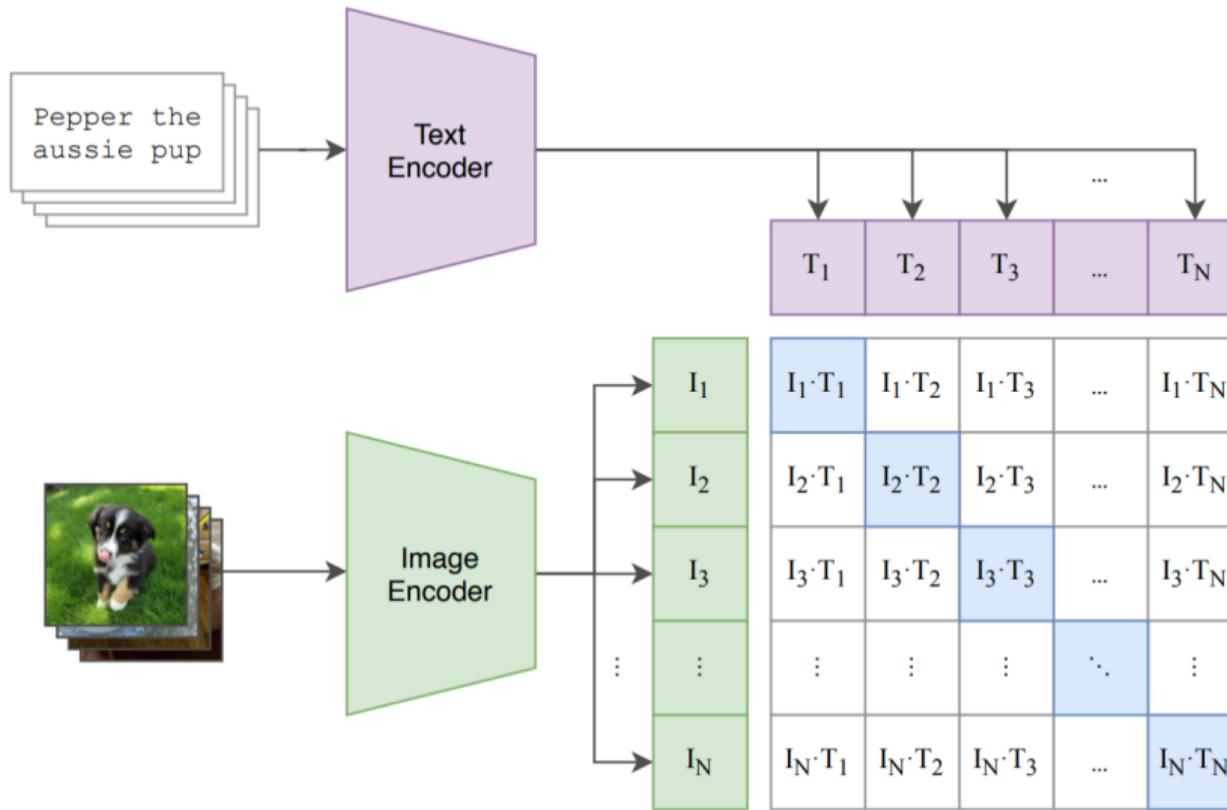
3.3 Huấn luyện CLIP

3. Mã hóa song song:

- **Image Encoder (ResNet/ViT)**: Mã hóa N hình ảnh thành N vector đặc trưng hình ảnh: $E_I = [e_{I1}, e_{I2}, \dots, e_{IN}]$.
- **Text Encoder (Transformer)**: Mã hóa N văn bản thành N vector đặc trưng văn bản: $E_T = [e_{T1}, e_{T2}, \dots, e_{TN}]$.
- Các vector được chiếu (projected) vào cùng một không gian nhúng và chuẩn hóa L2.

4. Tính toán ma trận tương đồng Cosine:

- Tạo ma trận $N \times N$, trong đó phần tử (i, j) là độ tương đồng cosine giữa e_{Ii} và e_{Tj} .
- Các phần tử trên đường chéo chính ($i=j$) đại diện cho các cặp (hình ảnh, văn bản) "đúng"(positive pairs).
- $N^2 - N$ phần tử còn lại là các cặp "sai"(negative pairs).



3.3 Huấn luyện CLIP

5. Hàm mất mát (Symmetric Cross-Entropy Loss):

- Mục tiêu: Tối đa hóa độ tương đồng của các cặp "đúng" và tối thiểu hóa độ tương đồng của các cặp "sai".

