

Trường Đại Học Sư Phạm Kỹ Thuật

Khoa Đào Tạo Chất Lượng Cao

Ngành Công Nghệ Thông Tin



ĐỒ ÁN TỐT NGHIỆP

**ỨNG DỤNG HỌC MÁY CHO BÀI TOÁN PHÁT SINH
ẢNH THỜI TRANG TỪ CÂU MÔ TẢ**

Sinh Viên Thực Hiện	:	NGUYỄN BÁ LÊ AN
MSSV	:	15110001
Sinh Viên Thực Hiện	:	NGUYỄN HỮU KHANG
MSSV	:	15110062
Khoá	:	K15
Ngành	:	CÔNG NGHỆ THÔNG TIN
GVHD	:	TS. NGUYỄN THIÊN BẢO

Tp. Hồ Chí Minh, tháng 07 năm 2020

Đồ án tốt nghiệp

Em xin cam đoan đây là đồ án của riêng nhóm em và được sự hướng dẫn của thầy Nguyễn Thiên Bảo. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào em xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình. Trường đại học Sư Phạm Kỹ Thuật không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 01 tháng 07 năm 2020

Tác giả 1

Nguyễn Bá Lê An

Tác giả 2

Nguyễn Hữu Khang



ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
**KHOA ĐÀO TẠO
CHẤT LƯỢNG CAO**
www.fhq.hcmute.edu.vn

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT
NAM**

Độc lập – Tự do – Hạnh phúc

Tp. Hồ Chí Minh, ngày 01 tháng 07 năm 2020

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ và tên sinh viên: Nguyễn Hữu Khang

MSSV: 15110062

Họ và tên sinh viên: Nguyễn Bá Lê An

MSSV: 15110001

Ngành: Công nghệ thông tin

Khóa: K15

Giảng viên hướng dẫn: TS. Nguyễn Thiên Bảo

Ngày nhận đề tài: 24/02/2020

Ngày nộp đề tài: 01/07/2020

1. Tên đề tài:

- Tìm hiểu bài toán phát sinh ảnh thời trang từ câu mô tả .

2. Các số liệu, tài liệu ban đầu:

- Tập dữ liệu Fashion-Gen.

3. Nội dung thực hiện đề tài:

- Tìm hiểu nghiên cứu tài liệu liên quan đến việc phát sinh ảnh từ mô hình GAN.
- Tìm hiểu các mô hình phát sinh ảnh từ câu mô tả áp dụng AttnGAN (Attention Gan).
- Hiện thực hóa mô hình sử dụng AttnGAN để giải quyết bài toán.
- Kiểm thử và so sánh với mô hình khác (StackGAN-v2) trong việc giải quyết bài toán phát sinh ảnh thời trang từ câu mô tả.

4. Sản phẩm:

- Source code.

TRƯỞNG NGÀNH

GIẢNG VIÊN HƯỚNG DẪN

Th.S Nguyễn Đăng Quang

T.S Nguyễn Thiên Bảo



PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên sinh viên: Nguyễn Hữu Khang

MSSV: 15110062

Họ và tên sinh viên: Nguyễn Bá Lê An

MSSV: 15110001

Ngành: Công nghệ thông tin

Tên đề tài: Tìm hiểu bài toán phát sinh ảnh thời trang từ câu mô tả

Giảng viên hướng dẫn: T.S Nguyễn Thiên Bảo

NHẬN XÉT

1. Về nội dung đề tài và khối lượng thực hiện:

Nhóm đã hoàn thành được các mục tiêu đề ra ban đầu của đề tài, trong khoảng thời gian xác định.

- Về lý thuyết:
 - Hiểu được kiến thức về học máy, học sâu như CNN, RNN, cơ chế Attention
 - Tìm hiểu bài toán phát sinh ảnh thời trang từ câu mô tả.
 - Sinh viên nắm được kiến trúc của mô hình phát sinh ảnh thời trang từ câu mô tả.
- Về thực hành:
 - Sinh viên chạy được demo về bài toán phát sinh ảnh từ câu mô tả dùng GAN với cơ chế Attention.

2. Ưu điểm:

- Hiểu được các lý thuyết về học sâu, trình bày được các cơ sở lý thuyết, toán học một cách chi tiết về mạng nơ-ron nhân tạo.
- Tìm hiểu các mô hình phát sinh ảnh từ câu mô tả.
- Xây dựng mô hình phát sinh ảnh thời trang từ câu mô tả dựa trên các công trình nghiên cứu đã tìm hiểu với hai phương pháp phổ biến là StackGAN-v2 và AttnGAN.
- Tiến hành thực nghiệm trên bộ dữ liệu Fashion-Gen và từ đó có những so sánh.

3. Khuyến điểm:

- Mặc dù hình ảnh được tạo ra với phân giải khá cao song vẫn còn một vài chi tiết chưa được tạo ra rõ ràng, cụ thể thể nhưng trong tương lai mô hình có thể được cải thiện được vấn đề trên.
- Chưa thể tạo ra hình ảnh cụ thể, hoặc hình nền phức tạp.

4. Đề nghị cho bảo vệ hay không?

.....

5. Đánh giá loại:

.....

6. Điểm:.....(Bằng chữ:.....)

Tp. Hồ Chí Minh, ngày 01 tháng 07 năm 2020

Giảng viên hướng dẫn

(Ký & ghi rõ họ tên)

T.S Nguyễn Thiên Bảo



ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
**KHOA ĐÀO TẠO
CHẤT LƯỢNG CAO**
www.fhq.hcmute.edu.vn

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT
NAM**

Độc lập – Tự do – Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

Họ và tên Sinh viên: Nguyễn Hữu Khang

MSSV: 15110062

Họ và tên Sinh viên: Nguyễn Bá Lê An

MSSV: 15110001

Ngành: Công nghệ thông tin

Tên đề tài: Tìm hiểu bài toán phát sinh ảnh thời trang từ câu mô tả

Họ và tên Giảng viên phản biện:.....

NHẬN XÉT

1. Về nội dung đề tài & khối lượng thực hiện :

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

2. Ưu điểm :

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

3. Khuyết điểm :

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

4. Đề nghị cho bảo vệ hay không ?

.....
.....

5. Đánh giá loại :

.....

6. Điểm:.....(Bằng chữ:.....)

Tp. Hồ Chí Minh, ngày tháng 07 năm 2020

Giáo viên phản biện

(Ký & ghi rõ họ tên)

LỜI CẢM ƠN

Nhóm em xin chân thành cảm ơn thầy Nguyễn Thiên Bảo đã tận tình hướng dẫn hỗ trợ nhóm em trong suốt quá trình nghiên cứu về đề tài đồ án này. Ngoài ra, nhóm em xin cảm ơn cô Võ Hoàng Anh, và các anh sinh viên khóa trên đã hỗ trợ chỉnh sửa khắc phục lỗi sai để nhóm em có thể hoàn thành bài đồ án này một cách tốt nhất.

Nhóm em cũng cảm ơn đến tác giả Han Zhang đã tạo ra một công trình mang tính thực tiễn và vô cùng có ích cho những nghiên cứu về trí tuệ nhân tạo và xử lý ảnh số.

Đây là lần đầu chúng em nghiên cứu đề tài này nên không tránh khỏi còn nhiều thiếu sót về nội dung mong quý thầy cô thông cảm bỏ qua và tận tình góp ý.

Chân thành cảm ơn!

