

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA CÔNG NGHỆ THÔNG TIN 1



ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

**ĐỀ TÀI: “TẠO ẢNH NGƯỜI TỪ MÔ TẢ TIẾNG VIỆT
SỬ DỤNG MÔ HÌNH KHUẾCH TÁN ỔN ĐỊNH”**

Giảng viên hướng dẫn	: PGS.TS. NGÔ XUÂN BÁCH
Sinh viên thực hiện	: NGUYỄN DUY DŨNG
Lớp	: D19HTTT02
Khoá	: 2019 – 2024
Hệ	: ĐẠI HỌC CHÍNH QUY

Hà Nội, tháng 12/2023

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn tới các thầy, cô trong Khoa Công nghệ thông tin 1 tại Học viện Công nghệ Bưu chính Viễn thông, trong hơn 4 năm vừa qua đã truyền đạt cho em biết bao kiến thức, kinh nghiệm quý báu cũng như những hành trang, những kỹ năng giúp em vững bước vào tương lai.

Em xin đặc biệt gửi lời cảm ơn đến thầy Ngô Xuân Bách, người đã dẫn dắt và hỗ trợ rất nhiều cho em trong quá trình học tập cũng như thực hiện đồ án.

Em xin gửi sự biết ơn sâu sắc nhất đến gia đình mình, những người luôn sát cánh và là nguồn động lực, là hậu phương vững chắc cho em trong suốt những năm tháng theo học tại Học viện Công nghệ Bưu chính Viễn thông.

Do kinh nghiệm còn hạn chế nên đồ án chắc chắn còn nhiều thiếu sót, em mong nhận được sự góp ý chân thành cũng như chỉ bảo tận tình từ thầy, cô.

Em xin chân thành cảm ơn.

Hà Nội, tháng 12 năm 2023

Sinh viên thực hiện

Nguyễn Duy Dũng

LỜI CAM ĐOAN

Tôi xin cam đoan những khảo sát, nghiên cứu là do tôi thực hiện và tìm hiểu dưới sự hướng dẫn của PGS.TS. Ngô Xuân Bách.

Tất cả bài báo, tài liệu, công cụ, mã nguồn của các tác giả khác được sử dụng trong đồ án đều được trích dẫn tường minh về nguồn và nhóm tác giả trong phần danh sách tài liệu tham khảo.

Hà Nội, tháng 12 năm 2023

Sinh viên thực hiện

Nguyễn Duy Dũng

**NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP
CỦA GIẢNG VIÊN PHẢN BIỆN**

Giảng viên phản biện:..... Bộ môn:.....

NỘI DUNG NHẬN XÉT:

I. Nội dung báo cáo:

.....
.....
.....
.....
.....
.....
.....

..... **Sản phẩm:**

.....
.....

..... **Ưu nhược điểm:**

.....
.....

..... **Kết luận:**

.....
.....
.....

Điểm:..... (bằng chữ.....).

*Hà Nội, ngày.....tháng.....năm
20.....*

**GIẢNG VIÊN PHẢN
BIỆN**

NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP CỦA GIẢNG VIÊN HƯỚNG DẪN

NỘI DUNG NHẬN XÉT:

I. Nội dung báo cáo:

.....
.....
.....
.....
.....
.....
.....

..... **Sản phẩm:**

.....
.....

..... **Ưu nhược điểm:**

.....
.....

..... **Kết luận:**

.....
.....
.....
.....

Điểm:..... (bằng
chữ.....).

*Hà Nội, ngày.....tháng.....năm
20.....*

GIẢNG VIÊN HƯỚNG
DẪN

MỤC LỤC

LỜI CẢM ƠN.....	i
DANH MỤC THUẬT NGỮ.....	v
DANH MỤC HÌNH VẼ.....	vi
DANH MỤC BẢNG BIỂU.....	vii
LỜI MỞ ĐẦU.....	1
CHƯƠNG 1. GIỚI THIỆU BÀI TOÁN.....	3
1.1 Giới thiệu bài toán.....	3
1.3 Một số nghiên cứu liên quan.....	5
1.4 Kiến thức cơ bản.....	7
1.4.1 CNN.....	7
1.4.2 U-Net.....	10
1.4.3 GAN.....	12
1.4.4 AE và VAEs.....	13
1.4.5 Diffusion.....	15
1.4.6 So sánh GAN, VAEs và Diffusion.....	16
1.4.7 CLIP và M-CLIP.....	17
1.4.8 Vision Transformers.....	19
1.5 Phạm vi nghiên cứu.....	21
1.6 Đóng góp của đồ án.....	21
1.7 Kết luận chương.....	21
CHƯƠNG 2. TẠO ẢNH NGƯỜI TỪ MÔ TẢ TIẾNG VIỆT SỬ DỤNG MẠNG KHUẾCH TÁN ỔN ĐỊNH.....	22
2.1 Mô hình Stable Diffusion.....	22
2.2 Mô hình UPGPT.....	26
2.3 Mô hình U-ViT.....	28
2.4 Kết luận chương.....	29
CHƯƠNG 3. DỮ LIỆU BÀI TOÁN TẠO ẢNH NGƯỜI.....	30
3.1 Dữ liệu DeepFashion-MultiModal.....	30
3.2 Xử lý dữ liệu sang tiếng Việt.....	32
3.2.1 Thống nhất định nghĩa.....	33
3.2.2 Phương pháp dịch từ tiếng Anh sang tiếng Việt.....	36
3.3 Kết luận chương.....	38
CHƯƠNG 4. THỰC NGHIỆM ĐÁNH GIÁ.....	39
5.1 Quy trình thực nghiệm.....	39
5.2 Thiết lập thực nghiệm.....	39
5.2.1 Môi trường và công cụ thực nghiệm.....	41

5.2.2 Chi tiết thực nghiệm.....	42
5.3 Các chỉ số đánh giá cho bài toán.....	42
5.3.1 Inception Score (IS).....	43
5.3.2 Fechet Inception Distance (FID).....	44
5.3.3 Structural Similarity Index Measurement (SSIM).....	45
5.4 Kết quả thực nghiệm.....	45
5.5 Phân tích lỗi sai.....	46
5.6 Kết luận chương.....	47
KẾT LUẬN.....	48
A: PHỤ LỤC.....	49
A.1 Tổng quan hệ thống.....	49
A.2 Các công cụ sử dụng.....	49
A.3 Xây dựng hệ thống.....	49
A.4 Một số kết của của mô hình.....	50
TÀI LIỆU THAM KHẢO.....	52

DANH MỤC THUẬT NGỮ

ST T	Từ viết tắt	Tiếng Anh	Tiếng Việt/ Giải thích
1	Text2Image	Text to Image	Chuyển đổi văn bản thành ảnh
2	Image2Text	Image to Text	Chuyển đổi ảnh thành văn bản
3	ViT	Vision Tranfomers	Mô hình dùng để xử lý ảnh
4	CNN	Convolutional Neural Network	Mạng tích chập
5	SDN	Stable Diffusion Network	Mạng khuếch tán ổn định
6	LDM	Latent Diffsion Network	Mạng khuếch tán tiềm ẩn
7	AE	Autoencoder	Mạng tự động mã hóa

DANH MỤC HÌNH VẼ

Hình 1 Ảnh ví dụ về bài toán Text to Human.....	3
Hình 2 Tổng quan về các mô hình sinh ảnh [4].....	5
Hình 3 Mốc thời gian phát triển các mô hình sinh ảnh [5].....	6
Hình 4 Ví dụ mạng CNN với kiến trúc VGG-16 [15].....	8
Hình 5 Minh họa tính toán tích chập [16].....	8
Hình 6 Minh họa việc tính toán trên lớp Max Pooling [17].....	9
Hình 7 Minh họa lớp kết nối toàn bộ [18].....	10
Hình 8 Kiến trúc mô hình U-Net [19].....	11
Hình 9 Minh họa quá trình huấn luyện GAN [20].....	12
Hình 10 Kiến trúc và hàm mất mát của Autoencoder [21].....	13
Hình 11 Kiến trúc và hàm mất mát của Variational AutoEncoders [21].....	14
Hình 12 Minh họa quá trình diffusion [1].....	15
Hình 13 Điểm mạnh, điểm yếu của GANs, VAEs, Diffusion Models [22].....	16
Hình 14 Zero-shot với mô hình CLIP [23].....	17
Hình 15 Quá trình huấn luyện M-CLIP [24].....	18
Hình 16 Minh họa cơ chế Attention của Vision Transformers [25].....	20
Hình 17 Kiến trúc Vision Transformers [25].....	20
Hình 18 Các thành phần của Stable Diffusion [26].....	22
Hình 19 Quá trình sinh ảnh từ văn bản của Stable Diffusion [26].....	23
Hình 20 Quá trình huấn luyện U-Net [26].....	23
Hình 21 Quá trình khử nhiễu bằng U-Net [26].....	24
Hình 22 Diffusion trong latent space [26].....	24
Hình 23 Kiến trúc mô hình LDM/SD [27].....	25
Hình 24 Các bài toán UPGPT có thể xử lý [28].....	26
Hình 25 Kiến trúc tổng quan của UPGPT [28].....	27
Hình 26 Một vài mẫu của mô hình UPGPT [28].....	28
Hình 27 Kiến trúc U-ViT [31].....	29
Hình 28 Ví dụ dữ liệu DeepFashion-MultiModal.....	30
Hình 29 Biểu đồ tròn thể hiện tỉ lệ các loại quần áo.....	33
Hình 30 Ví dụ về các mẫu dịch.....	37
Hình 31 Cấu trúc file mô tả tiếng Việt.....	39
Hình 32 Trích xuất thông tin từ ảnh.....	40
Hình 33 Ví dụ về độ đo IS.....	43
Hình 34 Minh họa FID.....	44
Hình 35 Lỗi mờ mặt.....	46
Hình 36 Lỗi sai kiểu dáng khó.....	47
Hình 37 Lỗi không nhìn rõ trang sức nhỏ.....	47
Hình 38 Kiến trúc hệ thống sinh ảnh người sử dụng mô tả tiếng Việt.....	50

DANH MỤC BẢNG BIỂU

Bảng 1 Bảng kết quả của các mô hình trên tập MS-COCO [5].....	7
Bảng 2 Định dạng dữ liệu DeepFashion-MultiModal.....	31
Bảng 3 Nhận human parsing của DeepFashion-MultiModal.....	31
Bảng 4 Nhận kiểu dáng của DeepFashion-MultiModal.....	32
Bảng 5 Nhận họa tiết màu sắc của DeepFashion-MultiModal.....	32
Bảng 6 Thống kê số lượng dữ liệu.....	33
Bảng 7 Các loại nhãn quần áo tiếng Việt.....	34
Bảng 8 Các loại vải tiếng Việt.....	34
Bảng 9 Các loại họa tiết tiếng Việt.....	34
Bảng 10 Các loại kiểu dáng quần áo tiếng Anh.....	35
Bảng 11 Các loại kiểu dáng quần áo tiếng Việt.....	36
Bảng 12 Thang đo chất lượng dịch.....	36
Bảng 13 Bảng điểm sau khi thực hiện gán nhãn bằng tay.....	37
Bảng 14 Các mẫu prompt dùng để dịch dữ liệu.....	38
Bảng 15 Thông số phân cứng.....	41
Bảng 16 Môi trường phát triển.....	41
Bảng 17 Kết quả thực nghiệm.....	42
Bảng 18 Các độ đo của bài toán sinh ảnh.....	43
Bảng 19 Kết quả thực nghiệm.....	45
Bảng 20 Kết quả với mô tả là tiếng Anh.....	46

LỜI MỞ ĐẦU

Những năm gần đây, Trí tuệ nhân tạo (AI) đã có sự tiến bộ vượt bậc có thể giải quyết nhiều bài toán thực tế trong cuộc sống. Khả năng tạo ảnh từ mô tả mang nhiều thách thức và hứa hẹn. Sự phát triển của mô hình khuếch tán ổn định (Stable Diffusion) đã mở ra cánh cửa cho việc biến các mô tả chữ viết thành các hình ảnh sống động với độ chân thực và sáng tạo. Qua sự kết hợp giữa xử lý ngôn ngữ tự nhiên (natural language processing) và thị giác máy tính (computer vision) việc biến ý tưởng từ dòng văn bản thành hình ảnh đã trở thành hiện thực tạo tiền đề cho nhiều lĩnh vực khác nhau.

Trong đó, bài toán tạo ảnh người từ mô tả (Text to Human) là một trong các bài toán thu hút nhiều sự quan tâm của một nhóm nhà nghiên cứu trong lĩnh vực trí tuệ nhân tạo và trở thành một bài toán thú vị nhưng đầy thách thức. Tạo ảnh người từ mô tả giúp tự động tạo ra các hình ảnh người theo dựa theo các đặc điểm được cung cấp từ mô tả. Bài toán này có nhiều ứng dụng trong thực tế, trong đó ứng dụng nổi bật nhất là tạo, chỉnh sửa ảnh như Adobe Firefly.

Hiện nay không có nghiên cứu về vấn đề này đối với tiếng Việt do nguồn dữ liệu còn hạn chế. Trong khi đó các phương pháp tạo ảnh dựa trên mô tả tiếng Anh đã có các kết quả nhất định. Vì vậy trong đồ án này có cung cấp tập dữ liệu và đề xuất các phương pháp chuyên dụng dành riêng cho bài toán tạo ảnh người từ mô tả tiếng Việt. Các phương pháp này đều sử dụng mô hình khuếch tán ổn định (Stable Diffusion) sử dụng đặc trưng mô tả như một lời nhắc, để điều khiển mô hình tạo ra ảnh người.

Cụ thể, đồ án này trình bày về cách tạo tập dữ liệu tiếng Việt từ tập dữ liệu tiếng Anh (tập dữ liệu liDeepFashion-MultiModal) và các phương pháp U-ViT và UPGPT.

DeepFashion-MultiModal là một tập dữ liệu có chứa các ảnh người và mô tả với ảnh chất lượng cao. Các mô tả sẽ được dịch sang tiếng Việt để làm dữ liệu huấn luyện và kiểm tra.

Về phương pháp, cả 2 đều được tinh chỉnh để có thể dùng cho tiếng Việt. UPGPT được tối ưu riêng dành cho bài toán tạo ảnh người, sử dụng AutoEncoder và dùng kiến trúc U-Net cho phần Diffusion. Còn U-ViT thì dùng kiến trúc Vision Transformers cũng mang lại kết quả tốt trong việc tạo ảnh từ mô tả. UPGPT đã đạt được kết quả tốt nhất trên tập kiểm tra với điểm FID là 30.54 trong khi đó phương pháp U-ViT có điểm FID là 36.16. Đây là các kết quả đầu tiên cho thấy việc tạo ảnh người từ mô tả tiếng Việt là có thể thực hiện được.

Nội dung đồ án được bố cục thành 4 chương như sau:

Chương 1: Giới thiệu bài toán

Nội dung chương 1 sẽ khái quát các vấn đề, phương pháp cho bài toán tạo ảnh từ mô tả, đưa ra các kiến thức cơ bản và khảo sát về các phương pháp học sâu đã được áp dụng cho bài toán. Đồng thời nêu tổng quan về phạm vi nghiên cứu và đóng góp của đồ án này.

Chương 2: Tạo ảnh người từ mô tả tiếng Việt sử dụng mạng khuếch tán ổn định

Nội dung chương 2 trình bày về kiến trúc tổng quát và các mô hình học sâu được sử dụng trong đồ án. Đồng thời đồ án còn trình bày cơ sở lý thuyết, kiến trúc của các khối để giải thích.

Chương 3: Dữ liệu của bài toán tạo ảnh người

Nội dung chương 3 trình bày rõ về bộ dữ liệu gốc dùng để dịch và quy trình dịch tập dữ liệu đó sang tiếng Việt.

Chương 4: Thực nghiệm và đánh giá

Nội dung chương 4 trình bày quy trình huấn luyện, tổng hợp kết quả và phân tích các kết quả đạt được cũng như một số lỗi sai gặp phải.

Phụ lục A: Hệ thống sinh ảnh người từ mô tả tiếng Việt

Phụ lục này mô tả hệ thống sinh ảnh người từ mô tả tiếng Việt được phát triển từ nghiên cứu này. Hệ thống sẽ tự động trích xuất thông tin đầu vào là một đoạn mô tả người miền thời trang để sinh ra ảnh tương ứng.

CHƯƠNG 1. GIỚI THIỆU BÀI TOÁN

Trong chương 1, đồ án trình bày cái nhìn tổng quan về bài toán sinh ảnh từ mô tả, bao gồm giới thiệu bài toán, ứng dụng, một số nghiên cứu liên quan, các kiến thức cơ bản cùng với phạm vi đóng góp của đồ án.

1.1 Giới thiệu bài toán

Tạo ảnh từ mô tả (Text to Image) là việc sinh ra các hình ảnh liên quan, dựa trên các mô tả bằng ngôn ngữ tự nhiên. Một mô tả có thể có nhiều hình ảnh biểu diễn và một hình ảnh tốt sẽ thể hiện toàn bộ nội dung của mô tả. Bài toán này chính là một thách thức trong việc hiểu ngữ và là một sự kết hợp thú vị giữa hai lĩnh vực thị giác máy tính và xử lý ngôn ngữ tự nhiên.

Tạo ảnh người từ mô tả (Text to Human) là một bài toán nhỏ hơn của Text to Image. Trong đó, chúng ta tập trung vào việc sử dụng thông tin mô tả về một người để tạo ra hình ảnh tương ứng. Các thông tin có thể có trong mô tả có thể là thông tin của các loại quần áo, kiểu tóc, dáng đứng, v.v. Ví dụ “*Người phụ nữ mặc áo phông ngắn tay có họa tiết tron màu, váy ngắn vải bò.*” là mô tả của hình 1.



Hình 1 Ảnh ví dụ về bài toán Text to Human

Tổng quát, một bài toán tạo ảnh người từ mô tả sẽ có đầu vào là một đoạn văn bản miêu tả của một người bất kỳ và đầu ra là một hình ảnh người tương ứng với nội dung của mô tả.

Hầu hết các phương pháp tạo ảnh người từ mô tả sử dụng kiến trúc bộ mã hóa - giải mã (encoder-decoder), trong đó mô tả đầu vào được mã hóa thành một biểu diễn trung gian của thông tin trong mô tả, sau đó được giải mã thành hình ảnh.

1.2 Ứng dụng của bài toán

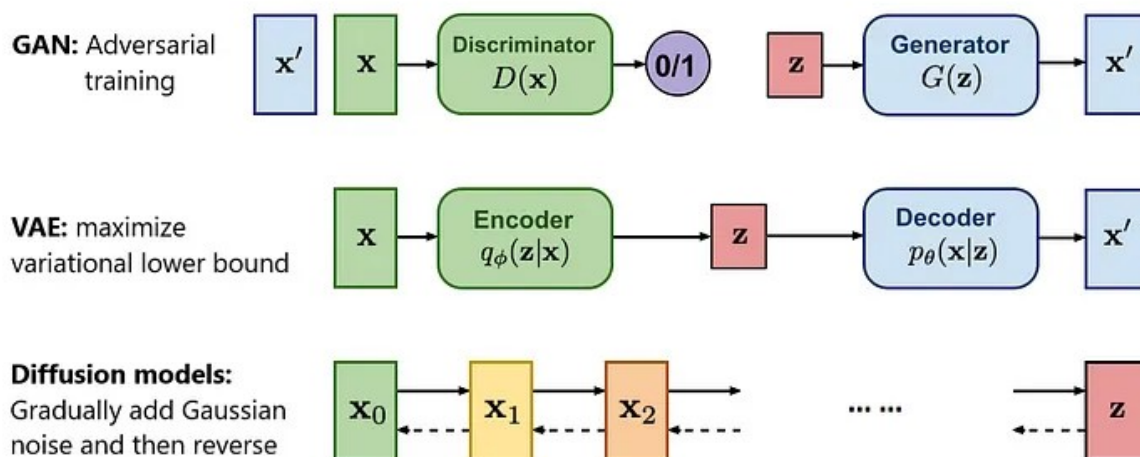
Mục đích chính của bài toán là tự động hóa việc tạo ra một hình ảnh người dựa trên các mô tả được cung cấp. Hiện tại, việc tạo ảnh từ mô tả đã có các ứng dụng như midjourney, GPT-4,... các ứng dụng cung cấp chức năng tạo ảnh từ mô tả cực kì ấn tượng, trong đó bao gồm cả việc tạo ảnh người.

Bài toán tạo ảnh người có thể áp dụng để giải quyết các bài toán trong thực tế như sau:

- Phục vụ điều tra tội phạm: Có thể sử dụng để tạo các hình ảnh phỏng đoán dựa trên mô tả của nhân chứng hoặc nạn nhân, giúp cảnh sát xác định nhanh chóng những người liên quan đến vụ án.
- Hỗ trợ tạo nhân vật game: Cho phép người chơi tạo hình ảnh nhân vật theo ý muốn từ mô tả, tăng tính tương tác và cá nhân hóa trong trò chơi hoặc môi trường ảo.
- Tạo hình ảnh đại diện: Cho phép người dùng tạo ra các ảnh đại diện theo mô tả cá nhân của họ một cách sáng tạo và thú vị.
- Nén dữ liệu: Dữ liệu dạng văn bản luôn chiếm ít bộ nhớ hơn dữ liệu dạng ảnh, chúng ta có thể dùng Image2Text để chuyển dữ liệu dạng ảnh người về dạng văn bản mô tả người, sau đó dùng Text2Image để chuyển lại từ mô tả người về dạng ảnh.
- Hỗ trợ bài toán chỉnh sửa ảnh: Việc thay đổi các bộ quần áo, trang sức trên ảnh không còn khó khăn, người dùng có thể sử dụng để thay đổi các chi tiết trang phục trong ảnh để thử trước trước khi mua quần áo.

1.3 Một số nghiên cứu liên quan

Các thuật toán tạo ảnh hiện nay hầu hết dựa vào 3 phương pháp chính bao gồm: mô hình Diffusion [1], mạng GAN [2] và Variational Autoencoder [3].