- Tính toán loss cho cả hai chiều: image-to-text và text-to-image.

$$\mathcal{L}_{i2t} = - \sum_{i=1}^N \log \frac{\exp(S_{ii}/\tau)}{\sum_{j=1}^N \exp(S_{ij}/\tau)}$$

$$\mathcal{L}_{t2i} = - \sum_{j=1}^N \log \frac{\exp(S_{jj}/\tau)}{\sum_{i=1}^N \exp(S_{ij}/\tau)}$$

- Loss tổng: $(\mathcal{L}_{i2t} + \mathcal{L}_{t2i})/2$.

- τ là tham số nhiệt độ (temperature parameter) có thể học được, điều chỉnh độ "sắc nét" của phân phối xác suất.

6. Tối ưu hóa:

- Sử dụng optimizer AdamW.

- Batch size rất lớn (ví dụ: 32,768) để có đủ lượng mẫu negative hiệu quả.

- Huấn luyện hỗn hợp chính xác (mixed-precision training) để tăng tốc và giảm bộ nhớ.

3.4 Multi-modal Embedding Space - Sản phẩm cốt lõi

- **Không gian thống nhất:** Một không gian vector cao chiều nơi biểu diễn của hình ảnh và văn bản có thể được so sánh trực tiếp.
- **Phản ánh mối quan hệ ngữ nghĩa:**
 - Các cặp có mối quan hệ sẽ có các vector nằm rất gần nhau (độ tương đồng cosine cao).
 - Các cặp không liên quan sẽ có các vector nằm xa nhau (độ tương đồng cosine thấp).
- **Khả năng "Open-set":** Vì được học từ dữ liệu ngôn ngữ tự nhiên đa dạng (WIT), không bị giới hạn bởi các lớp cố định. Có thể hiểu các khái niệm chưa từng thấy trong huấn luyện nếu có mô tả bằng ngôn ngữ tương ứng.

4.1 So sánh các mô hình biến thể 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.

Các biến thể chính và hiệu suất Zero-Shot trên ImageNet

- **ResNet-50:** Độ chính xác 59.6% (baseline)
- **ResNet-101:** Độ chính xác 62.2%
- **ResNet-50x4 (scaled ResNet):** Độ chính xác 65.8%
- **ViT-B/32 (Vision Transformer - Base, patch size 32):** Độ chính xác 63.2%
- **ViT-B/16:** Độ chính xác 68.6%
- **ViT-L/14 (Vision Transformer - Large, patch size 14):** Độ chính xác 75.3%
- **ViT-L/14@336px (fine-tuned với độ phân giải cao hơn):** Độ chính xác 76.2%

4.2 Ưu điểm và hạn chế của CLIP

Ưu điểm nổi bật

- Khả năng Zero-Shot mạnh mẽ.
- Học từ dữ liệu web tự nhiên.
- Mô hình nền tảng (Foundation Model).
- Robustness.

Hạn chế, thách thức

- Khó khăn với tác vụ chi tiết.
- Chi phí tính toán.
- Độ nhạy với Prompt Engineering.
- Dữ liệu "nhiều" và thiên kiến (Bias).
- Không phải là "All-in-one".
- Khả năng trừu tượng hóa hạn chế.

5. Ứng dụng của CLIP

- **Truy vấn** giữa ảnh và văn bản (Text-to-Image, Image-to-Text Retrieval)
- Nâng cao: **Phân biệt khuôn mặt**, từ đó phát triển tác vụ nhận diện khuôn mặt người.

5.1 Ứng dụng của CLIP trong truy vấn giữa ảnh và văn bản

- Text-to-Image, Image-to-Text Retrieval
- CLIP hỗ trợ truy vấn đa chiều: từ văn bản tìm ảnh, hoặc từ ảnh tìm văn bản mô tả tương ứng.
- Ví dụ: Gõ “người đàn ông đeo kính” → truy xuất ảnh phù hợp trong tập dữ liệu.
- Ứng dụng trong tìm kiếm hình ảnh, gợi ý nội dung, kiểm duyệt nội dung tự động,...