TÓM TẮT

Hiện nay, cùng với sự phát triển của linh kiện phần cứng máy tính thì các ứng dụng thị giác máy tính nói riêng và trí tuệ nhân tạo nói chung đang phát triển mạnh mẽ. Khởi tạo ảnh là một trong số những ứng dụng đang phát triển và có tầm ảnh hưởng nhất hiện nay.

Tuy nhiên, ứng dụng này chưa được áp dụng phổ biến trong các lĩnh vực đòi hỏi sự sáng tạo cao như thiết kế thời trang. Cho nên bài nghiên cứu này sẽ tập trung vào việc ứng dụng học máy vào phát sinh ảnh thời trang.

SUMMARY

Nowadays, along with the development of computer hardware components, computer vision applications in particular and artificial intelligence are generally thriving. Initializing photos is one of the most influential and ever-evolving applications.

However, this application has not been widely applied in areas that require high creativity such as fashion design. So this research will focus on the application of machine learning to the creation of fashion photos.

MỤC LỤC

	TRANG
Trang phụ bìa	
Nhiệm vụ đồ án tốt nghiệp.....	i
Trang phiếu nhận xét của giáo viên hướng dẫn.....	ii
Trang phiếu nhận xét của giáo viên phản biện.....	iii
Lời cảm ơn.....	v
Tóm tắt.....	vi
Mục lục	ix
Danh mục các chữ viết tắt	xi
Danh mục các hình ảnh, biểu đồ.....	xii
CHƯƠNG 1.....	1
TỔNG QUAN ĐỀ TÀI	
1.1 Giới thiệu đề tài.....	1
1.2 Phát biểu bài toán.....	2
1.2.1 Mô tả bài toán.....	2
1.2.2 Phát biểu hình thức.....	2
1.3 Khó khăn và thách thức.....	2
1.4 Mục tiêu đề tài.....	3
1.5 Phạm vi đề tài.....	3
CHƯƠNG 2	4
CÔNG TRÌNH NGHIÊN CỨU LIÊN QUAN	
CHƯƠNG 3.....	6
CƠ SỞ LÝ THUYẾT	
3.1 Mạng nơ-ron tích chập (CNN).....	6
3.1.1 Giới thiệu.....	6
3.1.2 Cấu trúc của CNN.....	8
3.2 Mạng nơ-ron hồi quy (RNN).....	10
3.2.1 Giới thiệu.....	10
3.2.2 Mạng LSTM (Bộ nhớ dài ngắn hạn).....	16
3.3 Các kỹ thuật được sử dụng trong đồ án.....	24
3.3.1 Up-sampling.....	24
3.3.2 Down-sampling.....	25

CHƯƠNG 4	2
MÔ HÌNH ĐỀ XUẤT	
4.1 StackGAN-v2	27
4.1.1 Giới thiệu	27
4.1.2 Xấp xỉ phân phối	29
4.1.3 Phân phối ảnh có điều kiện và không điều kiện	30
4.2 AttnGAN	31
4.2.1 Giới thiệu	31
4.2.2 Cấu trúc	32
CHƯƠNG 5	38
KẾT QUẢ NGHIÊN CỨU	
5.1 Tập dữ liệu sử dụng	38
5.2 Tiêu chí đánh giá	40
5.2.1 Inception Score (IS)	40
5.2.2 Fréchet Inception Distance (FID)	41
5.3 Kết quả mô hình phát sinh	42
CHƯƠNG 6	45
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	
6.1 Kết luận	45
6.2 Hướng phát triển	45
TÀI LIỆU THAM KHẢO	47
PHỤ LỤC	49

DANH MỤC CHỮ VIẾT TẮT

CÁC THUẬT NGỮ

GAN	Generative Adversarial Network
AttnGAN	Attentional Generative Adversarial Network
CNN	Convolution Neural Network
RNN	Recurrent Neural Network

Danh mục biểu đồ và hình ảnh

1.1	Mô tả bài toán phát sinh ảnh trang phục từ câu mô tả	2
3.1	Mô hình CNN.....	7
3.2	Tầng maxpooling.....	9
3.3	Mô hình RNN.....	10
3.4	Mô hình one to many.....	12
3.5	Mô hình many to one.....	12
3.6	Mô hình many to many.....	13
3.7	Mô hình many to many khác.....	13
3.8	Cách hoạt động của RNN.....	15
3.9	Sigmoid cho vanishing/exploding gradients.....	16
3.10	Mô hình mạng LSTM.....	17
3.11	Kiến trúc mạng LSTM.....	18
3.12	Kí hiệu trong mạng LSTM.....	18
3.13	Đường truyền trạng thái.....	19
3.14	Cổng LSTM.....	20
3.15	Thông tin đầu vào của LSTM.....	21
3.16	Xử lý thông tin.....	21
3.17	Cập nhật vào Cell State.....	23
3.18	Thông tin đầu ra.....	23
3.19	Cấu trúc của Upsampling.....	24
3.20	Cấu trúc của Down-sampling.....	25
3.21	Cấu trúc của residual block.....	26
4.1	Mô hình StackGAN-v2 được tiểu luận sử dụng cho bài toán phát sinh ảnh thời trang.....	28

4.2	Mô hình xấp xỉ phân phối hình ảnh.....	31
4.3	Mô hình của AttnGAN được tiểu luận sử dụng cho bài toán phát sinh ảnh thời trang.....	32
5.1	Một số mẫu của tập dữ liệu Fashion-gen.....	38
5.2	Thông kê tập dữ liệu theo các loại quần áo.....	39
5.3	Thông kê tập dữ liệu theo tập huấn luyện và kiểm tra.....	40
5.4	Kết quả so sánh giữa ảnh được StackGANv2 và AttnGAN tạo ra với quần và giày.....	43
5.5	Kết quả so sánh giữa ảnh được StackGANv2 và AttnGAN tạo ra với các loại áo thun và áo khoác.....	44

Chương 1

TỔNG QUAN ĐỀ TÀI

1.1 Giới thiệu đề tài

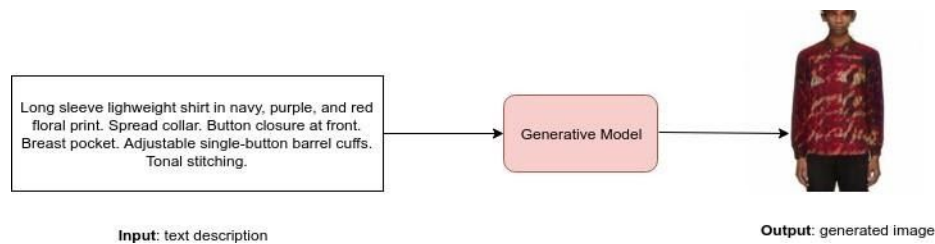
- Ngày nay trước sự bùng nổ của máy tính, trí tuệ nhân tạo cụ thể là thị giác máy tính được ứng dụng vào nhiều lĩnh vực trong đời sống như: nhận dạng khuôn mặt, cảm xúc, nhận dạng đối tượng hay thậm chí còn được ứng dụng trong lĩnh vực yêu cầu độ chính xác cao như: y tế (chẩn đoán các loại bệnh ung thư, xác định vùng bệnh ung thư,...), nông nghiệp,...
- Ngoài ra trong hơn một thập kỷ gần đây Học sâu (Deep learning) trở thành một trong những công cụ phổ biến được ứng dụng vào nhiều bài toán và mang lại hiệu quả đáng kể không những xét về độ chính xác mà còn cả độ hiệu quả trong các ứng dụng thời gian thực. Trước sự phát triển vượt bậc của các giải thuật và sức mạnh của máy tính như hiện nay, những ứng dụng tưởng chừng như không tưởng trước đây đã được đưa vào để giải quyết và có được những kết quả đáng kể. Một trong số đó là việc ứng dụng Học máy vào giải quyết các bài toán liên quan đến thời trang, đây là một trong những hướng tiếp cận mới.
- Một cách cụ thể, trong tiểu luận này các giải thuật Học máy được sử dụng để giải quyết bài toán thiết kế ra một trang phục mới từ một câu mô tả cho trang phục mà qua đó thể hiện mong muốn của khách hàng. Bài toán này mang đến nhiều lợi ích không những xét về ý nghĩa khoa học lẫn ý nghĩa thực tiễn.
- Tuy nhiên, đây là một bài toán đầy thách thức bởi ảnh thời trang có sự đa dạng, phức tạp cả về hình dáng, màu sắc, và chất liệu tương ứng với từng loại trang phục, không những vậy những ảnh hưởng của các đối tượng không liên quan như: khuôn mặt, màu da, tóc, và nền xung quanh cũng tác động không nhỏ đến việc giải quyết bài toán này.