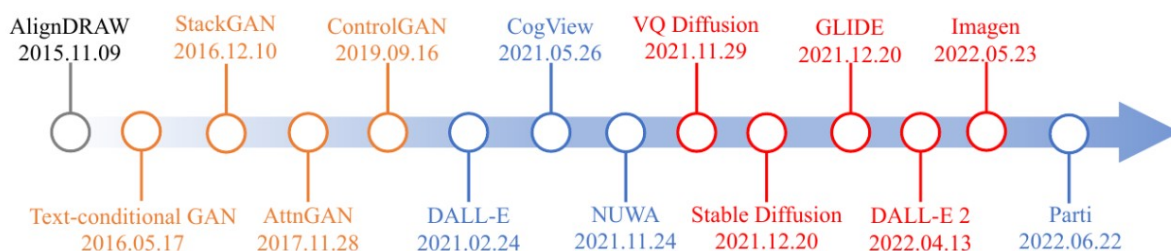
Hình 2 Tổng quan về các mô hình sinh ảnh [4]

GAN học để tạo ra dữ liệu mới giống với tập dữ liệu huấn luyện. Nó bao gồm hai mạng neural, một bộ tạo và một bộ phân biệt, chơi một trò chơi hai người chơi. Bộ tạo nhận giá trị ngẫu nhiên được lấy mẫu từ phân phối chuẩn và tạo ra một mẫu tổng hợp, trong khi bộ phân biệt cố gắng phân biệt giữa mẫu thực và mẫu được tạo ra. Bộ tạo được huấn luyện để tạo ra đầu ra có tính thực tế có thể đánh lừa bộ phân biệt, trong khi bộ phân biệt được huấn luyện để phân biệt chính xác giữa dữ liệu thực và dữ liệu được tạo ra. Hàng đầu của Hình 2 hiển thị sơ đồ hoạt động của nó.

VAEs bao gồm một bộ mã hóa và một bộ giải mã. Bộ mã hóa ánh xạ dữ liệu đầu vào là ảnh sang một biểu diễn chiều thấp, trong khi bộ giải mã cố gắng tái tạo lại dữ liệu đầu vào ảnh ban đầu bằng cách ánh xạ biểu diễn này trở lại dạng ban đầu của nó. Bộ mã hóa đầu ra phân phối chuẩn của mã tiềm ẩn dưới dạng biểu diễn chiều thấp bằng cách dự đoán các vector trung bình và độ lệch chuẩn. Hàng giữa của Hình 2 minh họa cách thức hoạt động của nó.

Mô hình diffusion gồm hai quá trình: diffusion xuôi và diffusion ngược. Quá trình diffusion xuôi là một chuỗi Markov mà từ từ thêm nhiễu vào dữ liệu đầu vào cho đến khi nó trở thành nhiễu trắng. Quá trình này không học và thường mất khoảng 1000 bước để hoàn thành. Quá trình diffusion ngược nhằm đảo ngược từng bước của quá trình diffusion xuôi để loại bỏ nhiễu và khôi phục lại dữ liệu gốc. Quá trình diffusion ngược được thực hiện thông qua một mạng neural có thể huấn luyện được. Hàng dưới của Hình 2 minh họa cho quá trình này.

Quá trình phát triển các mô hình tạo ảnh dựa trên các phương pháp đã liệt kê được thể hiện trong hình 3.



Hình 3 Mốc thời gian phát triển các mô hình sinh ảnh [5]

Mô hình Text2Image đầu tiên là AlignDRAW [6], đã được giới thiệu vào năm 2015 bởi các nhà nghiên cứu đến từ đại học Toronto ở Canada. AlignDRAW là cải tiến của kiến trúc DRAW [7] (Deep Recurrent Attentive Writer - sử dụng recurrent variational autoencoder và cơ chế attention). Các hình ảnh tạo ra bởi AlignDRAW mờ và không chân thực nhưng mô hình có khả năng tổng quát hóa và tạo ra các đối tượng không có trong tập huấn luyện.

Thế hệ tiếp theo sử dụng GAN để tạo ảnh, mô hình đầu tiên là Text-conditional GAN [8]. Mô hình với 2 thành phần Generator (học cách tạo ảnh từ văn bản) và Discriminator (học cách phân loại giữa ảnh được tạo và ảnh thật). Đặt nền móng cho các mô hình tiếp đó là StackGAN [9], AttnGAN [10], ControlGAN [11].

DALL-E [12] là một mô hình tạo hình ảnh đột phá của OpenAI, được công bố vào đầu năm 2021. Tên của nó kết hợp giữa tên của Salvador Dalí, nghệ sĩ nổi tiếng với tác phẩm sáng tạo, và WALL-E, nhân vật trong phim hoạt hình nổi tiếng của Pixar. DALL-E sử dụng mô hình Transformer, cùng với một kiến trúc mở rộng, để tạo ra hình ảnh từ mô tả văn bản.

Cuối năm 2021 khởi điểm bắt đầu của sự bùng nổ các mô hình tạo ảnh một cách ấn tượng. Các mô hình đều phát triển dựa trên kiến trúc diffusion, trong đó có stable diffusion sẽ được nói chi tiết trong bài luận này.

Bảng 1 dưới đây là kết quả của một số mô hình nổi bật trên tập dữ liệu MS-COCO [13]. Chúng ta có thể thấy rằng với tiếng Anh các mô hình hoạt động tốt và cho ra điểm FID thấp.

Mô hình	FID ↓
CogView	27.10
LAFITE	26.94
DALLE	17.89
GLIDE	12.24
Imagen	7.27

Stable Diffusion	12.63
VQ-Diffusion	13.86
DALL-E 2	10.39
Upainting	8.34
ERNIE-ViLG 2.0	6.75
eDiff-I	6.95

Bảng 1 Bảng kết quả của các mô hình trên tập MS-COCO [5]

Đối với bài toán tạo ảnh người từ mô tả, đây là một nhánh nhỏ của tạo ảnh từ mô tả. Các giải pháp để giải quyết bài toán đều dựa trên bài toán tạo ảnh từ mô tả nhưng sẽ có những cải tiến để tối ưu cho việc tạo ảnh người.

Nổi bật nhất là bài báo Text2Human: Text-Driven Controllable Human Image Generation [14], nhóm tác giả đã cung cấp tập dữ liệu ảnh người được đánh mô tả thủ công với ảnh người chất lượng cao. Bên cạnh đó nhóm tác giả còn cung cấp giải pháp dùng VQVAE tạo ảnh người kết hợp các thông tin khác ngoài văn bản như DensePose (thông tin 3D của dáng người được thể hiện dưới dạng 2D), Human Parsing (thông tin các vị trí tóc, mặt, áo, v.v. của người trong ảnh).

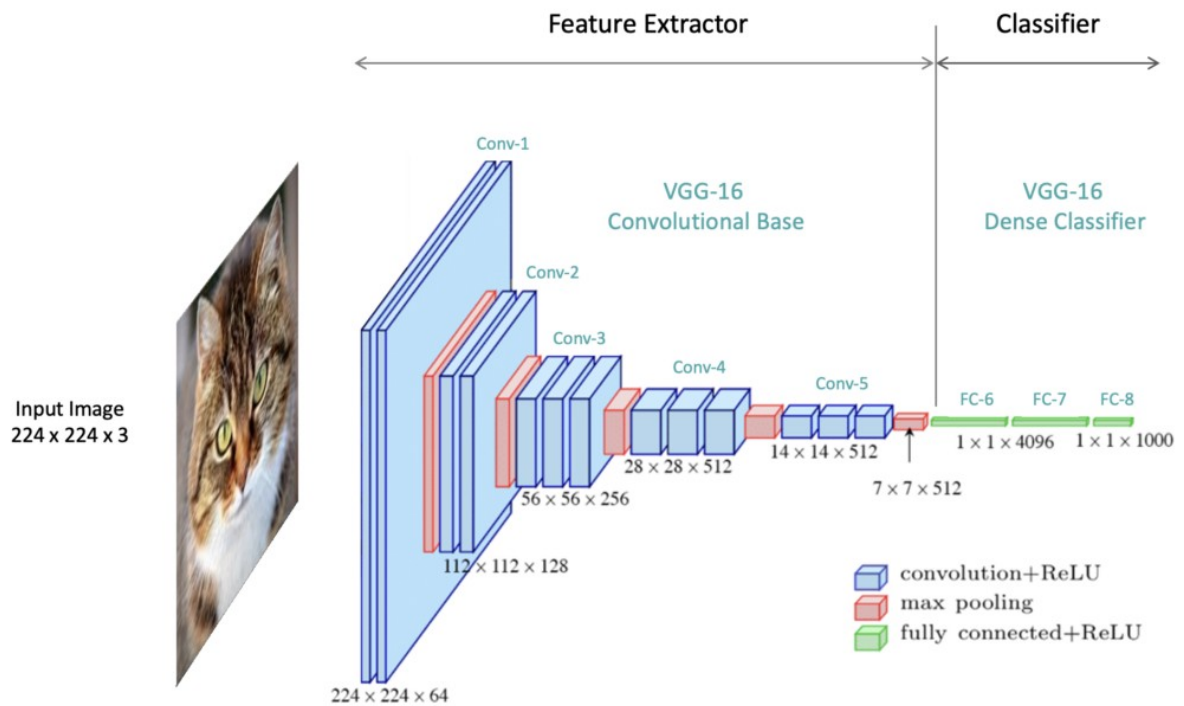
Đối với tiếng Việt, bài toán tạo ảnh người bằng mô tả chưa được phát triển do những khó khăn về dữ liệu, việc xây dựng một bộ dữ liệu yêu cầu cả hình ảnh và mô tả tương ứng vốn tốn rất nhiều thời gian, công sức, yêu cầu kiến thức chuyên môn để tạo ra mô tả. Bên cạnh đó cần một lượng lớn tài nguyên tính toán huấn luyện model và tạo ra các pre-trained cần thiết.

1.4 Kiến thức cơ bản

Để hiểu rõ hơn phương pháp sẽ dùng để giải quyết bài toán, đề án cung cấp các kiến thức cơ bản về các thành phần, mô hình, đưa ra sự so sánh giữa các phương pháp lý giải cho việc chọn mô hình khuếch tán (Diffusion) làm tiền đề giải quyết bài toán trong đề án.

1.4.1 CNN

Mạng nơ-ron tích chập (Convolutional Neural Networks - CNN) được phát triển để xử lý dữ liệu hình ảnh hiệu quả hơn. Điều này phần lớn nhờ vào việc sử dụng các phép toán tích chập để trích xuất các đặc trưng từ hình ảnh. Điều này cho phép chúng ta phát hiện các mẫu đặc trưng không thay đổi theo vị trí khi nhân chập di chuyển trên hình ảnh. Cách tiếp cận này cải thiện hiệu quả mô hình bằng cách giảm đáng kể tổng số tham số cần huấn luyện so với các lớp kết nối đầy.

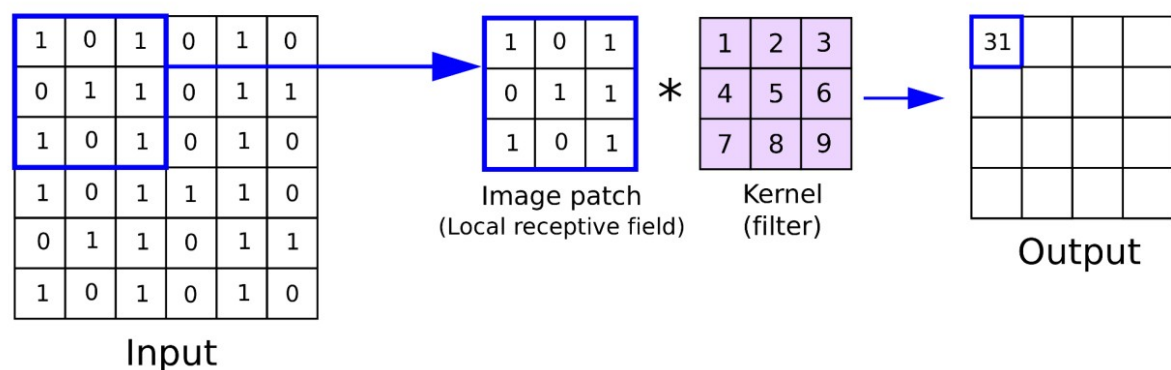


Hình 4 Ví dụ mạng CNN với kiến trúc VGG-16 [15]

Một mạng CNN bao gồm chuỗi các lớp liên tiếp, mỗi lớp sẽ có các hàm kích hoạt riêng. Có 3 lớp chính để xây dựng một mạng CNN: lớp tích chập (Convolutional layer), lớp pooling (Pooling layer), lớp kết nối toàn bộ (Fully-connected layer). Sau đây đồ án sẽ trình bày chi tiết về quá trình xử lý trong 3 lớp. Ví dụ hình 4, kiến trúc VGG-16 sử dụng mạng CNN để phân loại ảnh.

a) Lớp tích chập

Nhiệm vụ chính của lớp tích chập trích chọn đặc trưng một cách tự động mà không phải làm thủ công như các phương pháp khác trong Machine Learning.



Hình 5 Minh họa tính toán tích chập [16]

Hình 5 mô tả 1 lần tính toán sử dụng tích chập. Bộ lọc có kích thước thường là 3x3 hoặc 5x5 và được cho chạy dọc bức ảnh. Mỗi phần được trượt qua sẽ tính toán cho ra 1 giá trị duy nhất, công việc này được lặp lại đến khi bộ lọc trượt hết qua toàn bộ ảnh theo cả 2 chiều. Đầu ra của phép tích chập là một tập các giá trị của ảnh được gọi

là features map. Ở các lớp đầu tiên, phép tích chập đơn giản là phép tìm biên ảnh để làm hiện lên các đặc trưng của đối tượng trong ảnh như các góc ảnh, đường xung quanh đối tượng,... Các lớp tiếp theo sẽ có nhiệm vụ trích xuất các đặc trưng ở mức chi tiết của đối tượng đó.

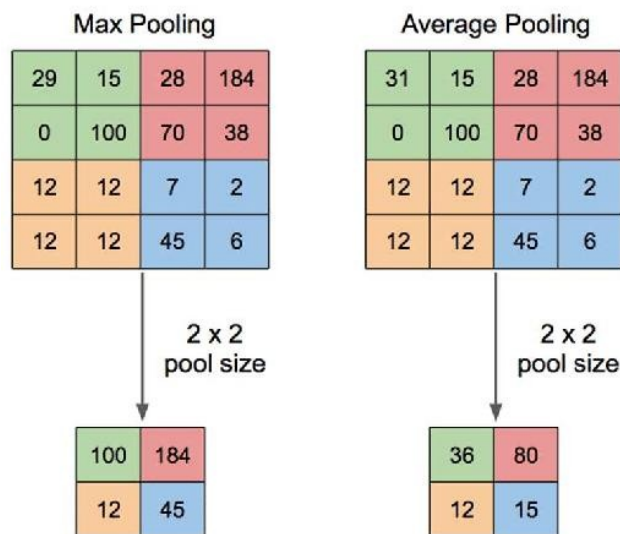
Một số khái niệm cơ bản trong lớp tích chập:

- **Filter hoặc Feature Detector:** ma trận lọc, thông thường có kích thước 3x3 hoặc 5x5.
- **Convolved Feature, Activation Map hoặc Feature map:** là đầu ra của ảnh khi cho bộ lọc chạy qua khi sử dụng phép tính tích vô hướng.
- **Receptive Field:** là các vùng nhỏ trong ảnh để tính tích chập có kích thước giống với bộ lọc.
- **Depth:** số lượng bộ lọc.
- **Stride:** là khoảng cách dịch chuyển của bộ lọc sau mỗi lần tính tích chập trên 1 receptive field. Ví dụ với stride=1 tương đương việc dịch sang phải hoặc xuống dưới 1 pixel tùy vào vị trí của vùng ảnh vừa tính toán.
- **Zero padding:** là việc thêm các giá trị 0 ở xung quanh biên ảnh để tính toán thêm đặc trưng ở các vùng biên

b) Lớp pooling

Tác dụng của lớp pooling là giảm kích thước đầu vào, giảm độ phức tạp tính toán và giúp kiểm soát hiện tượng overfitting. Có 2 phương pháp pooling được sử dụng phổ biến, đó là:

- **Max pooling:** thay thế giá trị vùng ảnh bởi giá trị pixel lớn nhất trong nó. Ý nghĩa là giúp giữ lại những vùng thông tin lớn nhất đại diện cho vùng ảnh.
- **Average Pooling:** thay thế giá trị vùng ảnh bởi giá trị trung bình các pixel trong vùng. Ý nghĩa là giúp lấy thông tin tổng thể để đại diện cho vùng ảnh.

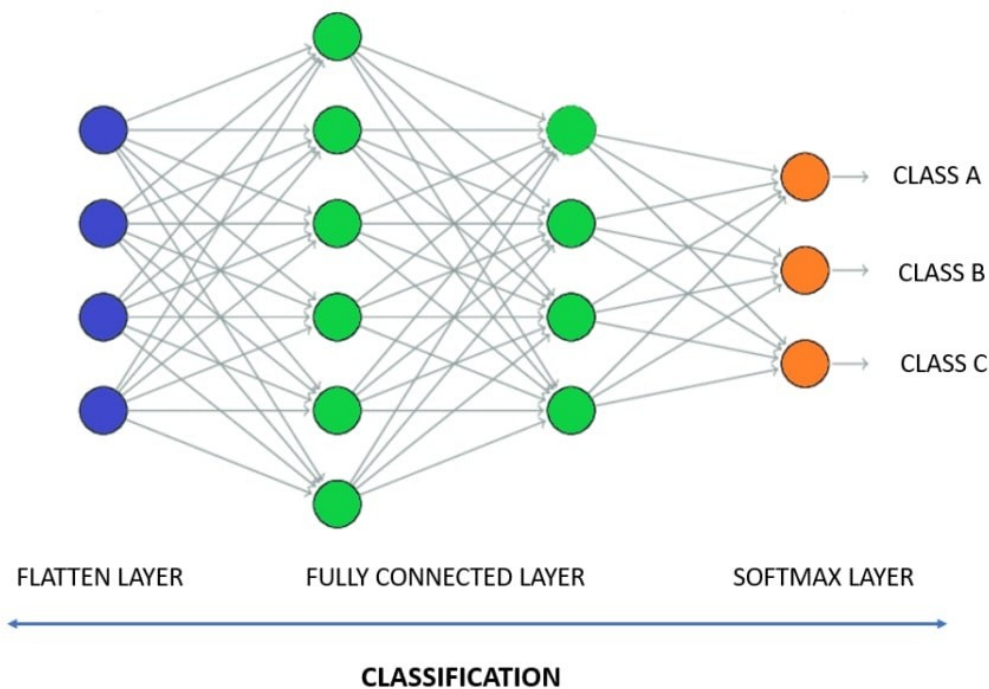


Hình 6 Minh họa việc tính toán trên lớp Max Pooling [17]

Thông thường các lớp pooling có kích thước là 2 và sử dụng stride=2. Nếu 2 giá trị này lớn nó sẽ làm phá vỡ cấu trúc ảnh và mất mát thông tin nghiêm trọng. Hình 6 minh họa một bước max pooling.

c) Lớp kết nối toàn bộ

Tại lớp này, mỗi một nơ-ron của 1 lớp sẽ liên kết tới mọi nơ-ron của lớp khác. Đầu vào của các lớp này là mảng 1 chiều đã được dàn phẳng của các đặc trưng ảnh. Tại các lớp cuối sẽ sử dụng hàm softmax để phân loại đối tượng dựa trên những đặc trưng đã tính toán trước đó.

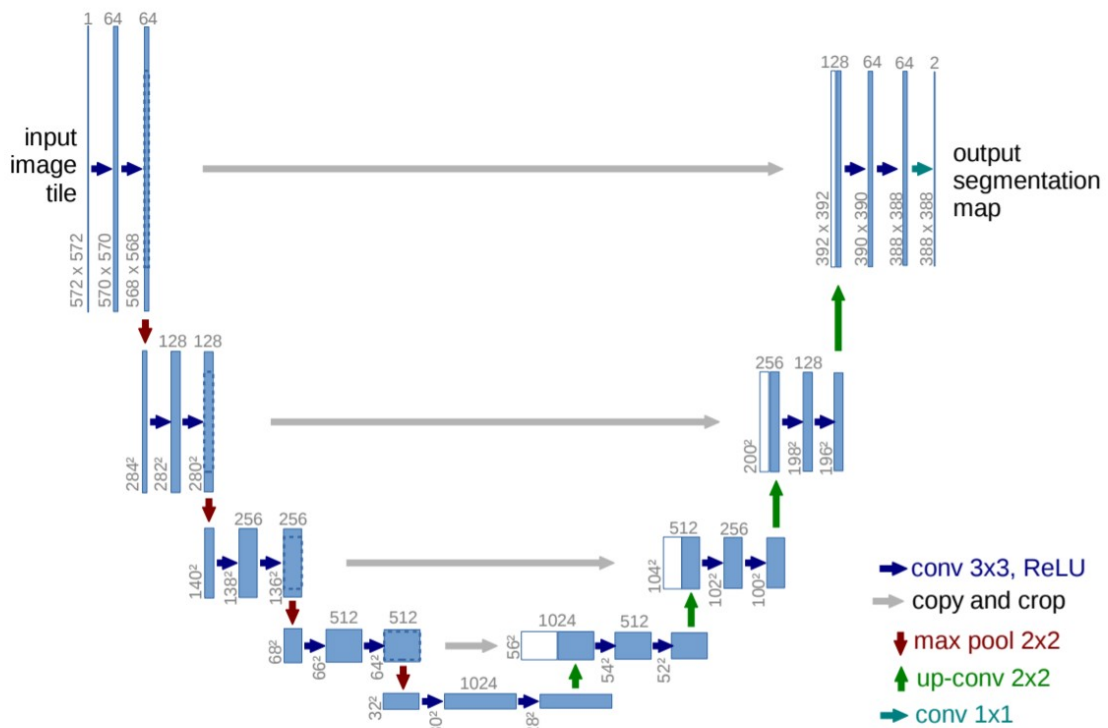


Hình 7 Minh họa lớp kết nối toàn bộ [18]

1.4.2 U-Net

Unet là một kiến trúc được phát triển bởi Olaf Ronneberger và các cộng sự phát triển nhằm phân vùng các cấu trúc nơ-ron thần kinh trong não người.