5.1.1 Giới thiệu bộ dữ liệu COCO 2017

- **COCO** (Common Objects in Context) là bộ dữ liệu chuẩn cho nhiều bài toán thị giác máy tính: phát hiện đối tượng, phân đoạn ảnh, và truy xuất ảnh từ văn bản.
- **Số lượng:**
 - **Train2017:** 118.000 ảnh có gán nhãn.
 - **Val2017:** 5.000 ảnh dùng để đánh giá mô hình.
 - **Test2017:** 41.000 ảnh (không có nhãn công khai).
- **Chú thích:** Mỗi ảnh đi kèm trung bình 5 câu mô tả tự nhiên (caption) — phù hợp cho huấn luyện/truy vấn ảnh-văn bản.
- **Định dạng:** Ảnh '.jpg' và nhãn định dạng '.json' (COCO-style annotations).
- **Ứng dụng:** Được dùng rộng rãi trong huấn luyện và đánh giá mô hình như CLIP, BLIP, Flamingo, ViLT,...

5.1.2 Thực hiện bài toán truy vấn với CLIP (ViT-B/32)

Ghi chú: Do bài toán yêu cầu huấn luyện trên tập dữ liệu lớn và đòi hỏi tài nguyên GPU đáng kể, mô hình CLIP đã được huấn luyện sẵn bởi OpenAI là một lựa chọn phù hợp và hiệu quả.

Các bước thực hiện:

- **Bước 1:** Nạp mô hình CLIP ViT-B/32 được huấn luyện sẵn từ thư viện OpenAI.
- **Bước 2:** Tiền xử lý ảnh và văn bản đầu vào (chuẩn hóa kích thước, token hóa văn bản).
- **Bước 3:** Trích xuất vector đặc trưng (embedding) cho cả ảnh và văn bản thông qua mô hình CLIP.
- **Bước 4:** Tính độ tương đồng cosine giữa các vector đặc trưng để đo mức độ phù hợp.
- **Bước 5:** Truy xuất kết quả phù hợp nhất: tìm ảnh gần nhất với truy vấn văn bản, hoặc ngược lại.

5.1.3 Kết quả và Nhận xét trên tập COCO

Kết quả đánh giá trên tập COCO validation:

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

Nhận xét:

- CLIP đạt kết quả tốt với Recall@10 trên 83%, cho thấy khả năng định vị đúng ảnh nằm trong top đầu là rất cao.
- Recall@1 còn hạn chế ($\sim 50\%$) – mô hình có thể nhầm khi truy vấn có ngữ nghĩa gần nhau.
- Phù hợp với ứng dụng cần độ bao phủ cao (top-5/top-10), cần cải tiến thêm nếu yêu cầu chính xác tuyệt đối.

5.1.4 Ví dụ minh họa kết quả truy vấn



Ảnh minh họa truy vấn

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:

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

5.2 Ứng dụng CLIP để phân biệt khuôn mặt người

- CLIP học mô hình liên hệ giữa ảnh và văn bản từ hàng trăm triệu cặp dữ liệu.
- Dù không thiết kế riêng cho nhận diện khuôn mặt, CLIP có khả năng biểu diễn ảnh mạnh mẽ.

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.

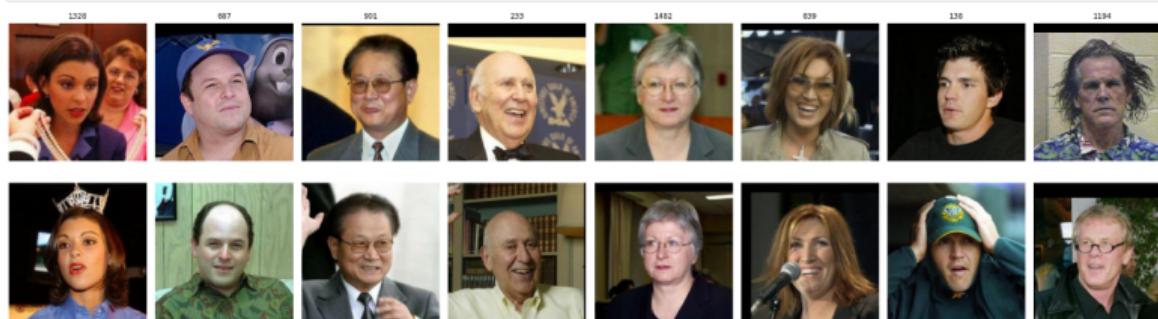
5.2.1 Dữ liệu

Dữ liệu: Kaggle Face Recognition Dataset

- Dựa trên LFW, ảnh 250x250 JPG.
- Mỗi thư mục tương ứng với một người nổi tiếng (2–50 ảnh).