- Bên cạnh đó, từ câu mô tả ảnh trang phục còn cho thấy sự đa dạng và phức tạp về mặt ngữ nghĩa của các từ vựng liên quan đến thời trang. Ngoài ra, việc thể hiện mối liên hệ giữa từ ngữ và hình ảnh trong mô hình cũng cho thấy sự phức tạp của bài toán này.

1.2 Phát biểu bài toán

1.2.1 Mô tả bài toán

Từ câu mô tả về ảnh thời trang cho trước, mô hình sẽ thực hiện xử lý câu mô tả dựa vào các giải thuật Học máy và từ đó phát sinh ra một ảnh thời



trang có nội dung giống với câu mô tả đầu vào.

Hình 1.1: Mô tả bài toán phát sinh ảnh trang phục từ câu mô tả

1.2.2 Phát biểu hình thức

$$\hat{y} \leftarrow f(t) \quad (1.2.1)$$

Trong đó \hat{y} là ảnh phát sinh được tạo ra từ hàm f với t là câu mô tả ảnh đầu vào.

1.3 Khó khăn và thách thức

Một số vấn đề chung thường gặp phải cho bài toán phát sinh ảnh:

- Tốn khá nhiều thời gian cho việc huấn luyện mô hình phát sinh ảnh, và tối ưu tham số cho mô hình vì chưa thật sự có một độ đo nào phù hợp nhất để đo đặc phân bố xác suất của dữ liệu phát sinh và dữ liệu gốc.
- Vấn đề chất lượng hình ảnh tạo ra chưa đạt được mong đợi. Hình ảnh tạo ra chưa đủ chi tiết, phân giải còn khá thấp.
- Hiện tượng Vanishing gradient xảy ra làm cho mô hình khó khăn trong quá trình huấn luyện.
- Hiện tượng Mode collapse dẫn đến tạo ra những hình ảnh giống nhau.

- Một số vấn đề riêng cho bài toán phát sinh ảnh thời trang:
- Các thông tin chi tiết thuộc tính thời trang rất khó thể hiện trong ảnh được phát sinh.
- Các đối tượng ảnh thời trang có hình dạng và cấu trúc ảnh phức tạp.
- Sự phức tạp trong câu mô tả dẫn đến khó khăn trong việc phát sinh ảnh thời trang phù hợp.

1.4 Mục tiêu đề tài

- Đồ án tập trung vào tìm hiểu bài toán sinh ảnh thời trang từ đoạn mô tả cho trước thông qua các giải thuật được sử dụng phổ biến hiện nay. Dựa trên cơ sở lý thuyết tìm hiểu, tiến hành xây dựng, phân tích và lựa chọn mô hình phù hợp cho bài toán phát sinh ảnh thời trang.

1.5 Phạm vi đề tài

- Đồ án tập trung nghiên cứu trên đối tượng ảnh thời trang và các câu mô tả kèm theo tương ứng với từng ảnh, phương pháp được sử dụng dựa vào Học máy với mô hình GAN, để từ đó phát sinh ảnh thời trang mới từ câu mô tả đầu vào. Phạm vi đề tài chỉ giới hạn trên tập ảnh thời trang với nền trắng và chỉ phát sinh một loại quần áo tại một thời điểm.

1.6 Bố cục

Đồ án bao gồm các chương sau:

- CHƯƠNG 1: Tổng quan đề tài
- CHƯƠNG 2: Công trình nghiên cứu liên quan
- CHƯƠNG 3: Cơ sở lý thuyết
- CHƯƠNG 4: Mô hình đề xuất
- CHƯƠNG 5: Kết quả thực nghiệm
- CHƯƠNG 6: Kết luận và hướng phát triển

Chương 2

CÔNG TRÌNH NGHIÊN CỨU LIÊN QUAN

Trước khi đi vào các công trình liên quan, ta sẽ đi qua khái niệm về GAN và các công trình có liên quan tới mô hình kiến trúc mà nhóm em đang nghiên cứu.

Mô hình GAN [3] công bố vào năm 2014. Và đây được xem là nền tảng phát triển cho các công trình GAN khác trong tương lai.

GANs là kiến trúc mạng neural được hình thành trên sự cạnh tranh (adversarial) của 2 mạng nơron khác:

- Generator network (ký hiệu G) mục tiêu sinh ra dữ liệu giả từ không gian tiềm ẩn Z (latent space) sao cho giống với dữ liệu thật nhất.
- Discriminator (ký hiệu D) nhận nhiệm vụ phân biệt dữ liệu được tạo ra từ G và dữ liệu thật cho trước.

Mô hình này được dùng để phát sinh ra hình ảnh sao cho có thể giống với ảnh thật nhất. Điều này giúp ta có thể tạo ra những chữ ký ảo có thể giống với người thật hay tạo ra một vật thể gì đó dựa trên hiểu biết của máy nhưng vẫn hợp với thực tế cuộc sống. Tuy lợi ích là vậy thế nhưng mô hình vẫn còn vướng phải khá nhiều khuyết điểm như việc mô hình trong quá trình huấn luyện không được ổn định, hình ảnh được tạo ra nhưng phân giải còn khá thấp, các vấn đề Vanishing Gradient, vấn đề tạo ra khá nhiều mẫu giống nhau (mode collapse).

Nhờ vào sự có mặt của GAN mà các công trình sau này dựa trên nền tảng này phát triển khá nhiều và một trong số đó phải kể đến là LAPGAN.

Tiểu luận tham khảo bài báo khoa học của Emily L. Denton [2] được công bố 2015, cơ sở dữ liệu là LSUN, CIFAR10 và STL10.

Mục tiêu là tạo ra hình ảnh chất lượng cao từ những hình ảnh dư thừa của những trạng thái trước ứng với từng tầng của kim tự tháp. Mô hình sử dụng kỹ thuật kim tự tháp Laplacian kết hợp với nhiều lớp mạng tích chập (CNNs).