Hình 8 là kiến trúc mô hình Unet. Mỗi một thanh chữ nhật màu xanh là một feature map đa kênh. Kích thước chiều rộng x chiều dài được kí hiệu góc trái bên dưới của thanh chữ nhật và số lượng channels được kí hiệu trên đỉnh của feature map. Các thanh chữ nhật màu trắng bên nhánh phải của hình chữ U được sao chép từ nhánh bên trái và concatenate vào nhánh bên phải. Mỗi một mũi tên có màu sắc khác nhau tương ứng với một phép biến đổi khác nhau như chúng ta có thể thấy trong mô tả của mạng.



Hình 8 Kiến trúc mô hình U-Net [19]

Kiến trúc mạng Unet bao gồm 2 phần là **phần thu hẹp (contraction)** ở bên trái và **phần mở rộng (expansion)** ở bên phải. Mỗi phần sẽ thực hiện một nhiệm vụ riêng như sau:

- **Phần thu hẹp:** Làm nhiệm vụ trích lọc đặc trưng để tìm ra bối cảnh của hình ảnh. Vai trò của phần thu hẹp tương tự như một Encoder. Một mạng CNN sẽ đóng vai trò trích lọc đặc trưng. Lý do nhánh được gọi là thu hẹp vì kích thước dài và rộng của các lớp giảm dần. Từ input kích thước 572x572 chỉ còn 32x32. Đồng thời độ sâu cũng tăng dần từ 1 lên 512.
- **Phần mở rộng:** Gồm các layer đối xứng tương ứng với các layer của nhánh thu hẹp. Quá trình Upsampling được áp dụng giúp cho kích thước layer tăng dần lên. Sau cùng ta thu được một ảnh mask đánh dấu nhãn dự báo của từng pixel.

Đặc trưng riêng trong cấu trúc của Unet đó là áp dụng kết nối tắt đối xứng giữa layer bên trái với layer bên phải.

Mặc dù có độ chính xác khá cao nhưng U-Net có tốc độ thấp. Với kiến trúc U-Net cho input 572x572 như bài báo gốc có tốc độ là 5 fps. Do đó nó không phù hợp để áp dụng vào các tác vụ yêu cầu thời gian thực như xe tự hành. Tuy nhiên, Unet lại thường được sử dụng khá phổ biến trong các tác vụ không đòi hỏi thời gian thực vì accuracy của nó cũng không tồi và kiến trúc dễ cài đặt.

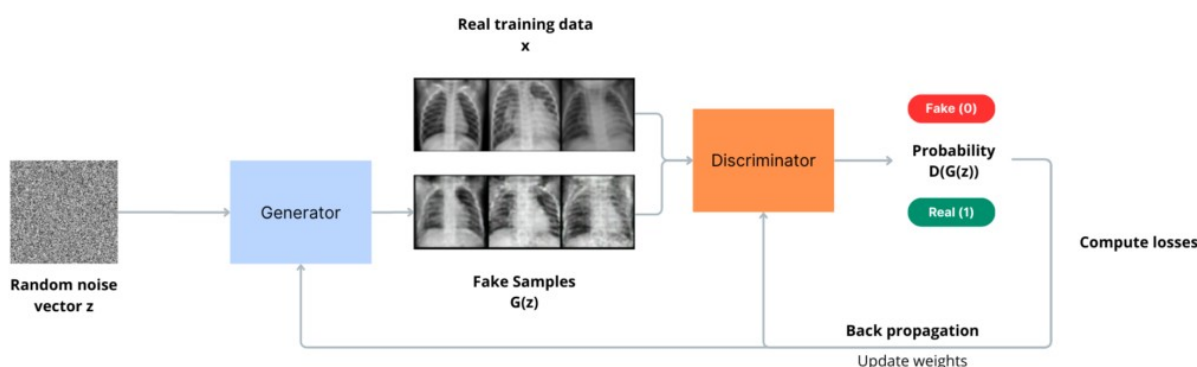
1.4.3 GAN

GAN viết tắt cho Generative Adversarial Networks. Generative giống như ở trên, Network có nghĩa là mạng (mô hình), còn Adversarial là đối nghịch. Tên gọi như vậy là do GAN được cấu thành từ 2 mạng gọi là Generator và Discriminator, luôn đối nghịch đầu với nhau trong quá trình train mạng GAN.

Một Mạng GAN được tạo thành từ hai thành phần chính: **máy tạo (Generator) G** và **máy phân biệt (Discriminator) D**.

Cả hai thành phần đều là mạng nơ-ron nhân tạo, nhưng vai trò của chúng khác nhau:

- Nhiệm vụ của **G** là tái tạo phân bố dữ liệu của tập dữ liệu huấn luyện x , để tạo ra các mẫu tổng hợp theo cùng phân bố dữ liệu đó. Dữ liệu này thường là hình ảnh, nhưng cũng có thể là âm thanh hoặc văn bản.
- Ngược lại, **D** giống như một "thẩm phán" sẽ ước tính xem một mẫu x có thật hay giả (đã được tạo ra). Thực chất, nó là một bộ phân loại sẽ nói xem một mẫu có xuất phát từ phân bố dữ liệu thực hay do máy tạo.



Hình 9 Minh họa quá trình huấn luyện GAN [20]

Trong hình 9 là quá trình huấn luyện mạng GAN, máy tạo bắt đầu với một vector nhiễu ngẫu nhiên (z) làm đầu vào và tạo ra các mẫu $G(z)$.

Khi quá trình huấn luyện diễn ra, máy tạo sẽ tinh chỉnh đầu ra của nó, làm cho dữ liệu được tạo $G(z)$ ngày càng giống với dữ liệu thật hơn. Mục tiêu của máy tạo là đánh lừa máy phân biệt để nó phân loại các mẫu được tạo ra là thật.

Trong khi đó, máy phân biệt được cung cấp cả các mẫu thật từ dữ liệu huấn luyện và các mẫu giả từ máy tạo. Khi nó học cách phân biệt giữa hai loại mẫu, nó sẽ cung cấp phản hồi cho máy tạo về chất lượng của các mẫu được tạo ra. Đây là lý do tại sao thuật ngữ "đối kháng" (adversarial) được sử dụng ở đây.

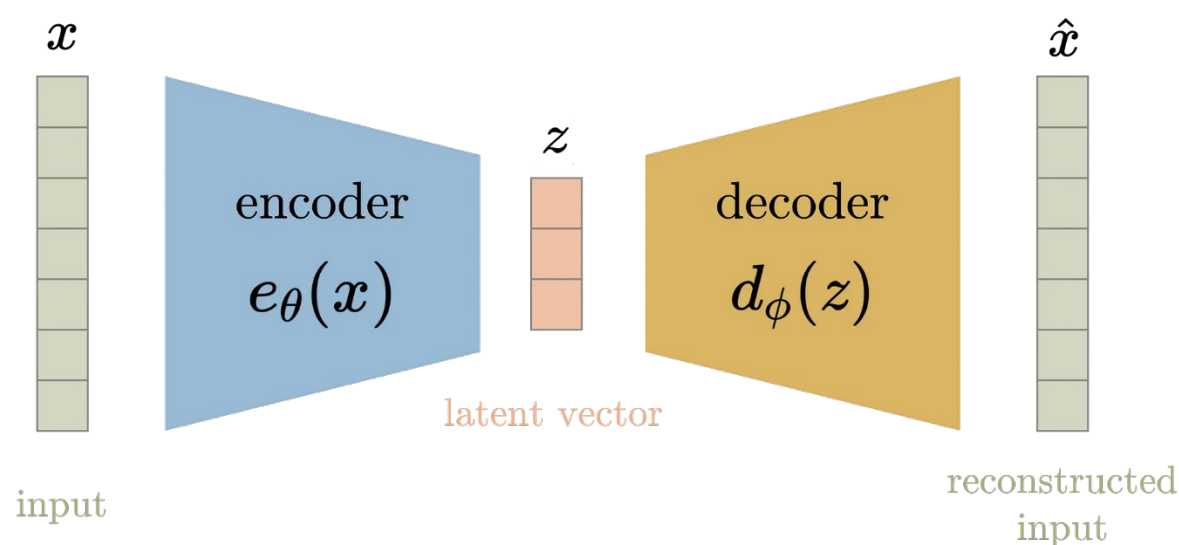
1.4.4 AE và VAEs

a) Autoencoder (AE)

Mạng tự mã hóa (AE) được sử dụng để học các biểu diễn hiệu quả của dữ liệu chưa được gán nhãn. Mạng tự mã hóa bao gồm hai phần, một bộ mã hóa (encoder) và một bộ giải mã (decoder).

Bộ mã hóa nén dữ liệu từ không gian chiều cao hơn thành không gian chiều thấp hơn (còn gọi là không gian tiềm ẩn), trong khi bộ giải mã thực hiện ngược lại, tức là chuyển đổi không gian tiềm ẩn trở lại không gian chiều cao hơn.

Bộ giải mã được sử dụng để đảm bảo rằng không gian tiềm ẩn có thể nắm bắt hầu hết thông tin từ không gian tập dữ liệu, bằng cách buộc đầu ra của nó giống những gì đã được đưa vào bộ giải mã.



$$loss = \|x - \hat{x}\|_2 = \|x - d_\phi(z)\|_2 = \|x - d_\phi(e_\theta(x))\|_2$$

Hình 10 Kiến trúc và hàm mất mát của Autoencoder [21]

Trong quá trình huấn luyện, dữ liệu đầu vào x được đưa vào hàm bộ mã hóa $e_\theta(x)$. Dữ liệu đầu vào được truyền qua một loạt các lớp (được tham số hóa bởi biến θ) nhằm giảm kích thước của nó để đạt được một vectơ tiềm ẩn được nén z . Số lượng lớp, loại và kích thước của các lớp cũng như kích thước không gian tiềm ẩn là các tham số do người dùng điều chỉnh. Việc nén được thực hiện nếu kích thước của không gian tiềm ẩn nhỏ hơn kích thước của không gian đầu vào, về cơ bản đây là đang loại bỏ các thuộc tính dư thừa. Kiến trúc của AE được mô tả trong hình 10.

Bộ giải mã $d_\phi(z)$ thường (nhưng không nhất thiết) bao gồm các lớp gần giống với các lớp được sử dụng trong bộ mã hóa nhưng theo thứ tự ngược lại. Một lớp gần giống với một lớp khác là lớp có thể được sử dụng để hoàn tác các phép toán (ở một mức độ nào đó) của lớp ban đầu.

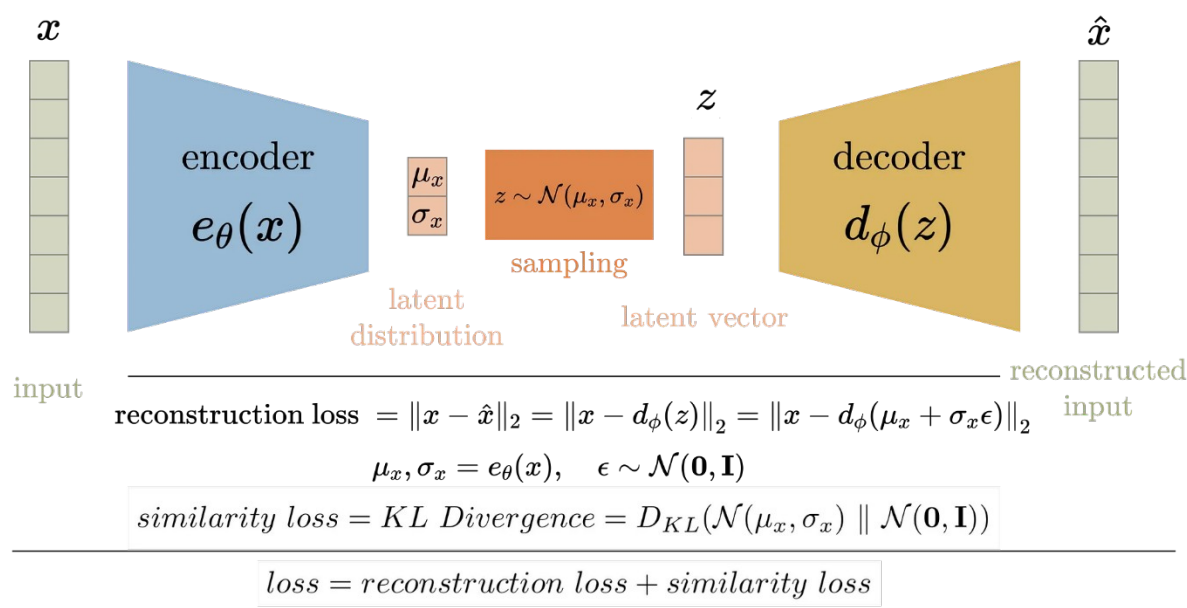
Toàn bộ kiến trúc bộ mã hóa - bộ giải mã được huấn luyện chung trên hàm mất mát (loss function) nhằm khuyến khích việc tái tạo dữ liệu đầu vào ở đầu ra. Do đó,

hàm mất mát được sử dụng là sai số bình phương trung bình (mean squared error) giữa đầu vào của bộ mã hóa và đầu ra của bộ giải mã.

Đối với các đầu vào hợp lệ, mạng tự mã hóa (AE) có thể nén chúng thành ít bit hơn, về cơ bản là loại bỏ sự dư thừa. Do tính chất của mạng nên chúng ta chỉ có thể tận dụng để tạo bộ mã hóa, rất khó để dùng bộ giải mã tạo ra dữ liệu mới.

b) Variational AutoEncoders

Đột phá quan trọng của VAEs là một mô hình xác suất mới giúp tạo ra nội dung mới tương tự nhưng khác biệt so với nội dung gốc. Trong VAEs, lớp trung gian cung cấp cách biểu diễn dữ liệu trong một trường xác suất, cho phép lớp lưu trữ nhiều dạng hơn với độ chính xác cao hơn.



Hình 11 Kiến trúc và hàm mất mát của Variational AutoEncoders [21]

Trong quá trình huấn luyện, dữ liệu đầu vào \mathbf{x} được đưa vào hàm bộ mã hóa $e_\theta(\mathbf{x})$. Giống như AE, đầu vào được truyền qua một loạt các lớp (được tham số hóa bởi biến θ) nhằm giảm kích thước của nó để đạt được một vector tiềm ẩn được nén \mathbf{z} . Tuy nhiên, vector tiềm ẩn không phải là đầu ra của bộ mã hóa. Thay vào đó, bộ mã hóa xuất ra giá trị trung bình (mean) và độ lệch chuẩn (standard deviation) cho mỗi biến tiềm ẩn. Vector tiềm ẩn sau đó được lấy mẫu từ giá trị trung bình và độ lệch chuẩn này để đưa vào bộ giải mã nhằm tái tạo dữ liệu đầu vào. Bộ giải mã trong VAE hoạt động tương tự như bộ giải mã trong AE. Kiến trúc của VAEs được mô tả trong hình 11.

Điểm khác biệt chính giữa VAE và AE nằm ở cách xử lý không gian tiềm ẩn:

- Trong AE, vector tiềm ẩn được tạo ra trực tiếp bởi bộ mã hóa.
- Trong VAE, bộ mã hóa chỉ xuất ra các tham số thống kê (mean và standard deviation) của phân phối tiềm ẩn. Vector tiềm ẩn sau đó được lấy mẫu từ phân phối này.

Việc lấy mẫu vector tiềm ẩn từ phân phối xác suất giúp VAE có được những ưu điểm sau:

- **Tính đa dạng:** VAE có thể tạo ra nhiều đầu ra khác nhau từ cùng một dữ liệu đầu vào, bằng cách lấy mẫu các vectơ tiềm ẩn khác nhau từ phân phối xác suất.
- **Tránh overfitting:** Việc sử dụng phân phối xác suất ngăn ngừa hiện tượng overfitting và giúp mô hình học được các đặc trưng có ý nghĩa hơn của dữ liệu.

Hàm mất mát huấn luyện của VAEs được định nghĩa là tổng của hai thành phần: mất mát tái tạo (reconstruction loss) và mất mát tương đồng (similarity loss).

- **Reconstruction loss:** giống như trong AE

- **Similarity loss:** là phân kỳ Kullback-Leibler (KL divergence) giữa phân phối của không gian tiềm ẩn và phân phối Gaussian chuẩn (có trung bình bằng 0 và phương sai bằng 1). Mục tiêu của mất mát này là khuyến khích không gian tiềm ẩn có dạng Gaussian chuẩn, từ đó giúp cho không gian tiềm ẩn có tính liên tục và dễ dàng tạo ra các mẫu mới từ nó.

VAEs thường được ưa chuộng hơn AE trong các nhiệm vụ cần tạo ra dữ liệu mới hoặc khám phá các cấu trúc tiềm ẩn trong dữ liệu.

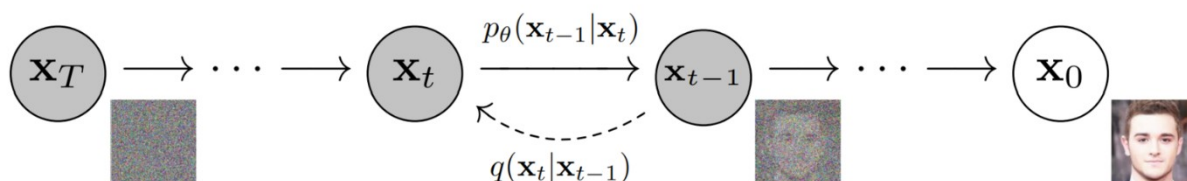
1.4.5 Diffusion

Các mô hình Diffusion trong học sâu được giới thiệu lần đầu bởi Sohl-Dickstein và đồng nghiệp trong bài báo quan trọng năm 2015 có tựa đề “Deep Unsupervised Learning using Nonequilibrium Thermodynamics.”

Nhưng vào năm 2019, Song và cộng sự đã xuất bản một bài báo có tựa đề “Generative Modeling by Estimating Gradients of the Data Distribution,” sử dụng cùng nguyên lý nhưng tiếp cận khác nhau. Năm 2020, Ho và đồng nghiệp đã xuất bản bài báo với tựa đề nổi tiếng hiện nay là “Denoising Diffusion Probabilistic Models” (viết tắt là DDPM).

Sau năm 2020, nghiên cứu về các mô hình diffusion phát triển mạnh mẽ. Đã có nhiều tiến bộ đáng kể trong việc huấn luyện và cải tiến mô hình sinh ra dữ liệu trong khoảng thời gian tương đối ngắn.

Cấu trúc (phân phối) của hình ảnh gốc được dần phá hủy bằng cách thêm nhiễu và sau đó sử dụng một mô hình mạng neural để tái tạo lại hình ảnh, tức là loại bỏ nhiễu ở mỗi bước. Thực hiện điều này đủ lần và với dữ liệu tốt, mô hình cuối cùng sẽ học được ước lượng về phân phối dữ liệu ẩn (gốc). Sau đó, chúng ta có thể đơn giản chỉ bắt đầu chỉ với nhiễu và sử dụng mạng neural đã được huấn luyện để tạo ra một hình ảnh mới đại diện cho tập dữ liệu huấn luyện gốc.



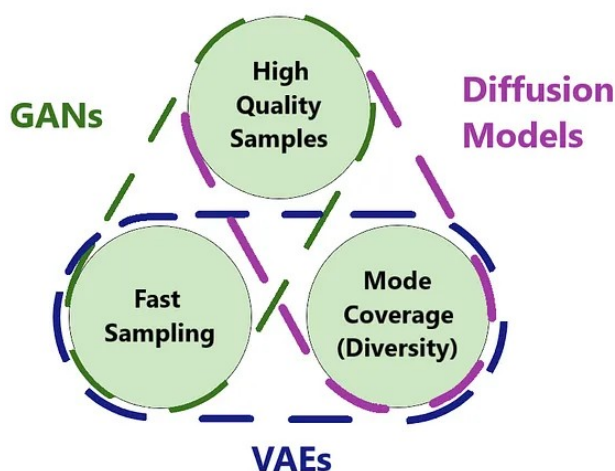
Hình 12 Minh họa quá trình diffusion [1].

Như hình 12, chúng ta sẽ có 2 bước để huấn luyện mô hình Diffusion:

- Forward Diffusion:
 - Ảnh gốc ban đầu (\mathbf{x}_0) bị làm nhiễu dần theo từng bước (tạo thành một chuỗi Markov chain) bằng việc thêm dần nhiễu Gaussian.
 - Quá trình này sẽ hoàn tất khi thực hiện T bước.
 - Ảnh ở bước t được tạo bằng cách: $\mathbf{x}_{t-1} + \epsilon_{t-1}$ (**nh nhiễu**) $\rightarrow \mathbf{x}_t$
 - Ở giai đoạn này không sử dụng mô hình nào.
 - Ở cuối của quá trình forward diffusion là \mathbf{x}_T , do việc thêm nhiễu lặp đi lặp lại chúng ta thu được một hình ảnh nhiễu hoàn toàn.
- Backward/Reverse diffusion:
 - Ở giai đoạn này, chúng ta lặp ngược lại quá trình Forward Diffusion. Nhiệm vụ là loại bỏ nhiễu được thêm vào trong quá trình Forward Diffusion, một lần nữa theo cách lặp đi lặp lại (một chuỗi Markov). Điều này được thực hiện bằng cách sử dụng một mô hình mạng neural.
 - Mô hình có nhiệm vụ như sau: cho một bước thời gian t và hình ảnh nhiễu \mathbf{x}_t , dự đoán nhiễu đã được thêm vào ảnh tại bước $t-1$.
 - $\mathbf{x}_t \rightarrow$ **Mô hình** $\rightarrow \epsilon$ (**nh nhiễu dự đoán**). Mô hình dự đoán (tính xấp xỉ) nhiễu được thêm vào tại bước forward \mathbf{x}_{t-1} .

1.4.6 So sánh GAN, VAEs và Diffusion

Các phần trên đồ án đã nêu chi tiết cấu trúc của 3 mô hình tạo ảnh tốt nhất hiện nay, sau đây sẽ là điểm mạnh và điểm yếu của từng mô hình được đem ra so sánh với nhau.



Hình 13 Điểm mạnh, điểm yếu của GANs, VAEs, Diffusion Models [22]

Để đánh giá các mô hình, ta sẽ đánh giá trên 3 tiêu chí tốc độ, chất lượng và sự đa dạng của mẫu. Hình 13 thể hiện điểm mạnh và yếu của các mô hình. Sau đây là điểm mạnh yếu của từng mô hình:

- **GANs:** Tốc độ tạo mẫu nhanh, mẫu tạo ra có chất lượng cao nhưng thiếu đi sự đa dạng và mô hình này rất khó huấn luyện.
- **VAEs:** Tốc độ tạo mẫu nhanh, mẫu tạo ra đa dạng nhưng chất lượng mẫu tạo ra lại có chất lượng kém hơn do cơ chế mã hóa-giải mã bị mất thông tin.
- **Diffusion Models:** Chất lượng mẫu tạo ra chất lượng và đa dạng nhưng tốc độ bị chậm cho cơ chế denoise từng bước một.