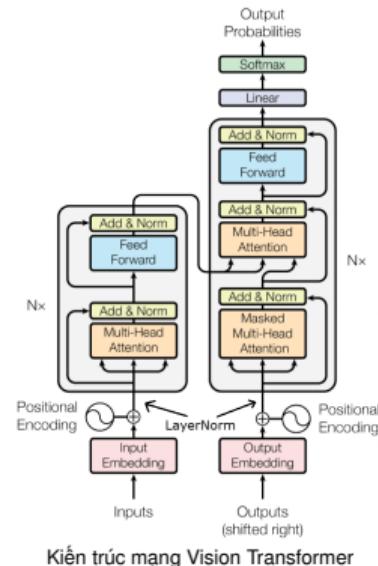
Xử lý dữ liệu:

1. Tăng cường dữ liệu (augmentation).
2. Chia thành Train/Test.
3. Tạo Dataloader với cặp ảnh.



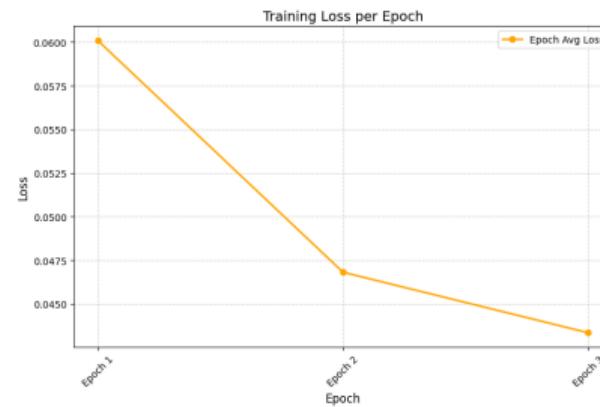
5.2.2 Sử dụng lớp VisionTransformer từ ViT-B/32 (CLIP)

- **Patch Embedding:** Ảnh 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($768 \rightarrow 3072$) → QuickGELU → Linear($3072 \rightarrow 768$).
- **Output LayerNorm:** Chuẩn hóa để lấy embedding có đầu ra 768.



5.2.3 Huấn luyện và Đánh giá mô hình ViT

- **Loss:** Contrastive Loss
- **Optimizer:** Adam (learning rate = 1×10^{-3})
- **Epoch:** 3
- **Các cặp ảnh:** cùng người, khác người
- **Nhận xét:**
 - Sau 3 epoch, loss giảm, mô hình chưa hội tụ. Embedding vector giữa các lớp chưa đủ tách biệt.
 - Khả năng phân biệt ảnh khác người còn yếu, dẫn đến precision thấp trên tập test. Mô hình dễ nhầm lẫn các cặp không cùng người. F1-score ở mức trung bình cho thấy cần cải thiện thêm.



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

5.2.4 Kết quả ảnh test

Thật: 1 | Dự đoán: 1

Ảnh 1
Label: 1357



Ảnh 2
Label: 1357



Thật: 0 | Dự đoán: 1

Ảnh 1
Label: 992



Ảnh 2
Label: 1321



Khoảng cách: 0.1810
Thật: 1, Dự đoán: 1

Khoảng cách: 0.1999
Thật: 0, Dự đoán: 1

- Tính **Khoảng cách** giữa hai vector đặc trưng ảnh.
- Nếu khoảng cách < 0.2 thì hai ảnh được dự đoán là **cùng nhãn**.

5.2.5 So sánh với mô hình CNN khác trong tác vụ phân biệt khuôn mặt người

Mặc dù không được thiết kế chuyên biệt cho tác vụ phân biệt khuôn mặt người, mô hình vẫn đạt hiệu quả cao nhờ chiến lược: **trích xuất embedding cho toàn bộ ảnh, tính khoảng cách, và gán nhãn theo ảnh gần nhất trong tập dữ liệu đã biết.**