Việc kết hợp này mang lại những ưu điểm sau cho mô hình. Mô hình train không phụ thuộc vào nhau giúp tránh được khó khăn cho mô hình khi ghi nhớ mẫu huấn luyện. Hình ảnh tạo ra có độ chân thật gần như giống với ảnh thật nhất do input ở các đầu vào là ảnh gốc và chỉ làm mờ và downsampling. Tuy nhiên, mô hình vẫn còn bị hạn chế ở việc tạo ra hình ảnh có phân giải thấp. Một vài chi tiết của ảnh chưa được sinh ra rõ ràng, cụ thể Một công trình khác có sự liên quan mật thiết tới mô hình của bài luận nhóm em đang áp dụng là StackGAN. Tiểu luận tham khảo bài báo khoa học của Han Zhang [8] được công bố vào năm 2016. Cơ sở dữ liệu được dùng trong bài báo là MSCOCO, Oxford-102 và CUB.

Mục tiêu tạo ra hình ảnh có độ phân giải cao và chi tiết hơn của một vật thể. Mô hình sử dụng mô hình GAN nhưng được chia thành 2 giai đoạn. Giai đoạn một nhằm khái quát các chi tiết và màu sắc cơ bản của đối tượng. Giai đoạn chia sẻ tập trung khai thác những chi tiết còn thiếu cũng như tăng phân giải của hình đó lên. Nhờ việc ứng dụng kỹ thuật trên mà mô hình tạo ra hình ảnh có độ phân giải cao và chi tiết hơn. Thế nhưng mô hình vẫn còn vướng phải lỗi Mode Collapse, lỗi Vanishing Gradient và vẫn còn hạn chế trong việc hình ảnh tạo ra có thể không khớp với câu mô tả được cấp sẵn.

Mô hình cuối cùng này được nhóm em áp dụng vào bài nghiên cứu là AttnGAN. Tiểu luận tham khảo bài báo khoa học của Tao Xu [7] được công bố vào năm 2017. Cơ sở dữ liệu được dùng trong bài báo là COCO và CUB

Mục tiêu nhằm tạo ra hình ảnh có độ phân giải cao đồng thời các chi tiết của đối tượng cũng sẽ rõ ràng hơn. Mô hình sử dụng mạng khởi tạo tập trung (Attentional Generative Network) và Deep Attentional Multimodal Similarity Model. Nhờ ra đời sau cùng nên mô hình khắc phục gần hết các khuyết điểm mà các công trình phía trên mắc phải. Mô hình đã có thể khắc phục được vấn đề Vanishing Gradient một lỗi vốn phổ biến ở mô hình GAN. Mô hình tạo ra hình ảnh có độ phân giải cao và chi tiết hơn. Mô hình cũng có phân ưu việt khi có thể tạo ra hình ảnh trùng khớp với từng câu mô tả mà mô hình StackGAN-v1.

Chương 3

CƠ SỞ LÝ THUYẾT

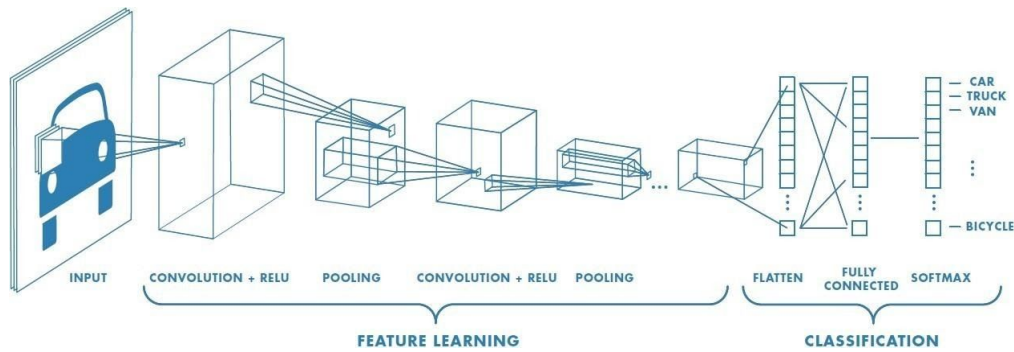
3.1 Mạng nơ-ron tích chập (CNN)

3.1.1 Giới thiệu

- Mạng neural tích chập là một tập hợp gồm nhiều tầng tích chập được xếp chồng lên nhau. Với mục đích nhằm tạo ra những thông tin có tính trừu tượng làm input cho những tầng tích kế tiếp.
- Mạng neural tích chập có ít tham số hơn so với những mạng neural truyền thống. Và một điểm khác biệt nữa ở mạng neural tích chập với mạng neural truyền thống nữa đó chính là cơ chế tích chập. Với những mạng neural truyền thống thì các tầng được liên kết với nhau thông qua 1 tham số W , nhưng ở mạng neural tích chập thì các tầng được liên kết với nhau qua cơ chế tích chập. Cơ chế tích chập này sẽ tạo ra 1 output là đặc trưng được rút ra ở tầng này và sẽ là input cho tầng tiếp theo sử dụng. Bên cạnh đó mạng neural tích chập còn có sử dụng tầng pooling. Tầng này có nhiệm vụ sẽ làm giảm số chiều của đặc trưng được rút ra, sẽ chắt lọc lại những thông tin có ích cho những tầng sau. Mạng neural tích chập có tính bất biến và tính kết hợp cục bộ. Nếu ta xét trên một đối tượng, độ chính xác của hình sẽ bị ảnh hưởng tùy thuộc vào góc độ chụp của hình đó. Tính bất biến của hình sẽ biểu hiện ở quá trình pooling qua các phép tính dịch chuyển, phép quay, phép co giãn.
- Tính kết hợp cục bộ cho ta các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn thông qua quá trình tích chập từ các bộ lọc là một trong những mô hình Deep Learning tiên tiến giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao. Ví dụ như diện khuôn mặt người dùng, phát triển xe hơi tự lái hay drone giao hàng tự động,... CNN được sử dụng nhiều trong các bài toán

nhận dạng các đối tượng trong ảnh. Mạng tích chập bao gồm một hay nhiều tầng tích chập, dùng để trích xuất thông tin cho các tầng tiếp theo.

Hình 3.1: Mô hình CNN



Nguồn: <https://towardsdatascience.com>

3.1.2 Định nghĩa tích chập

Tích chập có thể được xem như một cửa sổ trượt trên ma trận của hình ảnh. Việc tích chập sẽ làm giảm kích thước của ma trận xuống nhưng sẽ rút trích được những đặc trưng cơ bản của ma trận hình đó

Cách tính: 1 phép toán thực hiện nhân tích chập ma trận của ảnh với filter / mask / kernel (bộ lọc) để được ma trận điểm ảnh mới:

$$g(x, y) = h(x, y) * f(x, y) \quad (3.1.1)$$

Trong đó:

- f, g : input/output
- h :

mask/filter/kernel

Cách tính:

- Trượt filter trên ma trận ảnh.
- Nhân các phần tử tương ứng và sau đó tổng chúng lại với nhau.
- Lặp lại quy trình này cho đến khi tất cả các giá trị của hình ảnh được tính

3.1.3 Cấu trúc của CNN

Mạng CNN gồm rất nhiều lớp chồng lên và có 3 loại chính :

Lớp Convolution

- Đây là một trong những lớp quan trọng và cần thiết nhất trong CNN. Lớp convolution có vai trò rút ra những đặc trưng của đối tượng. Và là đầu vào của những lớp tiếp theo.
- Lớp Convolution sử dụng các hàm kích hoạt phi tuyến như ReLU và tanh để kích hoạt các trọng số trong các node. Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo. Các layer liên kết được với nhau thông qua cơ chế convolution. Layer tiếp theo là kết quả convolution từ layer trước đó, nhờ vậy mà ta có được các kết nối cục bộ. Như vậy mỗi neural ở lớp kế tiếp sinh ra từ kết quả của filter đặt lên một vùng ảnh cục bộ của neural trước đó.