1.4.7 CLIP và M-CLIP

1) CLIP

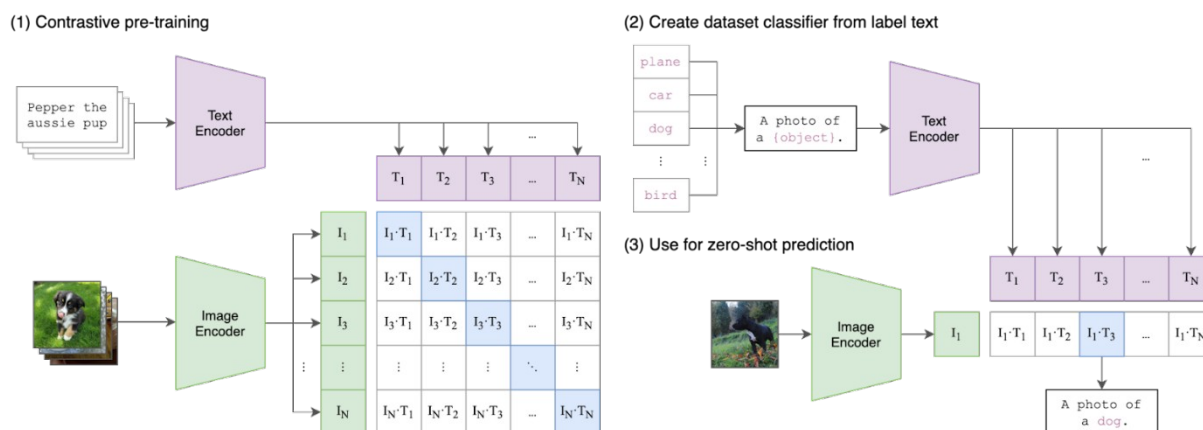
CLIP (Contrastive Language-Image Pre-Training) là một mô hình máy học sâu được phát triển bởi OpenAI, kết hợp cả xử lý ngôn ngữ tự nhiên và xử lý hình ảnh. Mô hình này được huấn luyện trên cặp dữ liệu (hình ảnh, văn bản) đa dạng để hiểu cách ánh xạ giữa hai loại dữ liệu này.

Khác với nhiều mô hình trước đó chỉ tập trung vào một nhiệm vụ cụ thể như phân loại hình ảnh, CLIP được huấn luyện để hiểu mối quan hệ giữa hình ảnh và văn bản. Nó có khả năng dự đoán văn bản phù hợp nhất cho một hình ảnh cho trước, và ngược lại, chỉ bằng cách sử dụng thông tin đã học từ việc huấn luyện trên dữ liệu (hình ảnh, văn bản).

Một điểm nổi bật của CLIP là khả năng zero-shot learning, cho phép nó dự đoán văn bản hoặc hình ảnh mà nó chưa từng thấy trong quá trình huấn luyện, dựa trên kiến thức đã học. Điều này giúp CLIP áp dụng kiến thức từ ngữ cảnh để hiểu hình ảnh và ngược lại, mà không cần có dữ liệu huấn luyện cụ thể cho từng nhiệm vụ.

Sau khi có được pre-train bằng cách dùng các cặp ảnh và văn bản ở bước 1 trong hình 14. Tiếp đó chúng ta tạo bộ nhãn bằng văn bản ở bước 2. Cuối cùng bước 3, tính độ tương đồng ảnh của ảnh với nhãn đã tạo để tìm ra nhãn cho ảnh.

CLIP có hiệu suất tương đương với ResNet50 gốc trên ImageNet "zero-shot" mà không sử dụng bất kỳ dữ liệu gốc nào trong 1.28 triệu dữ liệu đã được gán nhãn.



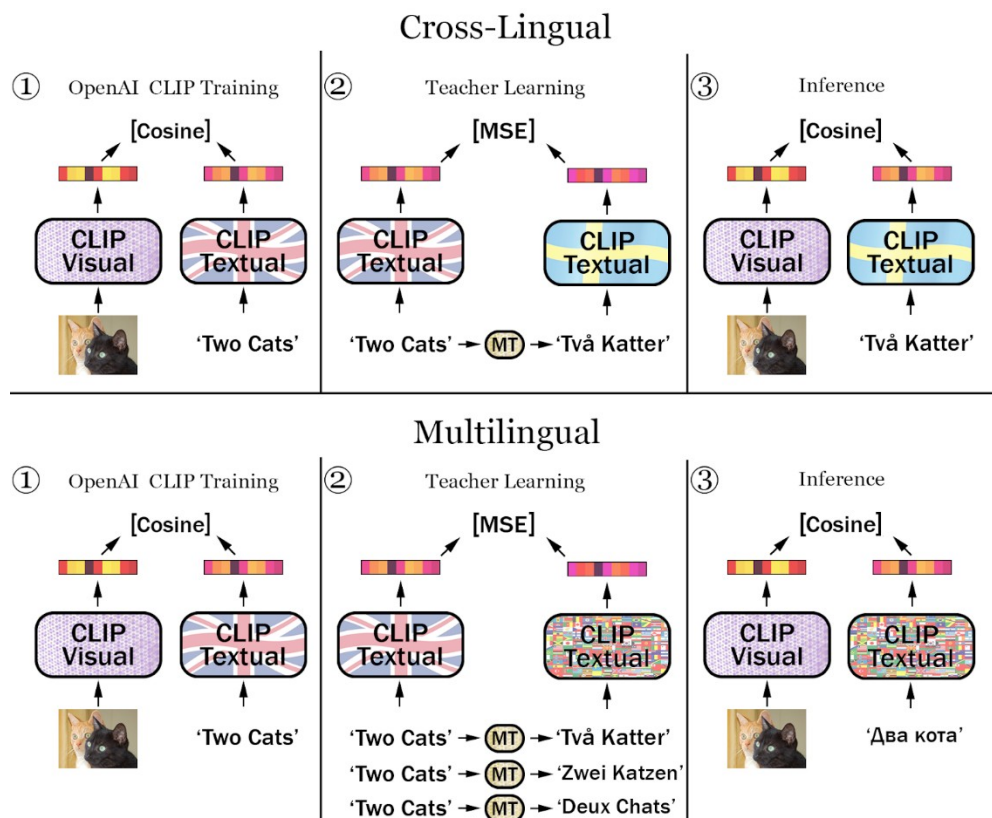
Hình 14 Zero-shot với mô hình CLIP [23]

CLIP có 2 thành phần là Text Encoder và Image Encoder:

- Text Encoder dùng Transformer để mã hóa văn bản
- Image Encoder thì có nhiều phiên bản như EfficientNet, ResNET, Vision Transformers.

2) M-CLIP

CLIP hiện nay có nhiều pretrain cho tiếng Anh nhưng hiện tại tiếng Việt chưa có riêng pretrain nào. Vì vậy tôi đề xuất sử dụng Multilingual-CLIP (M-CLIP) là một mô hình tương tự CLIP đã được huấn luyện với 100 ngôn ngữ khác nhau trong đó có tiếng Việt.



Hình 15 Quá trình huấn luyện M-CLIP [24]

Việc huấn luyện Multilingual-CLIP dùng phương pháp học theo thầy, CLIP của OpenAI sẽ đóng vai trò như là người thầy. Hình 15 mô tả quá trình huấn luyện như sau:

- Chuẩn bị một tập các cặp văn bản (tiếng Anh, tiếng Việt)
- Tính toán CLIP-Text embedding cho tiếng Anh
- Mô hình Multilingual-CLIP sẽ học cách để biểu diễn văn bản tiếng Việt như là embedding của văn bản tiếng Anh.

Như đã nêu quá trình học không cần dùng tới ảnh mà chỉ cần văn bản, điều này sẽ tiết kiệm chi phí và thời gian tính toán, giải quyết vấn đề thiếu dữ liệu để huấn luyện.

1.4.8 Vision Transformers

Vào năm 2022, Vision Transformer (ViT) nổi lên như một giải pháp thay thế cạnh tranh so với các mạng thần kinh tích chập (Convolutional Neural Network, gọi tắt là CNN) vốn đang là ứng dụng tiên tiến trong thị giác máy tính, và được sử dụng rộng rãi trong các tác vụ nhận dạng hình ảnh khác nhau. Các mô hình ViT được đánh giá là vượt trội hơn so với CNN gần 4 lần về hiệu quả tính toán và độ chính xác.

Mô hình Vision Transformer (ViT) đã được giới thiệu trong một bài báo nghiên cứu được xuất bản dưới dạng báo cáo hội nghị tại ICLR 2021, có tiêu đề “An Image is Worth 16*16 Words: Transformers for Image Recognition in Scale”. Nó được phát triển và xuất bản bởi Neil Houlsby, Alexey Dosovitskiy, và 10 tác giả khác của Google Research Brain Team.

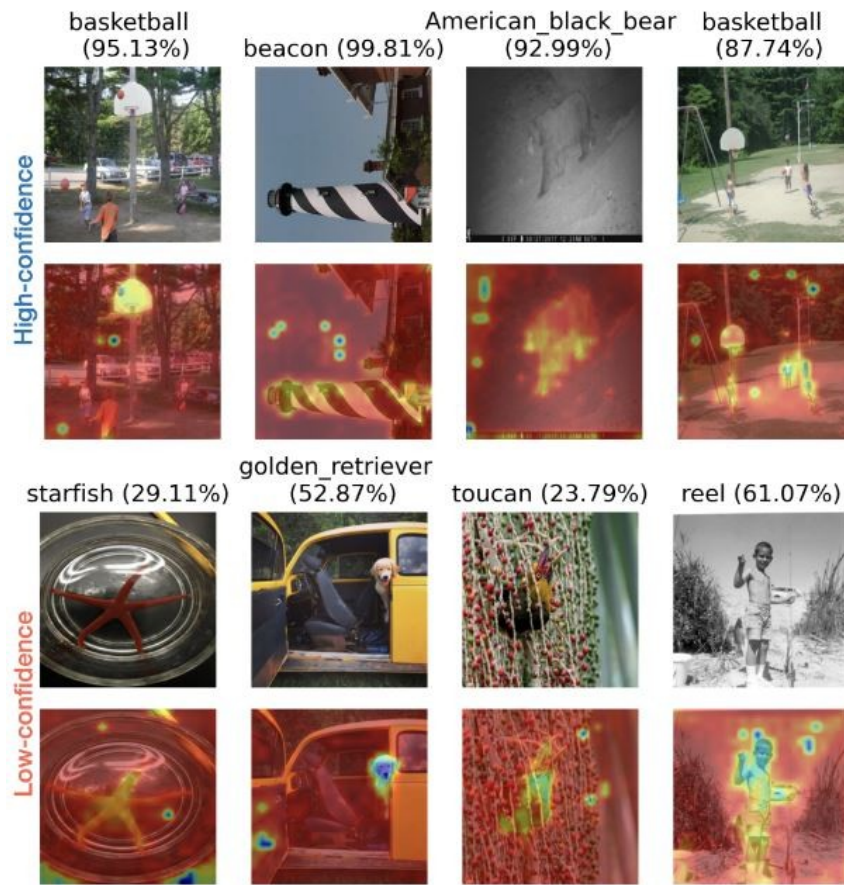
CNN sử dụng mảng pixel, trong khi ViT chia hình ảnh thành các tokens trực quan. Transformer trực quan chia hình ảnh thành các mảng hình ảnh có kích thước cố định, mã hóa từng mảng theo thứ tự làm đầu vào cho Transformer encoder. Hơn nữa, các mô hình ViT vượt trội hơn CNN gần bốn lần về hiệu quả tính toán và độ chính xác.

Lớp Self-attention trong ViT có khả năng tổng hợp thông tin trên toàn bộ hình ảnh. Mô hình này cũng học trên dữ liệu huấn luyện để mã hóa vị trí tương đối của các mảng ảnh nhằm tái tạo lại cấu trúc của hình ảnh.

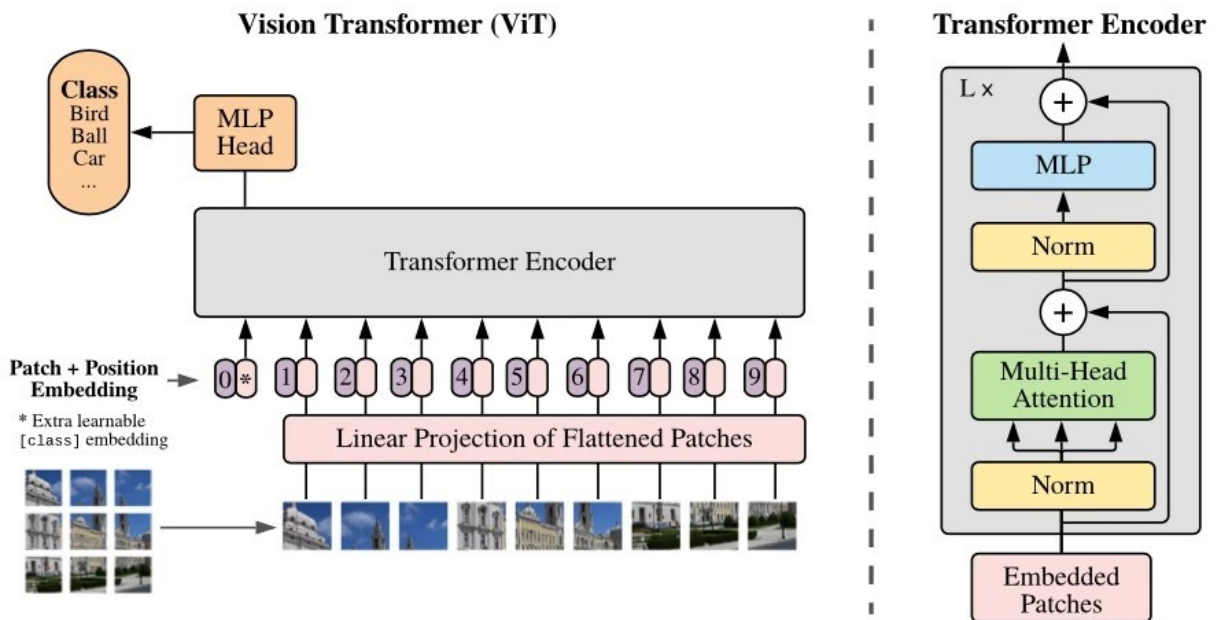
Transformer encoder bao gồm:

- Lớp Multi-Head Self Attention (MSP): Lớp này nối tất cả các kết quả đầu ra của Cơ chế Attention một cách tuyến tính theo đúng kích thước. Nhiều Attention head sẽ giúp huấn luyện những yếu tố phụ thuộc cục bộ và toàn bộ trong một hình ảnh.
- Lớp Multi-Layer Perceptrons (MLP): Lớp này chứa một hàm Gaussian Error Linear Unit hai lớp.
- Lớp thường: Lớp này được thêm vào trước mỗi khối, vì nó không bao gồm bất kỳ yếu tố phụ thuộc mới nào giữa các hình ảnh huấn luyện. Điều này giúp cải thiện thời gian đào tạo và hiệu suất tổng thể.

Cơ chế Attention, cụ thể hơn là Self-attention, là một trong những yếu tố thiết yếu của mô hình Transformer. Nó là một phép toán sơ khai được sử dụng để định lượng các tương tác thực thể theo từng cặp, giúp một mạng tìm hiểu cấu trúc phân cấp và sự liên kết hiện diện bên trong dữ liệu đầu vào. Cơ chế Attention đã được chứng minh là yếu tố then chốt để mạng đạt được độ chính xác cao hơn. Hình 16 minh họa cơ chế attention trên ảnh với nhãn, mô hình sẽ tập chung vào các phần quan trọng màu đậm hơn.



Hình 16 Minh họa cơ chế Attention của Vision Transformers [25]



Hình 17 Kiến trúc Vision Transformers [25]

Hình 17 là kiến trúc Vision Transformer (ViT), quá trình huấn luyện mô hình theo từng bước như sau:

- 1) Chia hình ảnh thành các mảng (patch) với kích thước từng mảng cố định
- 2) Làm phẳng các mảng hình ảnh
- 3) Tạo các feature embedding có chiều thấp hơn từ các mảng hình ảnh phẳng này
- 4) Bao gồm thứ tự các mảng
- 5) Chuỗi feature embedding được làm đầu vào cho transformer encoder
- 6) Thực hiện pre-train đối với mô hình ViT với các nhãn hình ảnh, sau đó được giám sát hoàn toàn trên một tập dữ liệu lớn
- 7) Tinh chỉnh model trên bộ dữ liệu riêng của từng bài toán

Trong khi kiến trúc ViT là một lựa chọn đầy hứa hẹn cho các tác vụ xử lý thị giác, hiệu suất của ViT vẫn kém hơn so với các giải pháp thay thế CNN có kích thước tương tự (chẳng hạn như ResNet) khi được huấn luyện từ đầu trên một tập dữ liệu cỡ trung như ImageNet.

1.5 Phạm vi nghiên cứu

Đồ án này sẽ tập chung vào việc tạo ra tập dữ liệu đầu tiên của bài toán tạo ảnh người bằng mô tả tiếng Việt. Kèm theo đó là những phương pháp, mô hình và các thực nghiệm đầu tiên cho tập dữ liệu này.

Tạo ảnh từ mô tả là một bài toán rộng và chứa nhiều tiềm năng trong tương lai cho các hệ thống thông minh nhưng để phát triển ở hiện tại vẫn gặp khá nhiều khó khăn. Do đó, tập dữ liệu sử dụng sẽ chỉ mô tả thông tin trang phục của người trong miền thời trang. Mong muốn đạt được kết quả tốt và đưa ra những đánh giá trực quan về bài toán.

1.6 Đóng góp của đồ án

Đồ án có một số đóng góp cơ bản sau:

- Giới thiệu bài toán tạo ảnh người bằng mô tả
- Xây dựng tập dữ liệu tiếng Việt cho bài toán
- Thực nghiệm kiến trúc mô hình sinh ảnh người bằng mô tả cho tập dữ liệu tiếng Việt. Từ đó đưa ra đánh giá, nhận xét về kết quả đạt được và đưa ra hướng dẫn cải tiến cho bài toán.

1.7 Kết luận chương

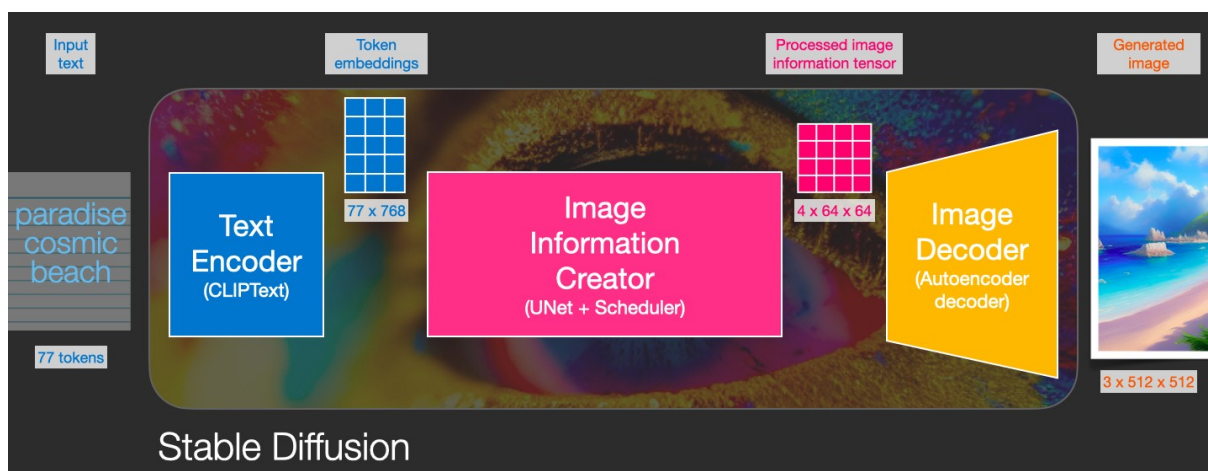
Chương 1 đã giới thiệu tổng quát về bài toán Tạo ảnh người bằng mô tả tiếng Việt, ứng dụng, một số nghiên cứu liên quan đến bài toán, kiến thức cơ bản, phạm vi nghiên cứu và các đóng góp của đồ án. Chương tiếp theo sẽ đi sâu vào mô tả những kiến trúc mô hình sử dụng cho bài toán này.

CHƯƠNG 2. TẠO ẢNH NGƯỜI TỪ MÔ TẢ TIẾNG VIỆT SỬ DỤNG MẠNG KHUẾCH TÁN ỔN ĐỊNH

Trong chương này, đồ án sẽ trình bày về các mô hình để giải quyết bài toán, bao gồm các phần sau: kiến trúc tổng quát của mô hình Stable Diffusion, mô hình UPGPT và U-VIT.