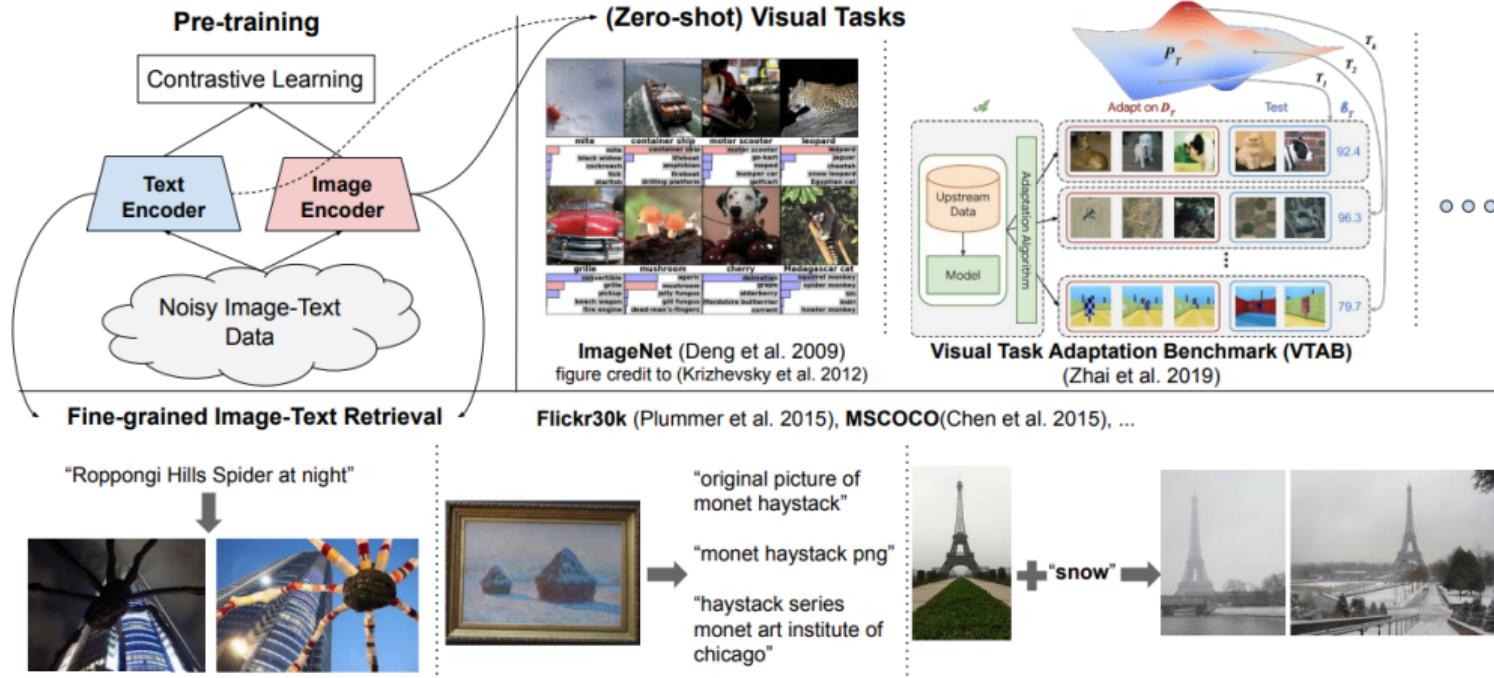
Mô hình	Accuracy	Precision	Recall	F1
CLIP (ViT-B/32)	0.73	0.6982	0.7311	0.7022
ResNet18	0.2151	0.1743	0.2151	0.1672
ResNet50	0.2169	0.1493	0.2169	0.1483
EfficientNet-B0	0.2218	0.1658	0.2218	0.1621
MobileNetV2	0.2602	0.1916	0.2602	0.1923

- CLIP pretrained mang lại hiệu suất cao, không cần huấn luyện lại.
- Ưu tiên sử dụng CLIP nếu tài nguyên tính toán cho phép.

6. Các mô hình tương tự CLIP

- ALIGN: A Large-scale ImaGe and Noisy-text embedding
- BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation