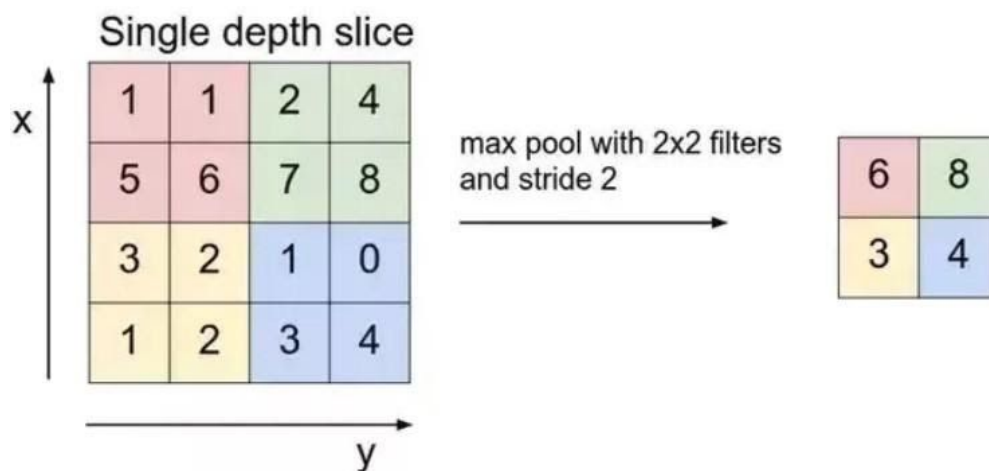
Lớp Pooling

Pooling/subsampling layer (Tầng tổng hợp) dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu). Việc sử dụng lớp pooling như vậy sẽ giúp làm giảm số chiều của ảnh. Trong quá trình huấn luyện mạng (training) CNN tự động học các giá trị qua các lớp filter dựa vào cách thức mà bạn thực hiện. Ví dụ trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra thông số tối ưu cho các filter tương ứng theo thứ tự raw pixel > edges > shapes > facial > high-level features

Có rất nhiều cách để tổng hợp, chẳng hạn như lấy trung bình hoặc cực đại. Thủ tục pooling được dùng nhiều nhất là max pooling.

Cách tính: Ta sẽ cho trượt không tuyến tính của sổ (filter) trên ảnh. Và ta sẽ chọn ra giá trị lớn nhất trong cửa sổ đó. Lặp lại cho tới khi các giá trị được tính.

Ví dụ:



Hình 3.2: Tầng maxpooling
Nguồn: <https://www.quora.com>

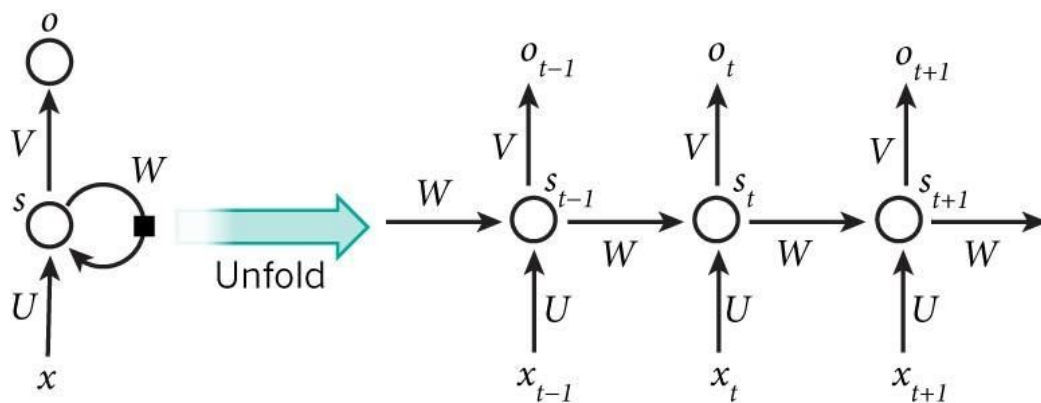
Lớp Fully connected

- Fully connected: Layer cuối cùng tổng hợp các kết quả từ quá trình convolution và subsampling.

3.2 Mạng nơ-ron hồi quy (RNN)

3.2.1 Giới thiệu

Là mạng hồi quy mang lại thành tựu to lớn trong ngành Deep learning nói chung cũng như là xử lý ngôn ngữ tự nhiên. Mạng này được gọi là hồi quy do mạng này xử lý thông tin theo dạng chuỗi. Như là xử lý cho 1 câu gồm nhiều từ hay một văn bản gồm nhiều câu. Và kết quả đầu ra ở một thời điểm i sẽ phụ thuộc vào việc tính toán dữ liệu của thời điểm $i-1$ trước đó. Nói cách khác, RNN là một mô hình có trí nhớ (memory), có khả năng nhớ được thông tin đã tính toán trước đó. Không như các mô hình mạng neural truyền thống đó là thông tin đầu vào (input) hoàn toàn độc lập với thông tin đầu ra (output). Về lý thuyết, RNNs có thể nhớ được thông tin của chuỗi có chiều dài bất kỳ, nhưng trong thực tế mô hình này chỉ nhớ được thông tin ở vài bước trước đó.



Hình 3.3: Mô hình RNN

Nguồn: <http://www.wildml.com>

Từ mô hình trên ta có:

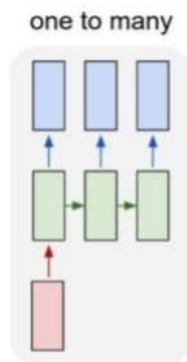
- x_t là input tại thời điểm thứ t . Và x_t ở đây sẽ được biểu diễn dưới dạng một one-hot vector tương ứng với 1 từ trong câu
- s_t là hidden state (memory) tại thời điểm thứ t . s_t được tính dựa trên các hidden state trước đó kết hợp với input của thời điểm hiện tại tại $s_t = f(Ux_t + Ws_{t-1})$.
- Hàm f là hàm phi tuyến tính thường là hàm tanh, ReLU. s_{t-1} là hidden state được khởi tạo là một vector không.
- o_t là output tại thời điểm thứ t . o_t là một vector chứa xác suất của toàn bộ các từ trong từ điển và được tính như sau $o_t = \text{softmax}(V s_t)$.

Từ mô hình trên ta có thể thấy các thành phần trong chuỗi có tính phụ thuộc lẫn nhau. Ví dụ nếu ta xét 1 câu gồm 7 chữ thì ứng với một chữ trong câu sẽ là một lớp mạng được dàn trải ra. Và ở mô hình RNN này thì bộ tham số được dùng là (U,V,W) sẽ được sử dụng cho toàn bộ quá trình huấn luyện.

Các dạng RNN

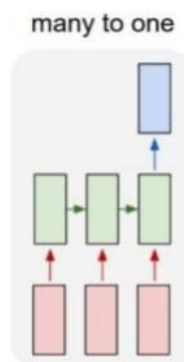
RNN có 4 dạng được sử dụng phổ biến là:

Dạng one – to – many: Từ 1 input cho ra nhiều output. Dùng để chú thích, mô tả hình ảnh. Thường thì người ta sẽ sử dụng mạng CNN để detect object trong ảnh và sau đó sẽ dùng RNN để sinh ra các câu có nghĩa để mô tả cho bức ảnh đó.