2.1 Mô hình Stable Diffusion

Phần này sẽ mô tả tổng quan về kiến trúc Stable Diffusion. Mô hình gồm 3 thành phần chính là bộ mã hóa văn bản (Text Encoder), Trình sáng tạo ảnh (Image Information Creator) và bộ giải mã hình ảnh (Image Decoder). Các thành phần được mô tả trong hình 18 sau đây:

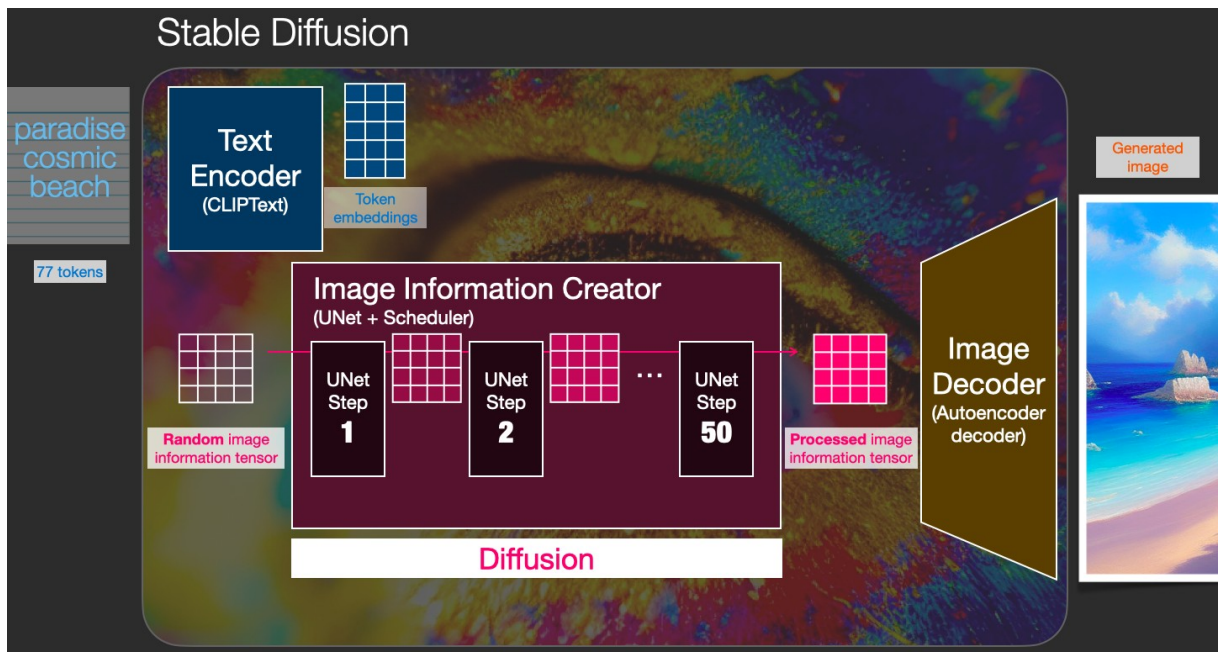


Hình 18 Các thành phần của Stable Diffusion [26]

- **TextEncoder:** ClipText cho việc mã hóa văn bản
 - Đầu vào: văn bản
 - Đầu ra: các vector embedding của 77 token, mỗi vector có chiều 768 dimensions
- **Image Information Creator:** U-Net + Scheduler để dần dần phục hồi lại thông tin trong latent space.
 - Đầu vào: các vector embedding của văn bản và một ma trận thông tin khởi tạo bởi nhiễu.
 - Đầu ra: Một ma trận thông tin đã được xử lý
- **Image Decoder:** VAE Decoder sẽ chuyển bức ảnh từ dạng ma trận thông tin thành dạng ảnh.
 - Đầu vào: Ma trận thông tin đã được xử lý (với số chiều là (4,64,64))
 - Đầu ra: Ảnh cuối cùng (có chiều (3,512,512) đại diện cho (màu, chiều rộng, chiều dài))

Quá trình sinh ảnh bằng Stable Diffusion sẽ được thực hiện như hình 19:

- 1) Sử dụng TextEncoder để lấy các vector embedding của văn bản đầu vào.
- 2) U-Net thực hiện loại bỏ nhiễu từng bước ma trận thông tin ngẫu nhiên khởi tạo ban đầu với sự hướng dẫn của vector embedding của văn bản để cho đầu ra là một ma trận thông tin đã được xử lý.
- 3) Ma trận thông tin này sẽ được xử lý trở lại thành ảnh có thể nhìn thấy.

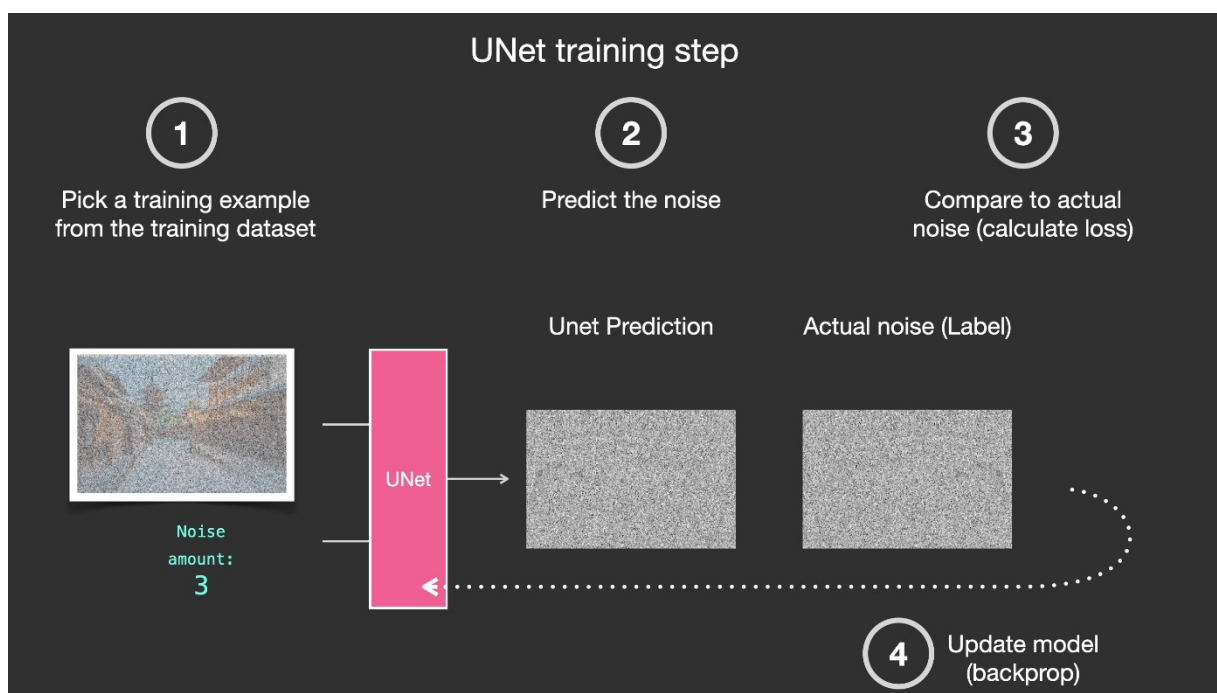


Hình 19 Quá trình sinh ảnh từ văn bản của Stable Diffusion [26]

Stable Diffusion sử dụng 3 thành phần, để huấn luyện 3 thành phần này cùng lúc là rất tốn kém. Sử dụng các pre-train có sẵn cho TextEncoder và Image Decoder, chỉ tập chung vào huấn luyện U-Net cho bước Diffusion sẽ giảm chi phí đi rất nhiều. Đây cũng là thành phần quan trọng nhất của Stable Diffusion.

Quá trình huấn luyện mô hình U-Net sẽ được thực hiện như hình 20:

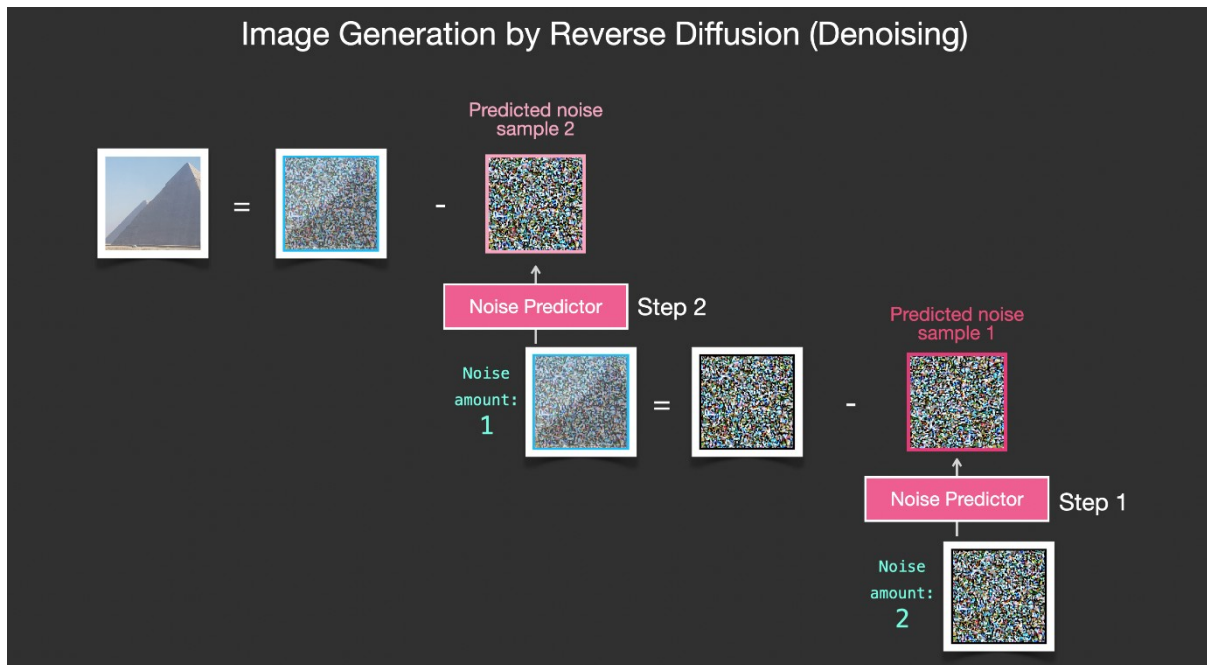
- 1) Lấy một mẫu dữ liệu từ tập dữ liệu gồm ảnh gốc với độ nhiễu.
- 2) Thông qua mô hình U-Net dự đoán nhiễu đã được thêm vào ảnh.
- 3) Tính mất mát giữa nhiễu dự đoán và nhiễu đã được thêm vào
- 4) Cập nhật mô hình U-Net



Hình 20 Quá trình huấn luyện U-Net [26]

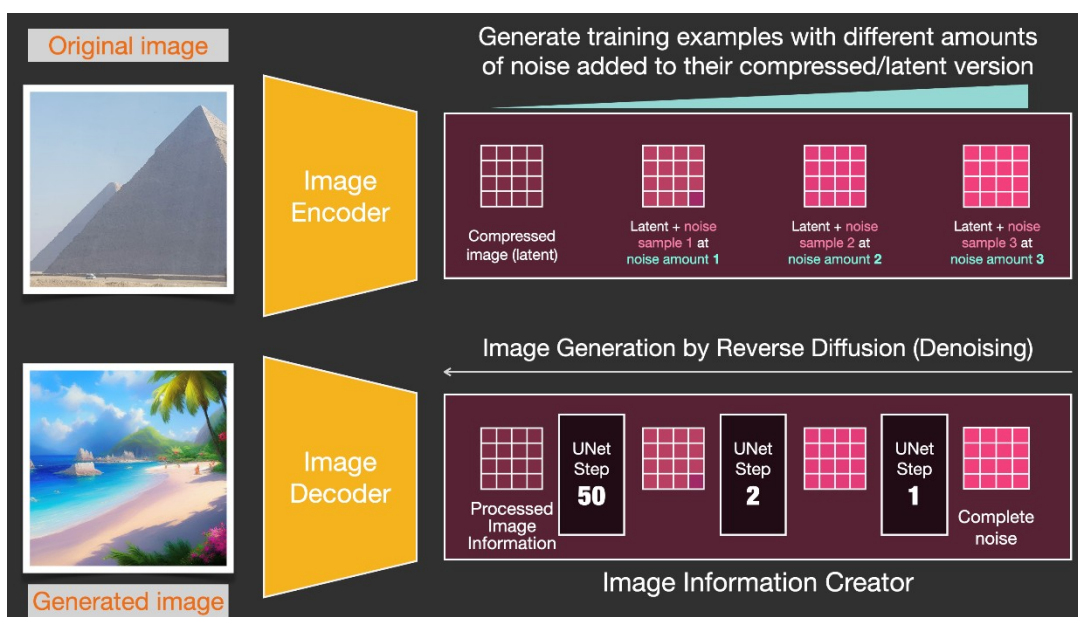
Quá trình quá trình khử nhiễu để tạo ảnh được thực hiện như hình 21:

- 1) Khởi tạo ảnh nhiễu ngẫu nhiên ban đầu
- 2) Dùng U-Net dự đoán nhiễu
- 3) Lấy ảnh đầu vào – nhiễu dự đoán thì ra được ảnh bớt nhiễu
- 4) Thực hiện lặp lại hữu hạn với số bước đã chọn để ra được ảnh.



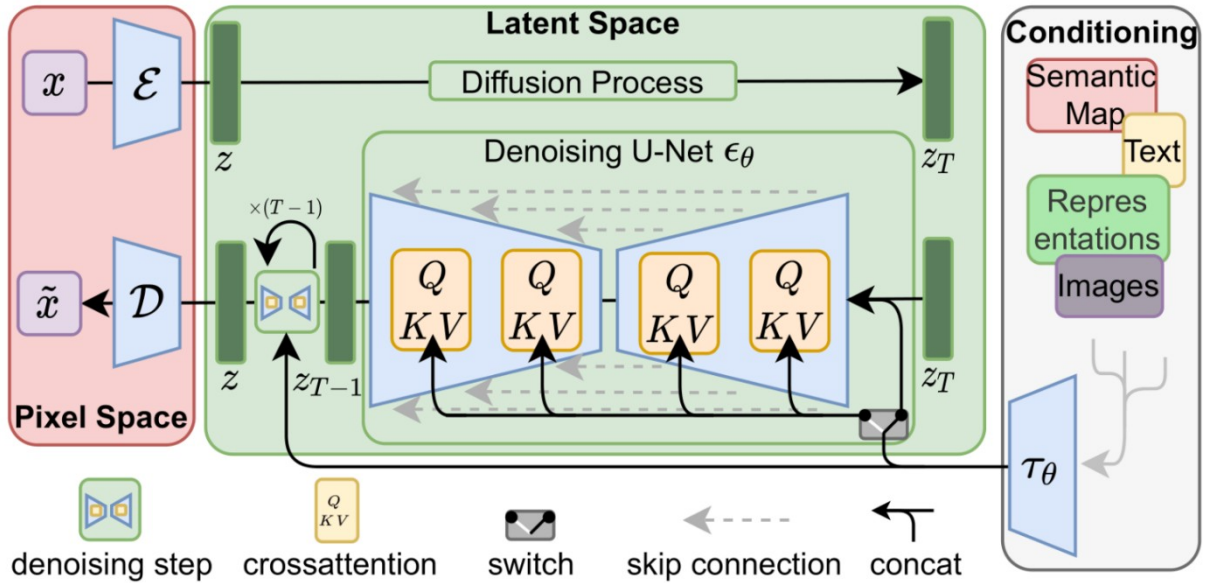
Hình 21 Quá trình khử nhiễu bằng U-Net [26]

Ở phần trên mô phỏng việc thực hiện diffusion với ví dụ trên ảnh, để tăng tốc độ Stable Diffusion đã thực hiện huấn luyện trên không gian vector ẩn, đầu vào U-Net là ảnh đã được cho qua Image Encoder. Rồi thực hiện khử nhiễu và dùng Image Decoder để chuyển lại về ảnh nhìn được như hình 22.



Hình 22 Diffusion trong latent space [26]

Mô hình khuếch tán đã nêu trên được gọi là mô hình khuếch tán tiềm ẩn (Latent diffusion model - LDM) hoạt động trong không gian tiềm ẩn thay vì không gian điểm ảnh, giúp giảm chi phí huấn luyện và tăng tốc độ suy luận.



Hình 23 Kiến trúc mô hình LDM/SD [27]

Trong hình 23, một encoder ϵ được dùng để nén ảnh đầu vào $x \in \mathbb{R}^{H \times W \times 3}$ thành vector ẩn 2D $z = \epsilon(x) \in \mathbb{R}^{h \times w \times c}$, tỉ lệ giảm $f = \frac{H}{h} = \frac{W}{w} = 2^m, m \in \mathbb{N}$. Sau đó một decoder $D(z)$ sẽ tái cấu trúc lại ảnh từ vector ẩn, $\tilde{x} = D(z)$.

Quá trình khử nhiễu được thực hiện trên vector ẩn z . Mô hình khử nhiễu là time-conditioned U-Net, được tinh chỉnh với cơ chế cross-attention để kiểm soát thông tin ảnh đầu ra. Với thiết kế này chúng ta có thể dùng nhiều hoặc kết hợp nhiều loại điều kiện khác nhau để làm đầu vào mô hình. Mỗi thông tin điều kiện sẽ có encoder riêng τ_θ để làm điều kiện đầu vào y cho cross-attention, $\tau_\theta \in \mathbb{R}^{M \times d_i}$:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V$$

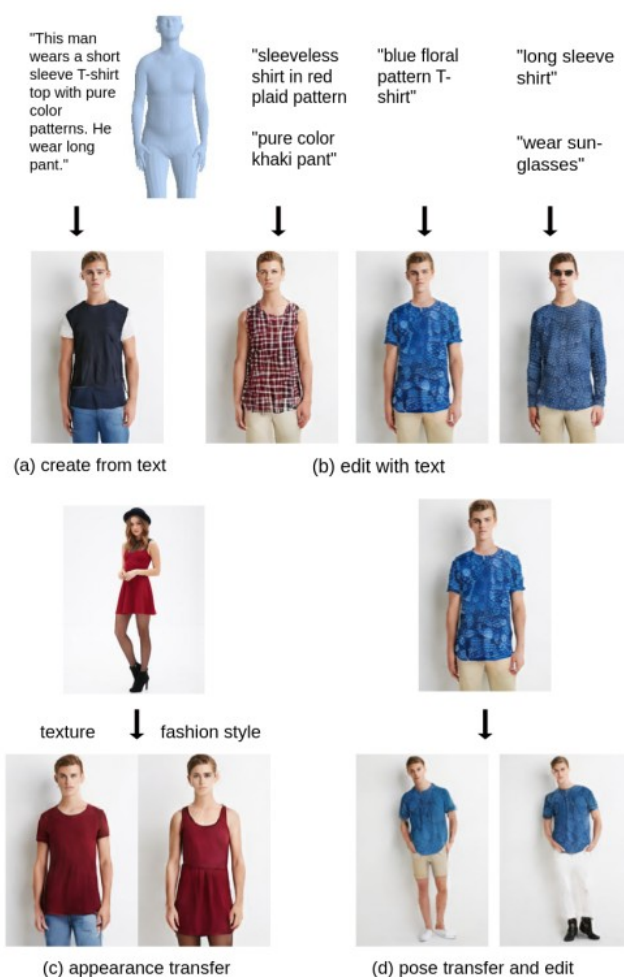
$$v\text{ới } Q = W_Q^{(i)} \cdot \varphi_i(z_i), K = W_K^{(i)} \cdot \tau_\theta(y), V = W_V^{(i)} \cdot \tau_\theta(y)$$

$$\text{và } W_Q^{(i)} \in \mathbb{R}^{d \times d_i}, W_K^{(i)} \in \mathbb{R}^{d \times d_i}, \varphi_i \in \mathbb{R}^{d \times d_i}, \tau_\theta(y) \in \mathbb{R}^{M \times d_i}$$

Trong phần này đồ án đã cung cấp thông tin về kiến trúc Stable Diffusion và cải tiến LDM. Đây là phương pháp đã đạt kết quả cao trong các bài toán Text2Image trên nhiều tập dữ liệu. Phần sau sẽ cung cấp thông tin về các phương pháp đã đạt kết quả cao trong bài toán sinh ảnh người. Các phương pháp ít nhiều đều có sự tinh chỉnh so với Stable Diffusion.

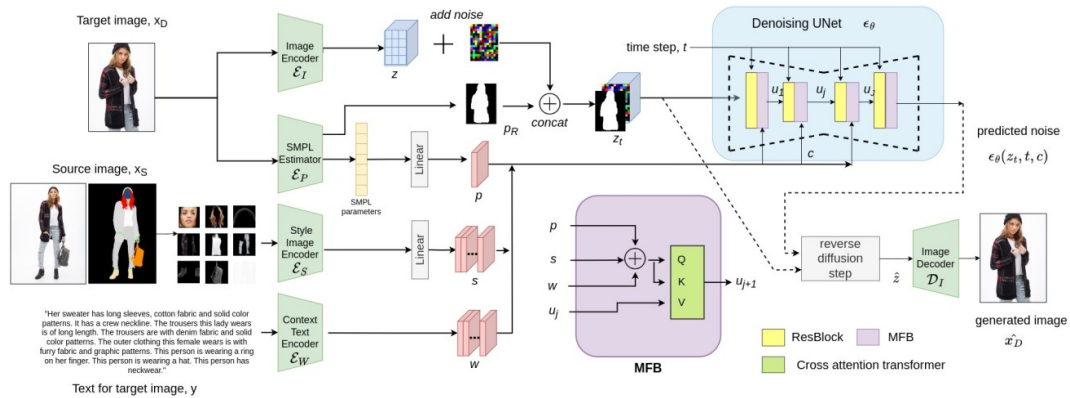
2.2 Mô hình UPGPT

Các mô hình sinh hình ảnh từ văn bản như Stable Diffusion đã được dùng để tạo ra các hình ảnh chất lượng của con người. Tuy nhiên, do tính ngẫu nhiên của quá trình tạo ra, người trong hình có vẻ khác nhau như tư thế, khuôn mặt và trang phục, mặc dù sử dụng cùng một văn bản mô tả. Sự không nhất quán về diện mạo khiến cho Text2Image không phù hợp để chuyển đổi tư thế. Tác giả của bài báo UPGPT [28] đã đề xuất ra một mô hình sử dụng kết hợp văn bản, dáng người và các thông tin kiểu dáng để giải quyết các bài toán tạo ảnh người, chuyển đổi tư thế và chỉnh sửa ảnh. Sử dụng trực tiếp thông tin 3D để làm đầu vào mô hình tạo ảnh đã cho ra kết quả cao tốt hơn hẳn so với các mô hình trước đó. Trong hình 24 cho thấy kết quả của mô hình với các bài toán tạo ảnh từ mô tả, chỉnh sửa ảnh bằng mô tả, chỉnh sửa quần áo, chỉnh sửa vị trí trong duy nhất một kiến trúc mô hình.