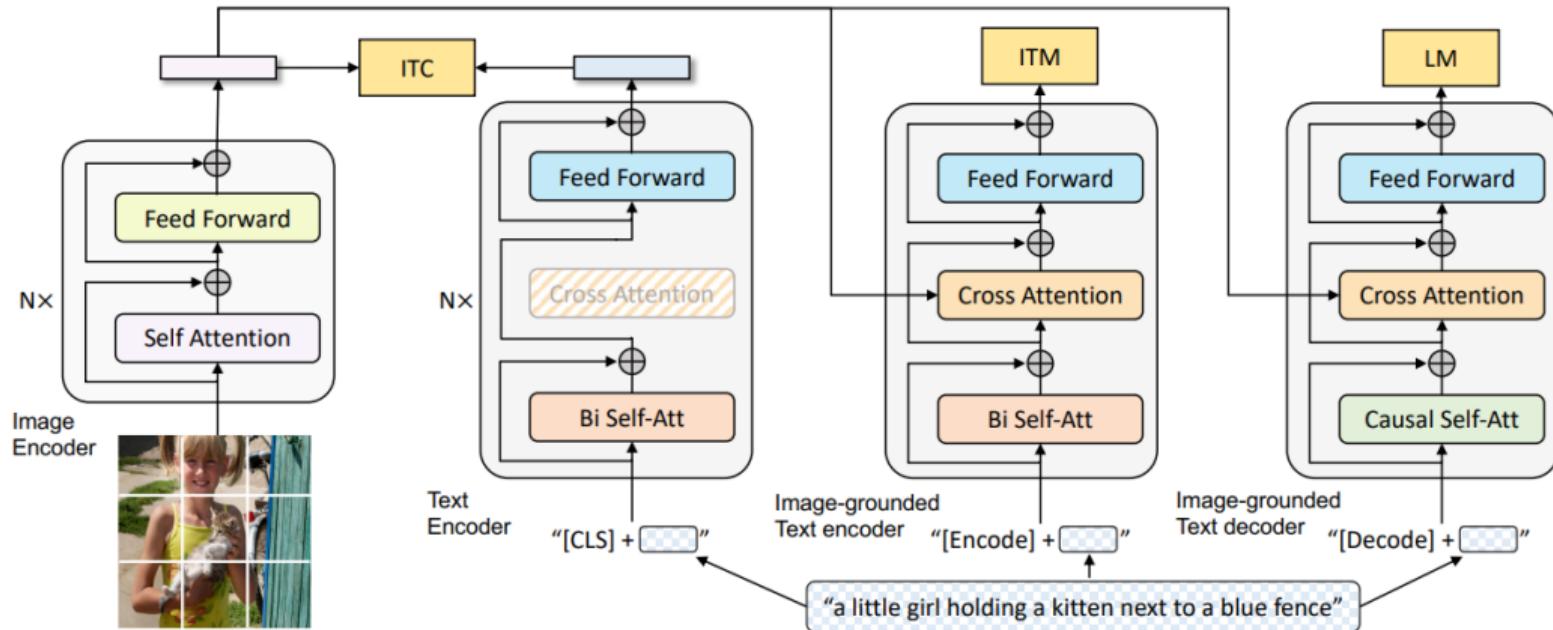
6.1 ALIGN



6.1 ALIGN

- **Mục tiêu:** Học biểu diễn vector dùng cho tác vụ thị giác và thị giác-ngôn ngữ với giả thiết: **Quy mô của dữ liệu có thể bù đắp cho sự nhiễu** của nó.
- **Kỹ thuật:**
 - **Dữ liệu:** 1.8 tỷ cặp ảnh và alt-text thu thập từ web với đặc điểm là rất nhiều, không hoàn hảo. Sử dụng quy trình lọc đơn giản.
 - **Kiến trúc mô hình:** Dual-encoder (EfficientNet và BERT).
 - **Loss function:** Contrastive loss InfoNCE.
- **Cách hoạt động:** Huấn luyện đồng thời Image encoder (EfficientNet) và Text encoder (BERT) với mục tiêu kéo gần các vector embedding đúng và đẩy xa các vector embedding sai trong không gian biểu diễn vector chung.

6.2 BLIP



6.2 BLIP

- **Mục tiêu:** Làm tốt hai nhiệm vụ **hiểu** (understanding) và **sinh** (generation).
- **Kỹ thuật CapFilt:** Tự cải thiện dữ liệu
 1. Huấn luyện BLIP cơ bản, fine-tuning thành 2 model: **Captioner, Filter**.
 2. Dùng Captioner sinh chú thích cho ảnh, dùng Filter lọc các cặp ảnh-văn bản không khớp.
 3. Kết hợp dữ liệu đã làm sạch, làm giàu này cùng các bộ dữ liệu chất lượng cao để huấn luyện BLIP cuối cùng.
- **Kỹ thuật Multimodal Mixture of Encoder-Decoder:**
 - Image/Text Encoder: Mã hóa ảnh/văn bản riêng biệt (ITC).
 - Image-grounded Text Encoder: Kết hợp thông tin hình ảnh vào biểu diễn văn bản (ITM).
 - Image-grounded Text Decoder: Sinh văn bản dựa trên ảnh đầu vào (LM).

6.3 So sánh CLIP, ALIGN và BLIP

Tính năng	ALIGN (Google)	CLIP (OpenAI)	BLIP (Salesforce)
Triết lý	Quy mô dữ liệu bù đắp sự nhiễu	Đặt 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.

7. Kết luận

CLIP (Contrastive Language-Image Pre-Training)

- **CLIP:** Hiểu nội dung hình ảnh và văn bản theo cách liên kết chúng lại với nhau.
- **Mô hình:** 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.