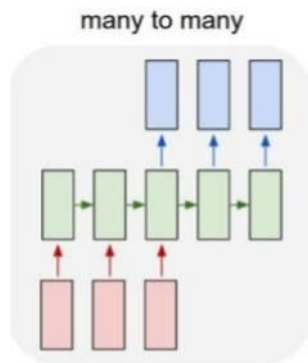
Hình 3.4: Mô hình one to many
 Nguồn: <https://viblo.asia>

Dạng many – to – one: Từ nhiều input cho ra 1 output. Dùng cho việc tính toán ra kết quả cuối cùng cho nhiều dữ liệu đầu vào



Hình 3.5: Mô hình many to one
 Nguồn: <https://viblo.asia>

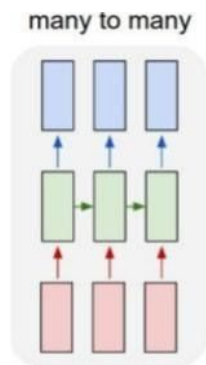
Dạng many – to – many: Dùng cho dịch thuật. Từ nhiều đầu vào cho ra nhiều đầu ra.



Hình 3.6: Mô hình many to many

Nguồn: <https://viblo.asia>

Một dạng khác của many – to – many: Dùng cho phân biệt video trên từng khung hình.



Hình 3.7: Mô hình many to many khác

Nguồn: <https://viblo.asia>

Công thức:

$$S_t = F_w(S_{t-1}, X_t)(1)$$

Trong đó:

X_t : Input tại thời điểm t

S_t : Trạng thái tại thời điểm

F_w : Hàm đệ quy với tham số W

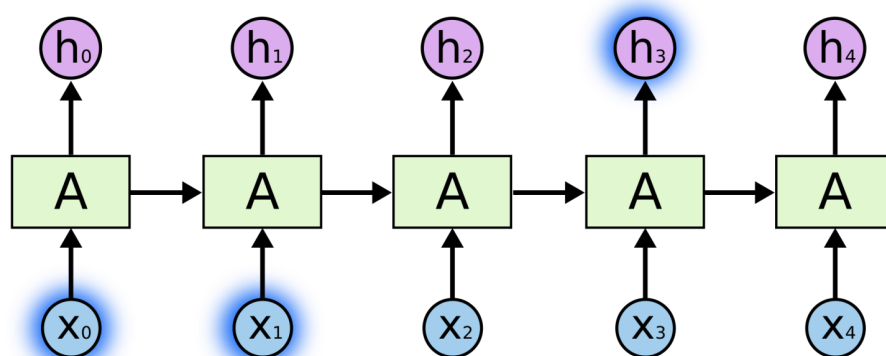
Ví dụ mạng RNN đơn giản:

Từ công thức (1) ta có trạng thái S_t tại thời điểm t được tính theo công thức đệ quy với hàm kích hoạt \tanh của tổng các tích của trạng thái đầu vào S_{t-1}

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t) \quad y_t = W_{hy}h_t$$

Không như với mạng neural truyền thống, chỉ sử dụng duy nhất một ma trận trọng số W để tính toán thì với RNN nó sử dụng 3 ma trận trọng số cho 2 quá trình tính toán. Đầu tiên ta sẽ tính \tanh của tổng 2 giá trị tại thời điểm h_{t-1} với thời điểm h_t . Sau đó lấy tích giá trị h_t với W_{hy} để tính ra kết quả y_t .

Mặc dù mạng RNN được giới khoa học kỳ vọng là mô hình mạng có thể giải quyết các vấn đề về phụ thuộc xa (long-term dependencies). Nhưng thực tế thì mô hình này có thực sự đúng như kỳ vọng của họ. Câu trả lời là không hạn.



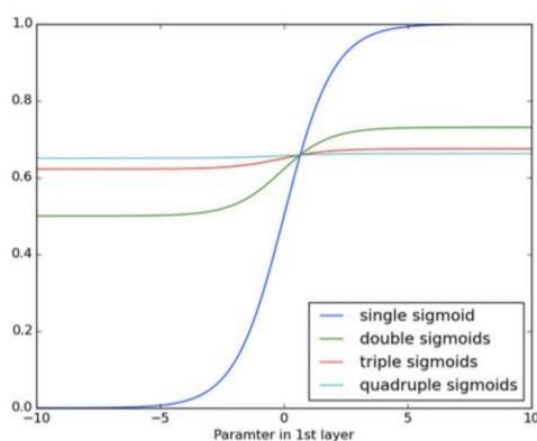
Hình 3.8: Cách hoạt động của RNN

Nguồn: <https://dominhhai.github.io>

Từ hình minh họa trên, trong mô hình hóa ngôn ngữ, chúng ta cố gắng dự đoán từ tiếp theo dựa vào các từ trước đó. Nếu chúng ta có câu “đám mây bay trên bầu trời”, thì chúng ta không cần xét quá nhiều từ trước đó, chỉ cần đọc tới “đám mây trên bầu” là đủ biết được chữ tiếp theo là “trời” rồi. Trong trường hợp này, khoảng cách tới thông tin liên quan được rút ngắn lại, mạng RNN có thể học và sử dụng

các thông tin quá khứ. Nhưng cũng với trường hợp này nếu ta xét với 1 câu dài hơn, nhiều thông tin hơn, nghĩa phụ thuộc vào ngữ cảnh. Ví dụ như ta dự đoán từ cuối cùng của 1 văn bản sau “I grew up in France... I speak fluent French.” Nếu như chỉ đọc “I speak fluent French” thì ta chỉ có thể dự đoán được tên ngôn ngữ chứ không thể xác định được chính xác đó là ngôn ngữ gì. Nếu muốn biết chính xác đó là ngôn ngữ gì thì ta cần phải xét luôn ngữ cảnh “I grew up in France” thì mới có thể suy luận được. Từ đây ta có thể thấy khoảng cách thông tin đã khá xa khiến cho việc dự đoán trở nên khó khăn và nhọc nhằn hơn và bên cạnh đó có thể kết quả trả ra sẽ sai sót. Việc thiếu sót của mạng RNN mang lại đã được 2 nhà bác học “Hochreiter (1991) [German] and Bengio, et al. (1994)” đưa ra và đã trở thành một nỗi lo lắng cho giới khoa học trong thời điểm nóng bỏng này.

Nhưng quan trọng hơn, 2 nhà bác học trên đã chỉ ra 2 yếu tố khiến cho mạng RNN không giải quyết được chính xác vấn đề “long term dependencies” là Vanishing và Exploding Gradients. Và 2 yếu tố này thường sẽ xuất hiện nhiều trong quá trình huấn luyện mô hình. Vanishing gradients chỉ xảy ra khi gradient signal ngày càng nhỏ theo quá trình huấn luyện, khiến cho quá trình tối thiểu hóa hàm lỗi hội tụ chậm hoặc dừng hẳn Exploding gradients chỉ xảy ra khi gradient signal ngày càng bị phân tán trong quá trình huấn luyện, khi đó quá trình tối thiểu hoá hàm lỗi không hội tụ.