Hình 24 Các bài toán UPGPT có thể xử lý [28]

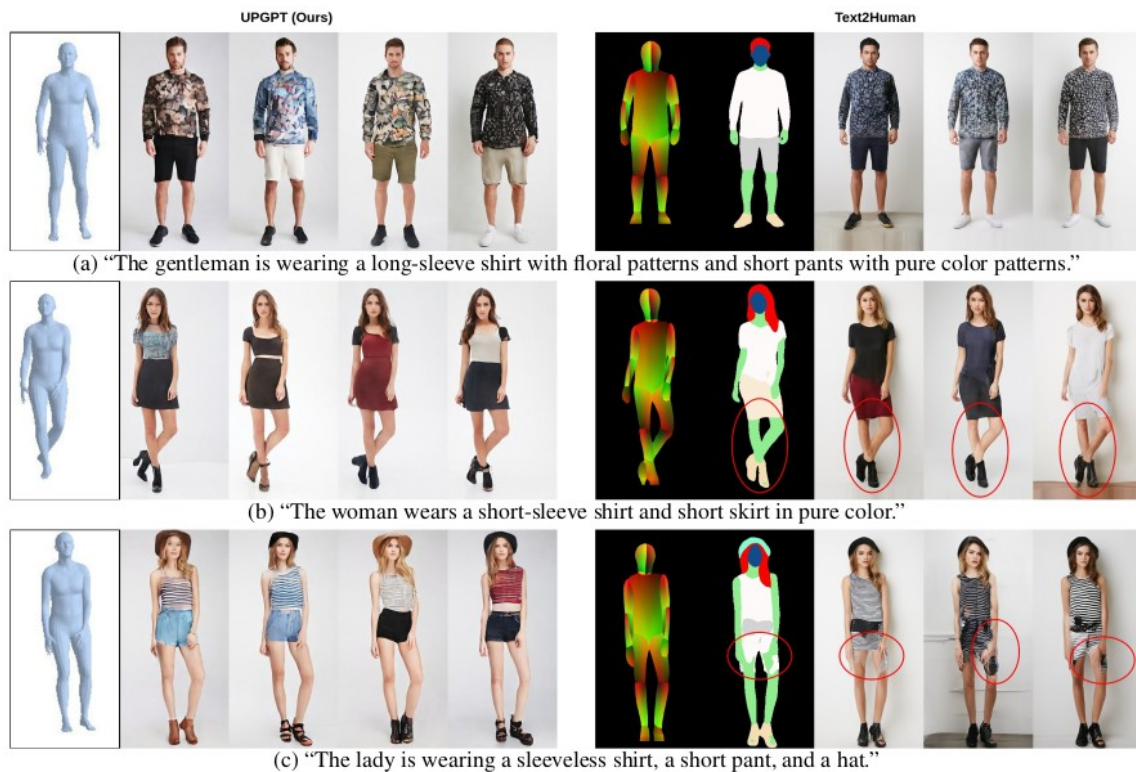
Tác giả sử dụng kiến trúc dựa trên Stable Diffusion nhưng cũng có nhiều cải tiến trong các thành phần.



Hình 25 Kiến trúc tổng quan của UPGPT [28]

Kiến trúc mô hình sử dụng rất nhiều thành phần như Image Encoder, SMPL Estimator, Style Image Encoder, Context Text Encoder. Nhưng với bài toán sinh ảnh người từ mô tả, chúng ta không cần thành phần Style Image Encoder (các thông tin về kiểu tóc, mặt, quần,... ở dạng ảnh sẽ được encode qua module này). Thông tin chi tiết về các module cần thiết trong hình 25:

- Image Latent sử dụng VAE để Encode và Decode.
- Text Encoder sử dụng ClipText đã được huấn luyện sẵn mục tiêu là các embedding của token.
- SMPL Pose: sử dụng PHOSA [29] với đầu vào là ảnh 2D đầu ra là 72 tham số SMPL đại diện cho 3 góc xoay của 24 điểm trên cơ thể, 10 tham số đại diện cho hình dạng cơ thể và 3 tham số đại diện cho góc nhìn. Các tham số này sẽ được cho qua một lớp Linear để tạo ra một embedding đại diện cho thông tin pose. Lớp Linear này sẽ được học trong quá trình huấn luyện.
- Bên cạnh việc lấy thông tin 3D thì PHOSA cũng cho ra một lớp mask giữa ảnh người và nền dùng để tăng cường học các thông tin người (reinforced person mask (RPM)). Ma trận mask này sẽ được thay đổi kích thước và concatenate với ma trận ảnh sau khi đã được cho qua autoencoder và thêm nhiễu.
- Các thông tin text embedding, pose embedding sẽ được concatenate lại thành một vector dùng để làm đầu vào cho mô hình U-Net.

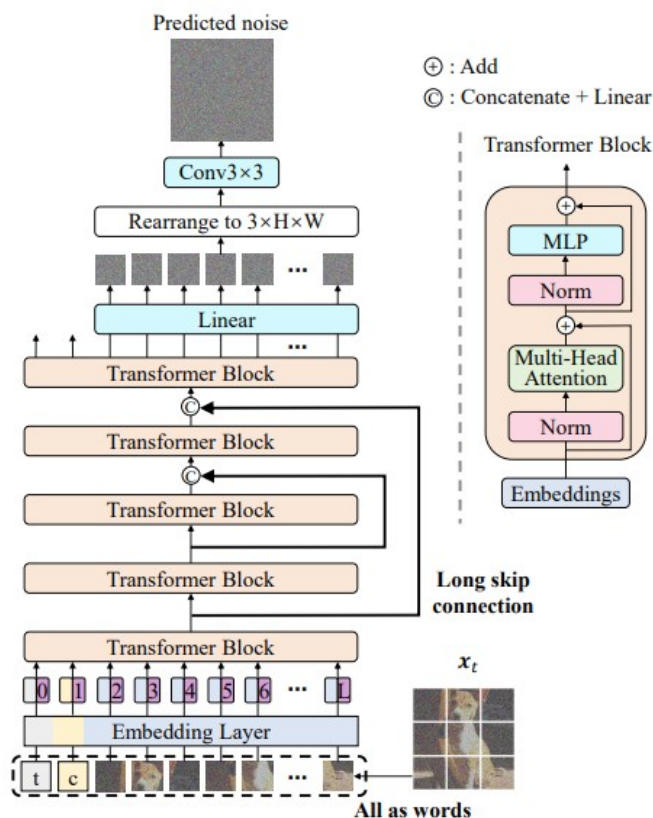


Hình 26 Một vài mẫu của mô hình UPGPT [28]

Hình trên so sánh UPGPT với Text2Human [29], cho thấy rằng cả hai đều có thể tạo ra hình ảnh chất lượng cao. Text2Human tạo ảnh người từ 3 thông tin: ảnh segment, ảnh chiều sâu kết hợp với văn bản. Khi ảnh đầu vào gặp một số lỗi bị chồng lên nhau thì ảnh đầu ra cũng vậy. UPGPT đã cải thiện tốt vấn đề này ảnh trong hình không còn bị lỗi trông chéo lên nhau. Điều này cho thấy sự tiềm năng của UPGPT để giải quyết bài toán Text2Human.

2.3 Mô hình U-ViT

Trong khi ViT đã cho thấy tiềm năng trong nhiều tác vụ thị giác khác nhau, thì mạng U-Net dựa trên mạng nơ-ron tích chập (CNN) vẫn là kiến trúc thống trị trong các mô hình khuếch tán. Nhóm tác giả đã thiết kế một kiến trúc đơn giản và tổng quát dựa trên ViT (gọi là U-ViT) để tạo ảnh với các mô hình khuếch tán. U-ViT [31] đặc biệt bởi việc xử lý tất cả các đầu vào, bao gồm số bước, điều kiện và các ô ảnh nhiều như các token, đồng thời sử dụng các kết nối dài giữa các lớp nông và sâu. U-ViT ngang ngửa hoặc vượt trội so với U-Net dựa trên CNN có kích thước tương tự. Đặc biệt, các mô hình khuếch tán tiềm ẩn với U-ViT đạt được điểm FID kỷ lục là 2,29 trong tạo ảnh có điều kiện theo lớp trên ImageNet 256x256 và 5,48 trong tạo ảnh từ văn bản trên MS-COCO.



Hình 27 Kiến trúc U-ViT [31]

Kiến trúc của U-ViT là sự kết hợp của ViT và Diffusion, đây là các điểm đặc biệt của mô hình:

- Kết hợp kết nối dài: giả sử $h_m, h_s \in R^{L \times D}$ là các embedding từ block chính và block kết nối dài. Kết hợp lại như sau $Linear(Concat(h_m, h_s))$.
- Kết hợp thông tin bước t : thông tin t sẽ được đưa vào mô hình như là một token.
- Thêm convolutional block sau Transformer: block này sẽ tính tích chập sau khi các miếng ảnh đã được chuyển qua embedding bằng linear theo kiến trúc ViT. Kết quả của block sẽ là đầu ra của mô hình U-ViT.

Kiến trúc ViT đã cho kết quả rất tốt trong bài toán Text2Image, nhóm tác giả cũng đã công bố pre-train của mô hình. Mặc dù nếu so với UPGPT, mô hình U-ViT không có sự tinh chỉnh riêng cho bài toán Text2Human. Đồ án chọn mô hình này vì muốn kiểm tra liệu ViT có tiềm năng giải quyết bài toán hay không.

2.4 Kết luận chương

Chương 2 đã giới thiệu về kiến trúc chung Stable Diffusion, thông tin chi tiết về mô hình UPGPT và U-ViT. Nêu ra lý do chọn 2 mô hình trên để giải quyết bài toán Text2Human.

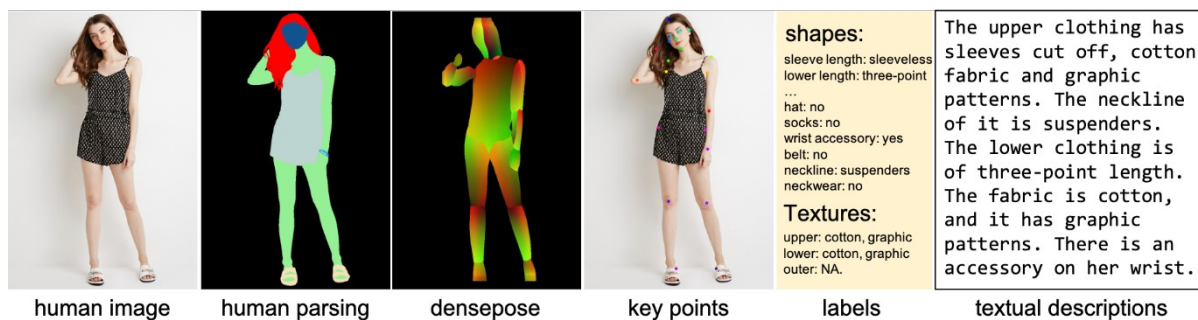
CHƯƠNG 3. DỮ LIỆU BÀI TOÁN TẠO ẢNH NGƯỜI

Trong phần này, đồ án sẽ trình bày về tập dữ liệu DeepFashion-MultiModal, cách xử lý tập dữ liệu này sang tiếng Việt và một số thống kê về tập dữ liệu.

3.1 Dữ liệu DeepFashion-MultiModal

DeepFashion-MultiModal [32] là một bộ dữ liệu con người thuộc miền thời trang có chất lượng cao, số lượng dữ liệu lớn với các chú thích đa dạng đa dạng. Các thông tin thuộc tính được thể hiện trong hình 28. Nó có những đặc tính sau đây:

- 1) Bao gồm 44,096 ảnh người chất lượng cao, bao gồm 12,701 ảnh toàn thân người.
- 2) Mỗi ảnh toàn thân sẽ được đánh 24 nhãn khác nhau.
- 3) Mỗi ảnh toàn thân đều được đánh key points
- 4) Được trích xuất sẵn DensePose cho mỗi ảnh người.
- 5) Mỗi ảnh đều được gán nhãn thuộc tính cho kiểu dáng và họa tiết của quần áo.
- 6) Mỗi ảnh đều có văn bản mô tả quần áo đang mặc của người trong ảnh.



Hình 28 Ví dụ dữ liệu DeepFashion-MultiModal

DeepFashion-MultiModal có thể được áp dụng vào việc tạo hình ảnh con người dựa trên văn bản, chỉnh sửa hình ảnh con người theo hướng dẫn văn bản, tạo hình ảnh con người dựa trên khuôn mẫu xương, ước tính tư thế con người, viết chú thích cho hình ảnh con người, nhận diện thuộc tính con người, dự đoán phân tích con người, v.v.

Tổng dung lượng của dữ liệu là khoảng 12GB, thông tin chi tiết về cấu trúc dữ liệu và định dạng sẽ có trong bảng 2:

Đường dẫn	Kích thước	Số lượng	Định dạng	Mô tả
DeepFashion-MultiModal	~12GB			thư mục chính
image	~5.4GB	44,096	JPG	các ảnh được lấy từ tập dữ liệu DeepFashion có kích thước 750x1101
parsing	~90MB	12,701	PNG	nhãn parsing
keypoints	956KB	2	TXT	nhãn keypoints

DensePose	~5.6GB	44,096	PNG	thông tin DensePose
labels	575KB	3	TXT	nhãn kiểu dáng, loại vải, màu sắc
textual descriptions	~11MB	1	JSON	văn bản mô tả cho mỗi ảnh

Bảng 2 Định dạng dữ liệu DeepFashion-MultiModal

Nhãn human parsing được định nghĩa trong bảng 3:

Danh sách nhãn			
0: 'background'	1: 'top'	2: 'outer'	3: 'skirt'
4: 'dress'	4: 'dress'	6: 'leggings'	7: 'headwear'
8: 'eyeglass'	9: 'neckwear'	9: 'neckwear'	11: 'footwear'
12: 'bag'	13: 'hair'	14: 'face'	15: 'skin'
15: 'skin'	17: 'wrist wearing'	18: 'socks'	19: 'gloves'
20: 'necklace'	21: 'rompers'	22: 'earrings'	23: 'tie'

Bảng 3 Nhãn human parsing của DeepFashion-MultiModal

Nhãn kiểu dáng được định nghĩa trong bảng 4:

id	Tên thuộc tính	Danh sách nhãn
0	sleeve length	0 sleeveless, 1 short-sleeve, 2 medium-sleeve, 3 long-sleeve, 4 not long-sleeve, 5 NA
1	lower clothing length	0 three-point, 1 medium short, 2 three-quarter, 3 long, 4 NA
2	socks	0 no, 1 socks, 2 leggings, 3 NA
3	hat	0 no, 1 yes, 2 NA
4	glasses	0 no, 1 eyeglasses, 2 sunglasses, 3 have a glasses in hand or clothes, 4 NA
5	neckwear	0 no, 1 yes, 2 NA
6	wrist wearing	0 no, 1 yes, 2 NA
7	ring	0 no, 1 yes, 2 NA
8	waist accessories	0 no, 1 belt, 2 have a clothing, 3 hidden, 4 NA

9	neckline	0 V-shape, 1 square, 2 round, 3 standing, 4 lapel, 5 suspenders, 6 NA
1 0	outer clothing a cardigan?	0 yes, 1 no, 2 NA
1 1	upper clothing covering navel	0 no, 1 yes, 2 NA

Bảng 4 Nhân kiểu dáng của DeepFashion-MultiModal

Nhãn họa tiết màu sắc được định nghĩa trong bảng 5:

Danh sách nhãn		
0 floral	1 graphic	2 striped
3 pure color	4 lattice	5 other
6 color block	7 NA	

Bảng 5 Nhãn họa tiết màu sắc của DeepFashion-MultiModal

NA: là không nhìn thấy hoặc bị che mắt trong ảnh

Bên trên là những thông tin chi tiết của tập dữ liệu DeepFashion-MultiModal. Tập dữ liệu này đã có nhiều nghiên cứu có kết quả tốt bên cạnh đó ở Việt Nam chưa có dữ liệu về Text2Human. Phần sau sẽ mô tả chi tiết cách xử lý tập dữ liệu sang tiếng Việt một cách hiệu quả.

3.2 Xử lý dữ liệu sang tiếng Việt

Để tạo một tập dữ liệu với một lượng lớn cặp (mô tả, ảnh) từ đầu là rất khó cả về mặt thời gian lẫn kiến thức chuyên môn để gán nhãn. Do vậy đồ án sẽ sử dụng giải pháp dịch tập dữ liệu DeepFashion-MultiModal từ tiếng Anh sang tiếng Việt. Vẫn giữ nguyên ảnh và các nhãn chỉ thay thế văn bản mô tả và thống nhất lại các định từ tiếng Anh sang tiếng Việt.

Để tối ưu chi phí và thời gian mà vẫn đủ dữ liệu tạo ra mô hình sinh ảnh người bằng tiếng mô tả tiếng Việt. Đồ án sẽ sử dụng tập dữ liệu đã qua bước tiền xử lý trong bài báo Text2Human với các bước sau:

- 1) Căn chỉnh cơ thể người về chính giữa của ảnh dựa trên thông tin tư thế người.
- 2) Làm một số nhãn sạch hơn và lọc bớt một số ảnh.
- 3) Gộp một số nhãn tương đồng nhau thành một.
- 4) Chia toàn bộ dữ liệu thành tập train và test.

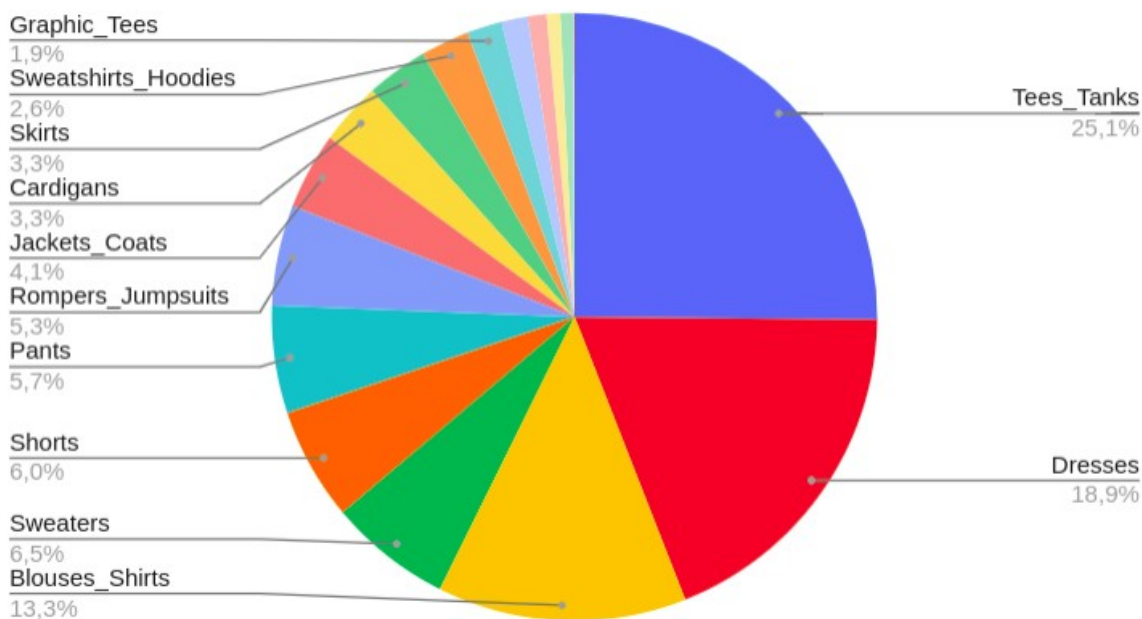
Sau khi tiền xử lý và chia tập train và test thì chúng ta sẽ có số lượng dữ liệu trong bảng 6:

Tập dữ liệu	train	test	tất cả
Số cặp text-image	10,335	1,149	11,484

Bảng 6 Thống kê số lượng dữ liệu

Với tỉ lệ các loại quần áo được thể hiện trong biểu đồ tại hình 29:

Tỉ lệ các loại quần áo



Hình 29 Biểu đồ tròn thể hiện tỉ lệ các loại quần áo

Phần trên là thông tin chi tiết về tập dữ liệu sẽ dùng cho quá trình huấn luyện và kiểm thử. Cho thấy Tees_Tanks và Dresses chiếm phần trăm khá lớn, điều này có thể dẫn đến mô hình khi train sẽ hoạt động tốt khi sinh ảnh loại này. Tiếp đến sẽ là thông tin chi tiết của từng bước xử lý.

3.2.1 Thống nhất định nghĩa

Do tính chất của ngôn ngữ, một từ có thể có nhiều nghĩa. Để giảm thiểu sự sai sót khi dịch thì bước đầu phải thống nhất các định nghĩa để không bị nhầm lẫn.

Định nghĩa về các loại quần áo sẽ được dịch sang tiếng Việt trong bảng 7:

Tên quần áo trong tiếng Anh	Kiểu đồ	Định nghĩa trong tiếng Việt
Suiting	bộ đồ	Vải may âu phục
Tees_Tanks	áo	Áo tank top/ Áo thun ba lỗ
Shorts	quần	Quần đùi
Leggings	quần	Quần bó/ Quần ôm sát chân
Denim	quần	vải bò
Sweaters	áo	Áo sweater/ Áo dài tay chui đầu
Jackets_Vests	áo	Áo khoác không tay/ Áo khoác gi lê

Jackets_Coats	áo	Áo khoác
Pants	quần	Quần thời trang
Graphic_Tees	áo	Áo thun in hình
Sweatshirts_Hoodies	áo	Áo hoodie/ Áo ni có mũ
Cardigans	áo	Áo khoác len
Shirts_Polos	áo	Áo polo
Rompers_Jumpsuits	bộ đồ	Bộ áo liền quần đùi/ Bộ áo liền quần dáng ngắn
Dresses	Bộ đồ	Cái đầm
Skirts	quần	chân váy
Blouses_Shirts	áo	Áo sơ mi cách điệu

Bảng 7 Các loại nhãn quần áo tiếng Việt

Định nghĩa về các các loại vải trong tập dữ liệu tại bảng 8:

Loại vải	Định nghĩa trong tiếng Việt
denim	bò
cotton	bông
leather	da
furry	lông
knitted	dệt kim
chiffon	voan
other	
NA	

Bảng 8 Các loại vải tiếng Việt

Định nghĩa về họa tiết màu sắc trong bảng 9:

Họa tiết	Định nghĩa trong tiếng Việt
floral	họa tiết hoa
graphic	hình học
striped	sọc
pure color	trơn màu/ Một màu
lattice	lưới
other	
color block	phối màu
NA	

Bảng 9 Các loại họa tiết tiếng Việt

Định nghĩa tiếng Anh về kiểu dáng các loại quần áo trong bảng 10 (attr là thuộc tính viết tắt):

id	Kiểu dáng	attr_0	attr_1	attr_2	attr_3	attr_4	attr_5	attr_6
0	sleeve length	sleeveless	short-sleeve	medium-sleeve	long-sleeve	not long-sleeve	NA	
1	lower clothing length	three-point	medium short	three-quarter	long-sleeve	NA		
2	socks	no	socks	leggings	NA			
3	hat	no	yes	NA				
4	glasses	no	eyeglasses	sunglasses	have a glasses in hand or clothes	NA		
5	neckwear	no	yes	NA				
6	wrist wearing	no	yes	NA				
7	ring	no	yes	NA				
8	waist accessories	no	belt	have a clothing	hidden	NA		
9	neckline	V-shape	square	round	standing	lapel	suspenders	NA
10	outer clothing a cardigan?	yes	no	NA				
11	upper clothing covering navel	no	yes	NA				

Bảng 10 Các loại kiểu dáng quần áo tiếng Anh

Bảng trên được dịch sang tiếng Việt như trong bảng 11:

id	shape	attr_0	attr_1	attr_2	attr_3	attr_4	attr_5	attr_6
0	độ dài tay áo	không có	ngắn	vừa	dài	không dài	NA	
1	độ dài của quần	ngắn	vừa	gần tới chân	dài	NA		
2	đi tất	không	tất	tất chân	NA			
3	đội mũ	không	có	NA				
4	dùng kính	không	kính	kính râm	đeo kính ở tay hoặc áo	NA		
5	vòng cổ	không	có	NA				
6	trang sức đeo tay	không	có	NA				
7	nhẫn	không	có	NA				
8	phụ kiện lưng	không	dây	buộc áo	không nhìn	NA		

			lưng		thấy			
9	kiểu cổ áo	hình V	vuông	tròn	cao	lapel	đai đeo quần	NA
1 0	bên ngoài có khoác áo không?	có	không	NA				
1 1	phần áo trên có phải áo croptop không ?	không	có	NA				

Bảng 11 Các loại kiểu dáng quần áo tiếng Việt

Sau khi định nghĩa rõ ràng và hiểu rõ dữ liệu, bước tiếp theo là tìm ra cách dịch tất cả các mẫu dữ liệu sang tiếng Việt một cách nhanh và đạt độ chính xác cao nhất.

3.2.2 Phương pháp dịch từ tiếng Anh sang tiếng Việt

Việc dịch thủ công tốn rất nhiều thời gian kể cả khi đã biết hết các định nghĩa chuyên ngành thời trang. Để tăng tốc độ và tiết kiệm công sức, dữ liệu sẽ được cho qua một mô hình dịch từ tiếng Anh sang Việt. Sau đó sẽ được xem lại và chỉnh sửa các lỗi sai của mô hình.

Hiện nay có 4 phương án tốt nhất dùng để dịch tiếng Việt:

- mô hình VietAI/envit5-translation
- mô hình vinai/vinai-translate-en2vi
- google dịch
- chatGPT (sử dụng chatGPT 3.5)

Để đánh giá mô hình dịch nào tốt nhất với tập dữ liệu. Tạo dữ liệu đánh giá mô hình dịch, lấy ngẫu nhiên 250 mẫu từ tập dữ liệu tiếng Anh. Mỗi mẫu sẽ có 4 bản dịch tương ứng với từng mẫu. Để cho công bằng và đảm bảo càng nhiều bản dịch tự nhiên nhất có thể, đề án tạo ra một bộ nhãn để tính điểm cho bản dịch của các mô hình dịch.

Nhãn	Mô tả	Điểm số
dịch chuẩn	dịch tốt, câu từ chuẩn, sát nghĩa, không sai ngữ pháp.	3
dịch ổn	chưa hoàn hảo, ở mức chấp nhận được.	2
dịch tệ	dịch tệ	1
dịch sai	dịch sai	0

Bảng 12 Thang đo chất lượng dịch

Ngoài ra có thêm nhãn “tốt nhất” để chọn ra bản dịch tự nhiên nhất (phòng trường hợp có điểm số bằng nhau). Các bản dịch của một mẫu sẽ được người đánh nhãn. Sau khi đánh nhãn xong sẽ chọn ra mô hình có điểm số cao nhất dịch toàn bộ dữ liệu. Dưới đây là bảng điểm sau khi đánh nhãn cho 250 mẫu.



model	số lượng	tổng	số lượng dịch	số lượng	số lượng	số lượng
-------	----------	------	---------------	----------	----------	----------

	tốt nhất	điểm	chuẩn	dịch ổn	dịch tệ	dịch sai
envit5-translation	40	434	16	164	61	9
vinai-translation	12	318	4	126	60	60
google-translation	54	437	14	171	53	10
chatgpt-translate	144	513	41	189	18	2

Bảng 13 Bảng điểm sau khi thực hiện gán nhãn bằng tay

Kết hợp hệ số trong bảng 12 và các thông tin số lượng trong bảng 13 tính được thông tin điểm bằng cách: **Điểm = số lượng dịch chuẩn*3 + số lượng dịch ổn*2 + số lượng dịch tệ*1**. Ví dụ: điểm số của **envit5-translation** = 16*3+164*2+61*1 = 437.

Ta có thể thấy rằng chatGPT là thành dịch tốt nhất với 513 điểm, và số lượng mẫu được đánh nhãn tốt nhất lên đến 144. Hình 30 dưới đây thể hiện một vài mẫu đã được dịch và đánh điểm cho các mô hình.

caption	envit5-translation	envit5-comment	vinai-translation	vinai-comment	google-translation	google-comment	chatgpt-translate	chatgpt-comment	best	image
This person wears a sleeveless tank top with graphic patterns and three-point pants. The tank top is with cotton fabric. The pants are with cotton fabric and graphic patterns. There is an accessory in his her neck.	Người này mặc một chiếc áo ba lỗ không tay với các họa tiết và một chiếc quần ba điểm. Chiếc áo ba lỗ làm bằng vải cotton. Chiếc quần làm bằng vải cotton và các họa tiết. Có một phụ kiện ở cổ cổ ta.	dịch ổn	Người này mặc một chiếc xe tăng không tay với họa văn đồ hoa và quần ba điểm. Đầu xe tăng là với vải cotton. Quần được làm bằng vải cotton và họa văn đồ hoa. Có một phụ kiện ở cổ cổ ấy.	dịch sai	Người này mặc áo ba lỗ không tay có họa văn hình hoa và quần ba chấm. Áo ba lỗ bằng vải cotton. Quần được làm bằng vải cotton và họa tiết đồ hoa. Có một phụ kiện trong cổ của mình.	dịch ổn	Người này mặc áo tank top không có tay với họa tiết đồ hoa và quần dài ba điểm. Áo tank top được làm bằng chất liệu cotton. Quần cũng làm từ vải cotton và có họa tiết đồ hoa. Còn có một phụ kiện ở cổ của anh ta / cô ấy.	dịch ổn	envit5-translation	
The tank top the lady wears has sleeves cut off, its fabric is cotton, and it has graphic patterns. The lady wears a three-point pants, with cotton fabric and graphic patterns. This female has neckwear. The person has a hat in her head.	Chiếc áo ba lỗ mà người phụ nữ mặc có tay áo bị cắt bỏ, vải của nó là cotton và có họa văn đồ hoa. Người phụ nữ mặc một chiếc quần ba điểm, vải cotton và họa văn đồ hoa. Người phụ nữ này mặc áo cổ. Người đó đội một chiếc mũ trên đầu.	dịch tệ	Chiếc xe tăng mà người phụ nữ mặc có tay áo bị cắt, vải của nó là cotton và nó có họa văn đồ hoa. Người phụ nữ mặc quần ba điểm, với vải cotton và họa văn đồ hoa. Người phụ nữ này có bông tai. Người có một chiếc mũ trong đầu.	dịch sai	Chiếc áo ba lỗ mà người phụ nữ mặc có tay áo bị cắt, vải của nó là cotton và có họa văn hình học. Đây có mặc quần ba chấm, bằng vải cotton, họa tiết hình hoa. Nữ này có khăn choàng cổ. Người đội mũ trong đầu.	dịch sai	Cô gái này đang mặc một chiếc áo tank top có tay bị cắt đứt, chất liệu của nó là cotton và có họa tiết đồ hoa. Cô gái đang mặc một chiếc quần dài ba điểm, cũng được làm từ vải cotton và có họa tiết đồ hoa. Cô ấy đang mang một phụ kiện trên cổ. Người này còn đội một chiếc mũ trên đầu.	dịch ổn	chatgpt-translate	

Hình 30 Ví dụ về các mẫu dịch

Việc chỉ đưa thông tin văn bản vào chatGPT để dịch cũng cho ra kết quả rất tốt nhưng vẫn còn có thể cải thiện thêm bằng cách cải thiện phần prompt đầu vào.

Với prompt “Dịch đoạn sau sang tiếng Việt: ...”, ở bảng trên chúng ta có thể thấy rằng số lượng dịch chuẩn chỉ có 41 mẫu. Để cải thiện hơn nữa, tôi thử cung cấp thêm thông tin cho chatGPT để nó dịch chuẩn hơn.

Trong bảng 14 là các mẫu prompt đã thử để tạo mô tả tiếng Việt từ tiếng Anh. Với {en_cloth}-mô tả tiếng Anh, {vi_label}-nhãn tiếng Việt, {vi_gender}-giới tính tiếng Việt là các thông tin cần điền vào mẫu.

stt	Prompt	Miêu tả	Ví dụ
1	Dịch đoạn văn sau sang tiếng Việt : “{en_cloth}”	Yêu cầu dịch thông thường, chat-gpt không hiểu được các từ như “three-point”, “three-quarter”, ...	Dịch đoạn văn tiếng anh sau ra tiếng Việt: "This person wears a tank shirt with graphic patterns. The shirt is with chiffon fabric. It has a suspenders neckline. The pants this person wears is of short

			length. The pants are with chiffon fabric and pure color patterns. There is a ring on her finger. This lady has neckwear."
2	Dùng nội dung "{vi_label}";", mô tả chi tiết về trang phục của {vi_gender}. Yêu cầu ngắn gọn, không mô tả bất kỳ ý kiến hoặc đánh giá.	Không dịch, bắt chatGPT gen ra một đoạn mô tả dựa trên các nhãn mà mulmodal dataset cung cấp. Yêu cầu chatGPT không tạo thêm thông tin không cần thiết.	Dùng nội dung “nam;áo sọt bông,trơn màu,cổ tròn;quần sọt bông,dài,họa tiết lưới” mô tả chi tiết về trang phục của người đàn ông. Yêu cầu ngắn gọn, không mô tả bất kỳ ý kiến hoặc đánh giá.
3	Dịch đoạn văn tiếng anh sau ra tiếng Việt:"{en_cloth}". Khi dịch sang tiếng Việt thì sử dụng các đặc điểm như trong bản tóm tắt sau: "{vi_label}"	Cung cấp thêm thông tin bản dịch tiếng Anh và cả thông tin các từ khóa tiếng Anh đã được dịch sang tiếng Việt.	Dịch đoạn văn tiếng anh sau ra tiếng Việt: "This person wears a tank shirt with graphic patterns. The shirt is with chiffon fabric. It has a suspenders neckline. The pants this person wears is of short length. The pants are with chiffon fabric and pure color patterns. There is a ring on her finger. This lady has neckwear." Khi dịch sang tiếng Việt thì sử dụng các đặc điểm như trong bản tóm tắt sau: "Đồ phía trên: kiểu áo hai dây, có họa tiết. Đồ phía dưới: ngắn, chất liệu làm bằng vải voan, có trơn màu . Phụ kiện:có vòng cổ, có đeo nhẫn".

Bảng 14 Các mẫu prompt dùng để dịch dữ liệu

Thấy rằng việc càng cung cấp thêm thông tin cho chatGPT thì chất lượng bản mô tả tạo ra càng chuẩn.

Với các tiêu chí tạo ra các bản mô tả chuẩn cho ảnh, đọc thật tự nhiên, đa dạng kiểu câu. Đồ án sẽ ưu tiên dùng bản dịch thứ 3, nếu bản dịch này sai ít thì sẽ sửa lại. Còn nếu sai nhiều thì sẽ chọn sang bản dịch thứ 2. Như vậy sẽ tăng tốc độ rà soát lại dữ liệu lên rất nhiều và độ chính xác do người dịch vẫn được đảm bảo.

3.3 Kết luận chương

Trong chương 3, đồ án đã mô tả chi tiết về tập dữ liệu DeepFashion-MultiModal, tiền xử lý và quá trình dịch từ tiếng Anh sang tiếng Việt. Nội dung trên đã chỉ rõ sự hiệu quả khi dùng chatGPT trên tập dữ liệu này. Dữ liệu đảm bảo đã được người rà soát lại toàn bộ để giảm sai sót xuống thấp nhất có thể. Phần tiếp theo sẽ là các thực nghiệm và đánh các mô hình với tập dữ liệu thuần Việt.

CHƯƠNG 4. THỰC NGHIỆM ĐÁNH GIÁ

Trong chương này sẽ trình bày về quy trình thực nghiệm bài toán sinh ảnh người từ mô tả tiếng Việt với các mô hình đã nêu trên, bao gồm: trình bày về các thiết lập thực nghiệm và công cụ sử dụng cho thực nghiệm, và cuối cùng là đưa ra kết quả thực nghiệm và các đánh giá. Các kết quả thực nghiệm trên các mô hình sẽ được thống kê và việc so sánh đánh giá các mô hình.

5.1 Quy trình thực nghiệm

5.2 Thiết lập thực nghiệm

Sau khi xử lý tập dữ liệu cấu trúc dữ liệu sử dụng sẽ như sau:

- Một folder ảnh chứa tất cả 11,484 ảnh của tập train và test.
- 2 file json train và test với nội dung chứa mô tả theo mỗi id của ảnh. Hình 31 là cấu trúc của file json.

```
"MEN-Denim-id_00004051-01_4_full": "Người đàn ông này mặc một chiếc áo
phông cô'tròn với màu trơn. Áo được làm bằng vải bông. Quần mà người đàn
ông này mặc là dài. Quần được làm bằng vải bò và có màu trơn.",
"WOMEN-Dresses-id_00005547-01_1_front": "Người phụ nữ đang mặc một chiếc
đầm dài gân tới chân, được làm từ sợi bông. Đầm không có tay và có họa tiết
trang trí. Kiểu đầm này có hai dây để giữ chặt trên vai. Ngoài ra, người
phụ nữ còn đeo vòng tay và nhẫn.",
"WOMEN-Tees_Tanks-id_00007274-01_7_additional": "Người phụ nữ mặc một chiếc
áo dài tay với màu trơn và một chiếc quần ngắn. Áo được làm bằng vải bông
và cô'áo tròn. Quần cũng được làm bằng vải bông và có màu trơn. Người này
đang đeo kính râm và có một chiếc nhẫn trên ngón tay.",
"WOMEN-Dresses-id_00000852-04_4_full": "Cô gái mặc một chiếc đầm ngắn, sợi
bông, tay dài, có họa tiết và cô'tròn. Cô ấy đeo vòng cổ'và nhẫn. Bên
ngoài, cô ấy khoác áo dẹt kim, trơn màu.",
"WOMEN-Sweatshirts_Hoodies-id_00000126-04_4_full": "Người này đang mặc một
chiếc áo phông dài tay với họa tiết sọc. Áo phông được làm từ vải dẹt kim.
Nó có cô'chữ V. Người này đang mặc một chiếc quần dài. Quần được làm từ vải
bò và có màu trơn. Người này đang đeo một chiếc nhẫn trên ngón tay.",
"WOMEN-Blouses_Shirts-id_00003217-01_1_front": "Cô gái ấy mặc một chiếc áo
cách điệu sợi bông, không có tay, trơn màu, cô'tròn. Cô ấy kết hợp với một
chiếc quần bò dài, trơn màu. Trên tay, cô gái ấy đeo một chiếc nhẫn.",
"WOMEN-Dresses-id_00002942-06_7_additional": "Người phụ nữ đang mặc một
chiếc đầm ngắn, được làm từ chất liệu voan mềm mại. Đầm không có tay và có
họa tiết trang trí trên bề'mặt. Kiểu dáng của đầm là hai dây. Người phụ nữ
cũng đang đeo một chiếc vòng cổ'và một chiếc nhẫn.",
```

Hình 31 Cấu trúc file mô tả tiếng Việt

5.2.1 Môi trường và công cụ thực nghiệm

Toàn bộ thực nghiệm được thực hiện trên môi trường với các thông tin phần cứng và môi trường thực hiện được miêu tả như hai bảng dưới:

Phần cứng	Thông số	Số lượng
CPU		1
GPU	Nvidia 3090 RTX	1
RAM	32GB	1
Ổ cứng SSD	1TB	1

Bảng 15 Thông số phần cứng

Công cụ phần mềm	Phiên bản
python	3.8.5
pytorch	1.11.0
torchvision	0.12.0
numpy	1.19.2
scikit-learn	1.1.3
pytorch_fid	0.3.0
transformers	4.19.2
opencv-python	4.1.2.30
pytorch-lightning	1.6.0
omegaconf	2.1.1
timm	0.3.2
taming-transformers	git+https://github.com/CompVis/taming-transformers.git
clip	git+https://github.com/openai/CLIP.git

Bảng 16 Môi trường phát triển

5.2.2 Chi tiết thực nghiệm

Đồ án thực nghiệm 2 kiến trúc mô hình trên tập dữ liệu đã dịch. Các mô hình đều sử dụng chung pretrained M-CLIP/XLM-Roberta-Large-Vit-L-14 để tạo word embedding cho từng từ token trong câu. Tập dữ liệu huấn luyện (train set) giúp cập nhật bộ trọng số tốt nhất cho từng mô hình. Do dữ liệu ban đầu không có tập dữ liệu thẩm định (validation set) nên trong thực nghiệm sẽ chọn ngẫu nhiên 100 ảnh trong tập huấn luyện để đánh giá mô hình trong lúc huấn luyện. Mô hình sau khi huấn luyện với bộ tham số đạt kết quả tốt nhất sẽ được sử dụng để đánh giá kết quả trên tập dữ liệu kiểm tra (test set).

Sau đây sẽ là bảng thống kê tham số thực nghiệm tốt nhất của các mô hình:

Mô hình	Batch size	lr	Max length	Num epoch	image size	sample steps	clip dim	các tham số khác
UPGPT	8	1e-6	146	100	256	50	1024	pose_dim = 1024
U-ViT	128	2e-4	146	100	256	50	1024	depth = 16

Bảng 17 Kết quả thực nghiệm

Diễn giải các tham số mô hình:

- Batch size (kích thước lô): số câu đầu vào cho mỗi lần thực hiện tính toán hàm mất mát và thực hiện lan truyền ngược.
- Lr (tốc độ học): lượng cập nhật tham số sau mỗi lần thực hiện lan truyền ngược.
- Max length: độ dài câu đầu vào tính theo token.
- Num epoch: số lượng epoch, mỗi epoch là một lần đưa hết tất cả dữ liệu vào mô hình, một epoch thường được chia nhỏ thành từng lô (batch).
- image_size: kích thước resize của các ảnh đầu vào trước khi cho vào encoder
- pose_dim: chiều của thông tin 3D sau khi được mã hóa
- depth: chiều sâu của mạng ViT
- clip dim: chiều của thông tin văn bản sau khi được mã hóa
- sample steps: số bước trong quá trình diffusion

5.3 Các chỉ số đánh giá cho bài toán

Không giống các bài toán khác, việc đánh giá một mô hình sinh ảnh có tốt hay không dựa trên 2 yếu tố chính:

- Chất lượng ảnh: Ảnh sinh ra phải có chất lượng cao, giống với dataset.
- Độ đa dạng: Generator cần sinh ra được nhiều ảnh khác nhau thuộc nhiều lớp khác nhau. Nếu Generator mãi chỉ sinh ra được một vài ảnh hay thuộc cùng 1 class thì cũng không tốt.

Cách đơn giản nhất đó, đó là trực tiếp dùng mắt thường để đánh giá. Ưu điểm của phương pháp này đó là xác nhận rất nhanh việc mô hình có học được không, nhưng nhược điểm của nó là không khách quan (mỗi người có một định nghĩa ảnh tốt ảnh đẹp khác nhau), không áp dụng được trên một tập dữ liệu lớn.

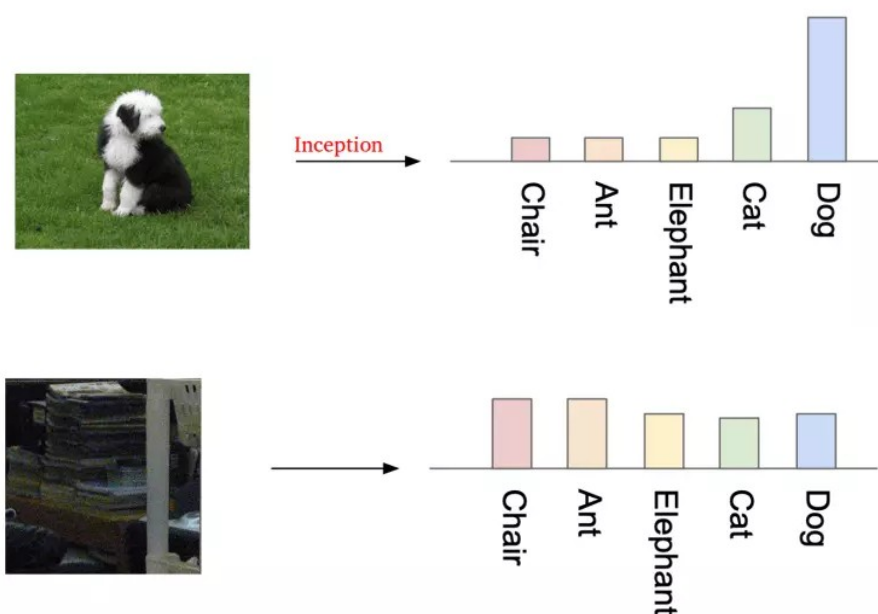
Sau đây sẽ là các độ đo mà được sử dụng để đánh giá mô hình sinh ảnh, các độ đo này cũng đã được dùng để đánh giá các mô hình sinh ảnh người bằng tiếng Anh.

	Inception Score (IS)	Frechet Inception Distance (FID)	SSIM
Cách đánh giá	IS càng cao càng tốt.	FID không âm và FID càng thấp càng tốt.	Trong khoảng (-1,1), càng gần 1 càng tốt.