Hình 3.9: Sigmoid cho vanishing/exploding gradients

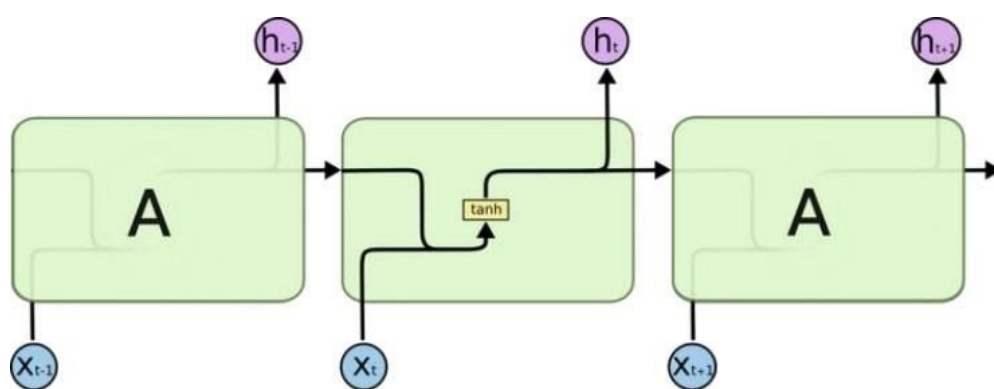
Nguồn: <https://dominhhai.github.io>

Và để giải quyết 2 vấn đề trên, mạng LSTM đã ra đời.

3.2.2 Mạng LSTM (Bộ nhớ dài ngắn hạn)

Giới thiệu

Mạng bộ nhớ dài-ngắn (Long Short Term Memory networks), thường được gọi là LSTM - là một dạng đặc biệt được cải tiến từ RNN, nó có khả năng học được các phụ thuộc xa. LSTM được giới thiệu bởi Hochreiter & Schmidhuber (1997), và sau đó đã được cải tiến và phổ biến bởi rất nhiều người trong ngành. LSTM được thiết kế để tránh được vấn đề phụ thuộc xa (long-term dependency). Việc nhớ thông tin trong suốt thời gian dài là đặc tính cơ bản của chúng, chứ ta không cần phải huấn luyện nó để có thể nhớ được. Tức là ngay nội tại của nó đã có thể ghi nhớ được mà không cần bất kỳ can thiệp nào. Mọi mạng hồi quy đều có dạng là một chuỗi các mô-đun lặp đi lặp lại của mạng neural. Với mạng RNN chuẩn, các modun này có cấu trúc rất đơn giản, thường là một tầng tanh.

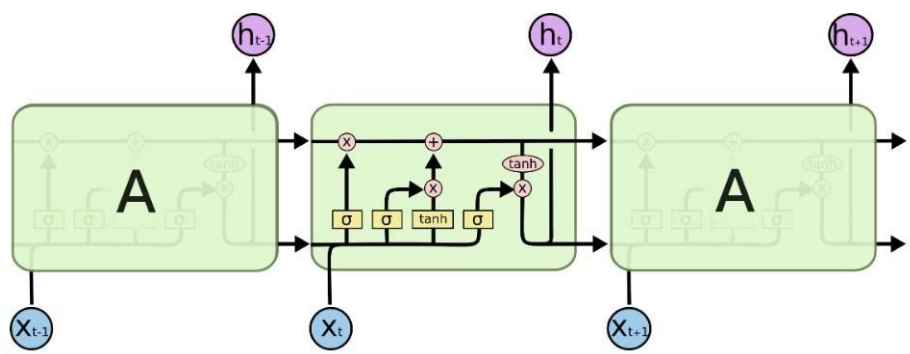


Hình 3.10: Mô hình mạng LSTM

Nguồn: <https://dominhhai.github.io>

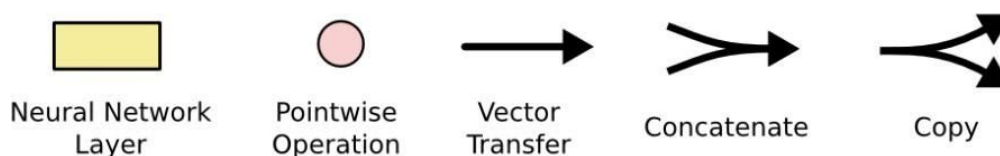
Cách hoạt động

Mạng LSTM là một biến thể được cải tiến từ chính RNN nên cũng mang kiến trúc dạng chuỗi như RNN. Nhưng chỉ khác so với RNN đó chính là có 4 tầng tương tác với nhau thay vì có 1 như RNN.



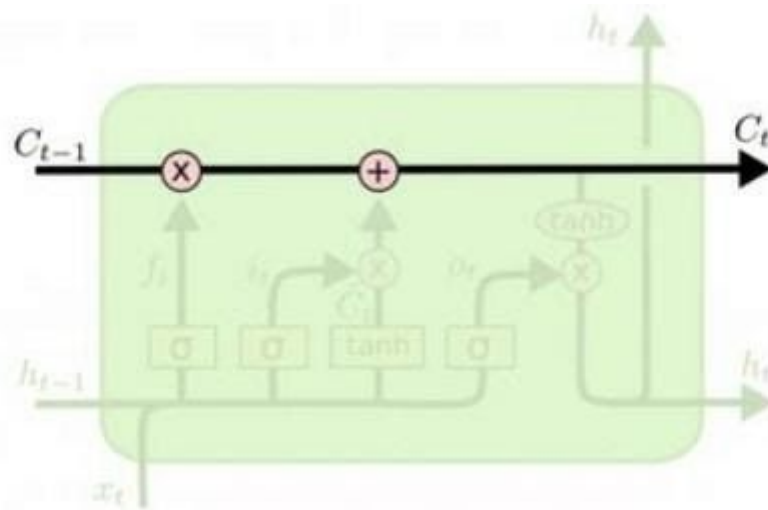
Hình 3.11: Kiến trúc mạng LSTM
 Nguồn: <https://dominhhai.github.io>

Trong đó sẽ có những ký hiệu sau:



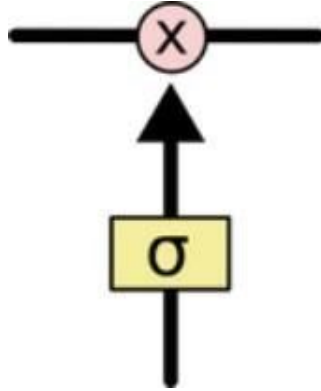
Hình 3.12: Ký hiệu trong mạng LSTM
<https://dominhhai.github.io>

Các ký hiệu trong hình trên theo thứ tự từ trái sang phải được giải thích như sau. Hình chữ nhật vàng là Lớp mạng neural. Hình tròn màu hồng là phép toán vector. Các dấu mũi tên hợp nhau thể hiện việc kết hợp, còn các dấu mũi tên rẽ nhánh thể hiện nội dung của nó được sao chép và chuyển tới các nơi khác nhau. Mô hình thiết kế của LSTM là một bảng mạch số, gồm các mạch logic và các phép toán logic trên đó. Thông tin, hay nói khác hơn là tần số của dòng điện di chuyển trong mạch sẽ được lưu trữ, lan truyền theo cách thiết kế bảng mạch. Mấu chốt của LSTM là cell state (trạng thái nhớ), đường kẻ ngang chạy dọc ở trên top diagram. Cell state giống như băng chuyền, chạy xuyên thẳng toàn bộ mắt xích, chỉ một vài tương tác nhỏ tuyến tính (minor linear interaction) được thực hiện. Điều này giúp cho thông tin ít bị thay đổi xuyên suốt quá trình lan truyền.



Hình 3.13: Đường truyền trạng thái
 Nguồn: <https://dominhhai.github.io>

LSTM có khả năng thêm hoặc bớt thông tin vào cell state, được quy định một cách cẩn thận bởi các cấu trúc gọi là cổng (gate). Các cổng này là một cách (tùy chọn) để định nghĩa thông tin bằng qua. Chúng được tạo bởi hàm sigmoid và một toán tử nhân pointwise.