Bảng 18 Các độ đo của bài toán sinh ảnh

5.3.1 Inception Score (IS)

IS được tính bằng cách dùng pre-trained model Inception trên dữ liệu ImageNet. Input của mạng Inception là một ảnh và output với hàm softmax sẽ cho ra được xác suất ảnh đấy thuộc từng lớp tương ứng.



Hình 33 Ví dụ về độ đo IS

Nhận xét: khi qua Inception, ảnh rõ ràng thuộc một lớp nào đó sẽ cho ra xác suất tại lớp đó là rất cao, còn ảnh không thuộc một lớp nào sẽ cho ra xác suất giữa các lớp gần giống nhau (uniform).

Dựa vào Inception sẽ kiểm tra được 2 yếu tố:

- Chất lượng ảnh: Cho ảnh sinh ra qua mạng Inception nếu ảnh rõ, tốt thì mạng sẽ phân loại tốt (xác suất 1 lớp cao hơn hẳn).
- Độ đa dạng của ảnh sinh ra: Cộng theo từng lớp các giá trị xác suất của tất cả

các ảnh sinh ra trong generator. Nếu Generator có thể sinh ra đa dạng dữ liệu thì tổng xác suất sẽ dạng uniform, ngược lại nếu dữ liệu chính sinh ra dữ liệu ở 1 hay 2 lớp thì tổng xác suất sẽ chỉ cao hơn ở 1 hay 2 lớp.

Tính giá trị KLD cho mỗi ảnh sinh ra rồi lấy trung bình lại làm giá trị IS cho model.

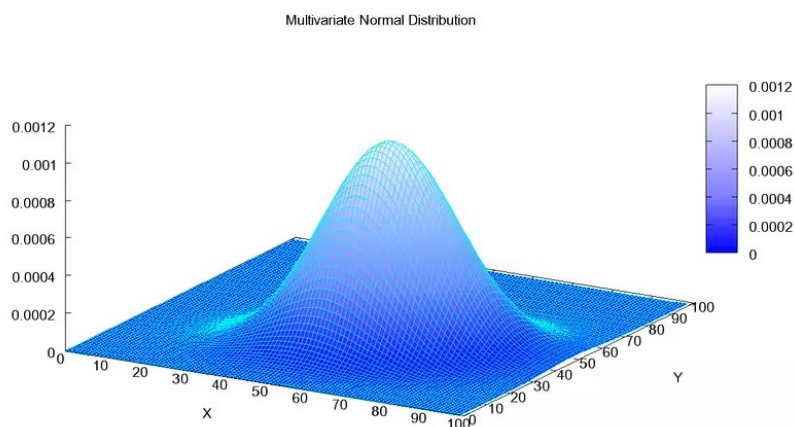
$$IS(G) = \exp(\mathbb{E}_{x \sim p_g(x)} D_{KL}(p(y) \| p(y|x)))$$

Hạn chế:

- Nếu Generator chỉ sinh được một ảnh mỗi lớp thì chỉ số KL vẫn có thể cao => Chỉ đa dạng lớp nhưng không đa dạng ảnh trong mỗi lớp.
- Nếu Generator nhớ và sinh ra các ảnh trong dataset thì chỉ số KL cũng cao => Không tốt

5.3.2 Fehet Inception Distance (FID)

Phân phối Gauss nhiều chiều, là tổng quát hóa của phân phối chuẩn một chiều cho không gian nhiều chiều hơn.



Hình 34 Minh họa FID

Cách tính FID phụ thuộc vào Inception network (giữ đến lớp AvgPool, bỏ phần sau). Như vậy, output của mỗi ảnh sẽ là 1 vector 2048×1 . Giả sử, tập datasets của chúng ta có n ảnh.

Các bước tiến hành:

- Cho n ảnh trong tập datasets chạy qua Inception network được n vector 2048×1 => tìm được 1 multivariate gaussian distribution ứng với n vector này.
- Cho n ảnh sinh ra cũng chạy qua Inception network được n vector 2048×1 => tương tự cũng tìm được 1 multivariate gaussian distribution
- Để các ảnh sinh ra giống các ảnh trong dataset thì ta mong muốn 2 multivariate gaussian distribution giống nhau, hay mean và variance gần nhau.

Công thức FID:

$$FID = \|\mu_x - \mu_g\|_2^2 + Tr(\Sigma_x + \Sigma_g - 2(\Sigma_x \Sigma_g)^{\frac{1}{2}})$$

Nhận xét:

- FID không âm và FID càng thấp thì 2 distribution càng gần nhau => ảnh sinh ra càng giống ảnh gốc trong dataset.
- Không giống như Inception Score (IS), chỉ đánh giá sự phân bố của các ảnh được sinh ra, FID so sánh sự phân bố của các hình ảnh được sinh ra với sự phân bố của các hình ảnh thực tế được sử dụng.

5.3.3 Structural Similarity Index Measurement (SSIM)

Chỉ số SSIM được sử dụng để đo mức độ giống nhau giữa hình ảnh đầu vào và ảnh sinh ra. Công thức SSIM dựa trên ba thống số để so sánh: độ chói (luminance), tương phản (contrast) và cấu trúc (structure). Một ảnh sinh ra là tốt nếu:

- Những điểm ảnh có mức độ sáng tốt khác nhau, và càng có nhiều mức độ sáng tối càng có nhiều chi tiết ảnh. => Ảnh chất lượng tốt.
- Một bức ảnh không phải có độ tương phản càng cao thì càng tốt mà nên có sự hài hòa cân đối giữa sáng và tối. => độ đa dạng.

Công thức SSIM:

$$SSIM(x,y)=[l(x,y)]^\alpha \cdot [c(x,y)]^\beta \cdot [s(x,y)]^\gamma$$

Nhận xét: SSIM có giá trị trong khoảng từ -1 đến 1, đạt giá trị bằng 1 trong trường hợp hai bộ dữ liệu giống hệt nhau. Chỉ số này có giá trị càng lớn thì tương ứng với model càng tốt.

5.4 Kết quả thực nghiệm

Đồ án thực nghiệm trên các mô hình trên và tinh chỉnh để phù hợp với tiếng Việt. Đây là kết quả trên tập test sau khi huấn luyện mô hình xong.

Độ đo\Mô hình	IS	FID	SSIM
UPGPT	3.6261	30.54	0.74
U-ViT	3.3339	36.16	0.734

Bảng 19 Kết quả thực nghiệm

Mô hình	FID
HumanDiffusion	30.42
Text2Human	24.52
UPGPT	23.46

Bảng 20 Kết quả với mô tả là tiếng Anh

Từ bảng kết quả thực nghiệm (Bảng 19) và kết quả các mô hình tiếng Anh đã đạt được (Bảng 20), tiếp theo đề án sẽ tiến hành đánh giá kết quả thu được như sau:

- Cả 2 mô hình đề có khả năng sinh ảnh người bằng văn bản tiếng Việt.
- Do được tối ưu cho bài toán sinh ảnh người bằng việc thêm thông tin đầu vào nên UPGPT có điểm số cao hơn hẳn so với U-ViT.
- So với kết quả tiếng Anh thì các mô hình tiếng Việt có kết quả thấp hơn, do có sự khác nhau của pre-train đa ngôn ngữ dùng cho tiếng Việt kém hơn mô hình dữ liệu lớn thuần tiếng Anh.

5.5 Phân tích lỗi sai

Từ các mô hình đã huấn luyện và các đánh giá trên tập dữ liệu kiểm tra, đề án chỉ ra các lỗi sai mà mô hình thường gặp phải trên mô hình tốt nhất UPGPT. Sau đây là chi tiết các lỗi sai:

- Khuôn mặt thường bị mờ do đặc trưng AE là nén dữ liệu nên rất khó để giữ được đặc trưng khuôn mặt với các chi tiết nhỏ như mắt, mũi, tai, v.v.



Hình 35 Lỗi mờ mặt

- Chưa xử lý được một số dáng người khó, ví dụ dáng chân trong ảnh người và hướng của người mẫu là chuẩn nhưng chi tiết dáng tay và hướng của đầu sau khi sinh ra lại sai.



Hình 36 Lỗi sai kiểu dáng khó

- Các thông tin nhỏ như vòng cổ, vòng tay, nhẫn nhiều khi bị biến mất do các chi tiết nhỏ khi decode ra thường bị mờ. Ví dụ mô tả sau: “*Người phụ nữ đang mặc một chiếc đầm dài, được làm từ sợi bông, không có tay. Cô ấy cũng đeo một chiếc vòng cổ và một chiếc nhẫn.*”



Hình 37 Lỗi không nhìn rõ trang sức nhỏ

5.6 Kết luận chương

Chương 5 đã trình bày cách làm thực nghiệm các mô hình trên bộ dữ liệu tiếng Việt mà đồ án đã xử lý. Dựa trên các kết quả thực nghiệm và kiến thức về các phương pháp, chương 5 cũng chỉ ra lỗi sai ưu và nhược điểm của các mô hình, từ đó đề xuất ra các hướng nghiên cứu cải tiến tiếp theo.

KẾT LUẬN

Đồ án này đã trình bày chi tiết về bài toán sinh ảnh người từ mô tả tiếng Việt, cùng với đó là hai kiến trúc mô hình giúp giải quyết bài toán. Từ đó, đồ án tiếng hành xử lý để có được tập dữ liệu đầu tiên của tiếng Việt cho bài toán sinh ảnh người từ mô tả. Đồ án tiến hành thực nghiệm để đánh giá độ chính xác, tính hiệu quả của các mô hình. Từ nghiên cứu đầu cho bài toán này, đồ án huy vọng sẽ lấy đó làm kiến thức, nền tảng để các nghiên cứu sau này phát triển các phương pháp học sâu mới, tận dụng những điểm mạnh của các mô hình trên và khắc phục những vấn đề còn tồn đọng, đặc biệt là các nghiên cứu mô hình trên tập dữ liệu tiếng Việt. Với nền tảng như vậy, đây sẽ là một bước đầu tiên làm nền tảng của các nghiên cứu sau này.

Từ kết quả thực nghiệm và kiến thức về các phương pháp sử dụng khuếch tán ổn định cho bài toán sinh ảnh người từ mô tả tiếng Việt, đồ án cũng đưa ra một số hướng cải thiện tiếp theo:

- Hiện nay các mô hình VAE đang phát triển rất mạnh với tốc độ nhanh hơn và kết quả chất lượng hơn. Việc tinh chỉnh tiếp khối này bằng cách thay các pre-train lớn hơn sẽ giúp mô hình tạo ảnh tốt hơn.
- Cải thiện phần TextEncoder bằng cách huấn luyện thêm cho với ngôn ngữ tiếng Việt để có thể cải thiện thêm ngữ nghĩa cho mô hình.
- Việc kết hợp thêm nhiều thông tin vào mô hình để tạo ảnh, như thêm thông tin chi tiết về màu sắc, tóc tai, v.v. sẽ làm mô hình tốt hơn nhiều nên việc mở rộng tập dữ liệu cả về kích thước, chất lượng của các mô tả trong tương lai có thể sẽ được bổ sung.

A: PHỤ LỤC

Phần này trình bày về hệ thống sinh mô tả tiếng việt cho ảnh dựa trên việc triển khai các mạng nơ ron học sâu đã trình bày ở trên.

A.1 Tổng quan hệ thống

Hệ thống sinh mô tả tiếng Việt cho ảnh là một trang web được xây dựng bằng ngôn ngữ python với framework fastapi, html với mục đích thử nghiệm độ chính xác của các mô hình đã khảo sát và thực nghiệm cũng như tính thực tiễn của chúng. Bất cứ người dùng nào truy cập vào web này cũng có thể sử dụng được hệ thống. Sau khi truy cập hệ thống, người dùng thực hiện nhập một mô tả người. Sau khi ấn nút “Sinh ảnh” để gửi dữ liệu cần xử lý, hệ thống sẽ sinh ra ảnh tương ứng và gửi kết quả về giao diện người dùng.

A.2 Các công cụ sử dụng

Các mô hình được triển khai trên server có cấu hình như sau:

- ubuntu 18
- GPU GeForce GTX 3090 Ti

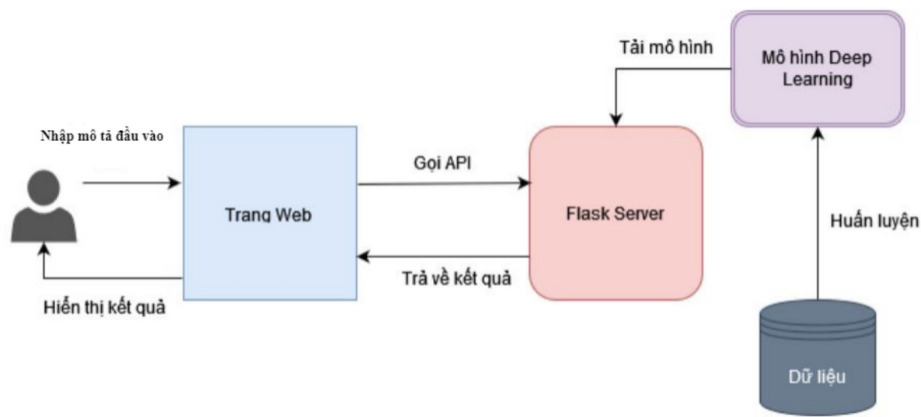
Người dùng có thể truy cập địa chỉ IP để trải nghiệm hệ thống. Các công cụ chính sử dụng để thực hiện hệ thống trong đề án được liệt kê ở bảng dưới đây:

Công cụ	Trang chủ
FastAPI	https://fastapi.tiangolo.com/
Anaconda	https://www.anaconda.com/

- FastAPI: Là nền tảng xây dựng ứng dụng Web sử dụng ngôn ngữ lập trình Python. Được sử dụng để xây dựng hệ thống sinh ảnh.
- Anaconda: Dùng để quản lý và cài đặt môi trường Python, dùng để chạy mô hình và web.

A.3 Xây dựng hệ thống

Hệ thống gồm các mô hình học sâu được huấn luyện trên tập dữ liệu thuần việt với miền là các văn bản mô tả thông tin quần áo, trang sức, v.v. của người. Do đó hệ thống chỉ có thể thực hiện ảnh với mô tả tiếng Việt tương ứng thuộc miền dữ liệu này.



Hình 38 Kiến trúc hệ thống sinh ảnh người sử dụng mô tả tiếng Việt

- Các mô hình sẽ được huấn luyện trước và được lưu trữ trong folder của hệ thống.
- Sau khi khởi động môi trường web, các mô hình này sẽ được tải vào FastAPI Server.
- Người dùng hệ thống thực hiện viết mô tả và chọn loại mô hình để sinh ra mô tả. Sau khi bấm vào nút “Sinh ảnh”, Server xử lý ảnh và gọi đến mô hình tương ứng. Mô tả được sinh ra được trả về cho trang Web. Cuối cùng trang web sẽ hiển thị kết quả thân thiện với người dùng.

A.4 Một số kết quả của mô hình

Sau đây sẽ là một số ví dụ kết quả của mô hình tốt nhất UPGPT, bên trái sẽ là ảnh sinh, ở giữa là ảnh gốc, bên phải là dáng người được trích xuất từ PHOSA.

1. “Áo của anh ấy có tay ngắn, chất liệu bằng bông và màu trơn. Áo có cổ tròn. Người đó mặc quần dài. Quần được làm bằng bông và có màu trơn.”



2. “Áo của cô ấy có tay ngắn, chất liệu bằng bông và có họa tiết. Cổ áo của nó là cổ tròn. Người phụ nữ này mặc một chiếc quần ngắn. Quần ngắn được làm bằng vải bò và màu trơn. Có phụ kiện trên cổ của cô ấy. Có phụ kiện trên cổ tay của cô ấy. Người phụ nữ này đang đeo nhẫn trên ngón tay của mình.”



3. Người phụ nữ mặc một chiếc áo len dài tay với họa tiết sọc và quần dài. Áo len được làm từ chất liệu bông và có cổ tròn. Quần được làm từ chất liệu vải bò và có màu tron. Người phụ nữ này đội một chiếc mũ. Cô ấy có một phụ kiện trên cổ tay. Người này còn đeo một món trang sức trên cổ. Người phụ nữ này cũng đeo một chiếc nhẫn.



4. Người phụ nữ này đang mặc một chiếc áo không tay với màu tron và một chiếc quần dài. Áo được làm bằng vải bông. Cổ áo tròn. Quần làm bằng vải bò và có màu tron.



TÀI

LIỆU THAM KHẢO

- [1] A. J. P. A. Jonathan Ho, "Denoising diffusion probabilistic models," in *NIPS'20: Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020.
- [2] M. Krichen, "Generative Adversarial Networks," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023.
- [3] D. P. a. W. M. Kingma, "An Introduction to Variational Autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, p. 307–392, 2019.
- [4] A. Gainetdinov, "Diffusion Models vs. GANs vs. VAEs: Comparison of Deep Generative Models," 20 12 2023. [Online]. Available: <https://pub.towardsai.net/diffusion-models-vs-gans-vs-vaes-comparison-of-deep-generative-models-67ab93e0d9ae>.
- [5] C. a. Z. C. a. Z. M. a. K. I. S. Zhang, "Text-to-image Diffusion Models in Generative AI: A Survey," arXiv e-prints, 2023.
- [6] Z. A. X. Y. L. L. B. S. a. H. L. S. Reed, "Generative adversarial text to image synthesis," *International conference on machine learning*, 2016.
- [7] K. a. D. I. a. G. A. a. J. R. D. a. W. D. Gregor, "DRAW: A Recurrent Neural Network For Image Generation," arXiv e-prints, 2015.
- [8] E. P. J. L. B. & R. S. Elman Mansimov, "GENERATING IMAGES FROM CAPTIONS WITH ATTENTION," *ICLR 2016*, 2016.
- [9] H. Z. a. T. X. a. H. L. a. S. Z. a. X. H. a. X. W. a. D. N. Metaxas, "StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative," *CoRR*, 2016.
- [10] P. Z. Q. H. H. Z. Z. G. X. H. X. H. Tao Xu, "AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks," *CVPR*, 2018.
- [11] B. a. Q. X. a. L. T. a. H. P. Li, "Controllable text-to-image generation," arXiv preprint arXiv:1909.07083, 2019.
- [12] A. a. P. M. a. G. G. a. G. S. a. V. C. a. R. A. a. C. M. a. S. I. Ramesh, "Zero-shot text-to-image generation," *International Conference on Machine Learning*, 2021.
- [13] T.-Y. L. a. M. M. a. S. J. B. a. J. H. a. P. P. a. D. R. a. P. D. a. C. L. Zitnick, "Microsoft COCO: Common Objects in Context," *European Conference on Computer Vision*, 2014.

- [14] Y. a. Y. S. a. Q. H. a. W. W. a. L. C. C. a. L. Z. Jiang, "Text2Human: Text-Driven Controllable Human Image Generation," *ACM Transactions on Graphics (TOG)*, vol. 41, pp. 1--11, 2022.
- [15] "Understanding Convolutional Neural Network (CNN): A Complete Guide," [Online]. Available: <https://learnopencv.com/understanding-convolutional-neural-networks-cnn/>.
- [16] "Convolutional Neural Network," [Online]. Available: <https://siddharthsankhe.medium.com/convolutional-neural-network-dc942931bff8>.
- [17] "Illustration of Max Pooling and Average Pooling," [Online]. Available: https://www.researchgate.net/figure/Illustration-of-Max-Pooling-and-Average-Pooling-Figure-2-above-shows-an-example-of-max_fig2_333593451.
- [18] "fully connected layers in convolutional neural networks," [Online]. Available: <https://indiantechwarrior.com/fully-connected-layers-in-convolutional-neural-networks/>.
- [19] O. R. a. P. F. a. T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *CoRR*, 2015.
- [20] "understanding-image-generation-beginner-guide-generative-adversarial-networks-gan," [Online]. Available: <https://blog.ovhcloud.com/understanding-image-generation-beginner-guide-generative-adversarial-networks-gan/>.
- [21] "difference-between-autoencoder-ae-and-variational-autoencoder-vae," [Online]. Available: <https://towardsdatascience.com/difference-between-autoencoder-ae-and-variational-autoencoder-vae-ed7be1c038f2>.
- [22] "diffusion-models-vs-gans-vs-vaes-comparison-of-deep-generative-models," [Online]. Available: <https://pub.towardsai.net/diffusion-models-vs-gans-vs-vaes-comparison-of-deep-generative-models-67ab93e0d9ae>.
- [23] "CLIP of openAI," [Online]. Available: <https://github.com/openai/CLIP>.
- [24] "Multilingual-CLIP," [Online]. Available: <https://github.com/FreddeFrallan/Multilingual-CLIP>.
- [25] "vision-transformer-vit," [Online]. Available: <https://viso.ai/deep-learning/vision-transformer-vit/>.
- [26] "illustrated-stable-diffusion," [Online]. Available: <https://jalammar.github.io/illustrated-stable-diffusion>.
- [27] R. R. a. A. B. a. D. L. a. P. E. a. B. Ommer, "High-Resolution Image Synthesis with Latent Diffusion Models," 2021.
- [28] S. Y. a. M. A. a. G. A. Cheong, "UPGPT: Universal Diffusion Model for Person Image Generation, Editing and Pose Transfer," *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pp. 4173-4182, 2023.
- [29] J. Y. a. P. S. a. J. H. a. R. D. a. M. J. a. K. A. Zhang, "Perceiving 3D Human-Object Spatial Arrangements from a Single Image in the Wild," *European Conference on Computer Vision (ECCV)*, 2020.
- [30] Y. a. Y. S. a. Q. H. a. W. W. a. L. C. C. a. L. Z. Jiang, "Text2Human: Text-Driven Controllable Human Image Generation," *ACM Transactions on Graphics (TOG)*, vol. 41, pp. 1--11, 2022.
- [31] F. a. N. S. a. X. K. a. C. Y. a. L. C. a. S. H. a. Z. J. Bao, "All are Worth Words: A ViT Backbone for Diffusion Models," *CVPR*, 2023.

- [32] Z. a. L. P. a. Q. S. a. W. X. a. T. X. Liu, "DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations," Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
- [33] "What are Diffusion Models?," [Online]. Available: <https://lilianweng.github.io/posts/2021-07-11-diffusion-models/>.