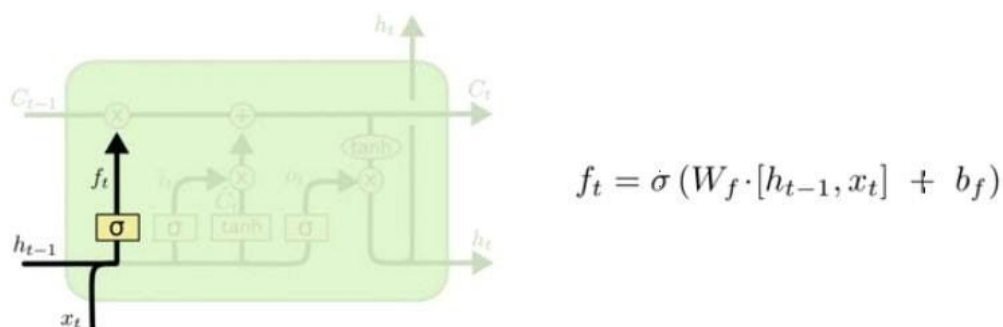


Hình 3.14: Cổng LSTM

Nguồn: <https://dominhhai.github.io>

Hàm kích hoạt Sigmoid có giá trị từ 0 – 1, mô tả độ lớn thông tin được phép truyền qua tại mỗi lớp mạng. Nếu ta thu được zero điều này có nghĩa là “không cho bất kỳ cái gì đi qua”, ngược lại nếu thu được giá trị là một thì có nghĩa là “cho phép mọi thứ đi qua”. Một LSTM có ba cổng như vậy để bảo vệ và điều khiển cell state.

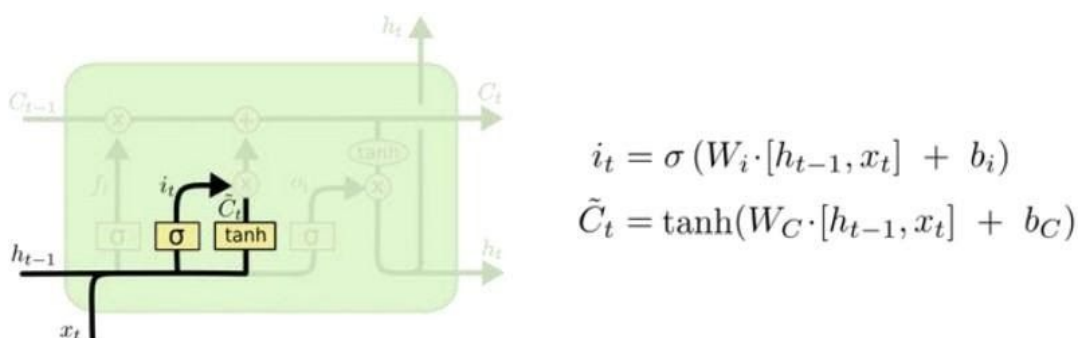
Quá trình hoạt động của LSTM được thông qua các bước cơ bản sau. Bước đầu tiên của mô hình LSTM là quyết định xem thông tin nào chúng ta cần loại bỏ khỏi cell state. Tiến trình này được thực hiện thông qua một sigmoid layer gọi là “forget gate layer” - cổng. Đầu vào là h_{t-1} và x , đầu ra là một giá trị nằm trong khoảng $[0, 1]$ cho cell state C_{t-1} . 1 tương đương với “giữ lại thông tin”, 0 tương đương với “loại bỏ thông tin”.



Hình 3.15: Thông tin đầu vào của LSTM

Nguồn: <https://dominhhai.github.io>

Bước tiếp theo, cần quyết định thông tin nào cần được lưu lại tại cell state. Ta có hai phần là single sigmoid layer được gọi là “input gate layer” quyết định các giá trị chúng ta sẽ cập nhật. Tiếp theo, một tầng layer tạo ra một vector ứng viên mới \tilde{C}_t được thêm vào trong cell state.

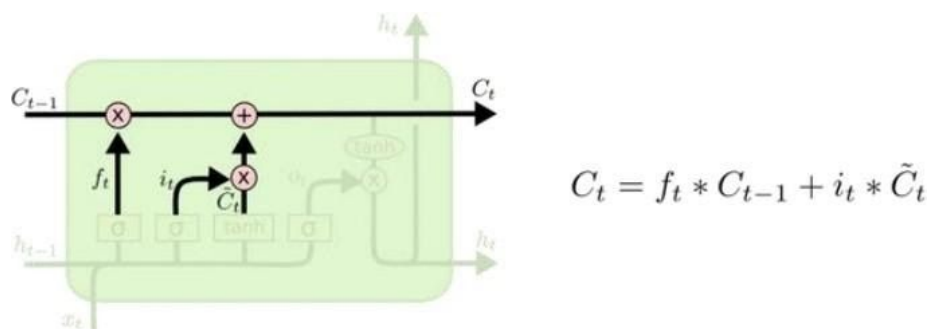


Hình 3.16: Xử lý thông tin

Nguồn: <https://dominhhai.github.io>

Ở bước tiếp theo, sẽ kết hợp hai thành phần này lại để cập nhật vào cell state. Lúc cập nhật vào cell state cũ, C_{t-1} , vào cell state mới C_t . Ta sẽ đưa state của hàm f , để quên đi những gì trước đó. Sau đó, ta sẽ thêm $i_t * \tilde{C}_t$. Đây là giá

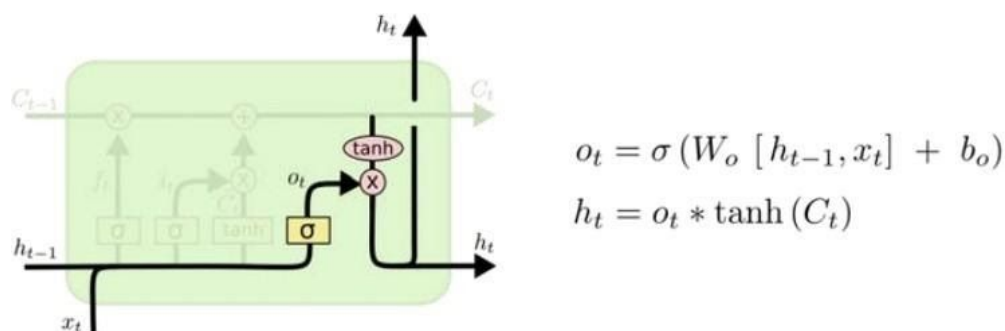
trị ứng viên mới, co giãn (scale) số lượng giá trị mà ta muốn cập nhật cho mỗi state.



Hình 3.17: Cập nhật vào Cell State

Nguồn: <https://dominhhai.github.io>

Cuối cùng, cần quyết định xem thông tin output là gì. Output này cần dựa trên cell state, nhưng sẽ được lọc bớt thông tin. Đầu tiên, áp dụng single sigmoid layer để quyết định xem phần nào của cell state chúng ta dự định sẽ output. Sau đó, ta sẽ đẩy cell state qua tanh (đẩy giá trị vào khoảng -1 và 1) và nhân với một output sigmoid gate, để giữ lại những phần ta muốn output ra ngoài.



Hình 3.18: Thông tin đầu ra

Nguồn: <https://dominhhai.github.io>

Mô hình LSTM là một bước đột phá đạt được từ mô hình RNN. Nó giải quyết triệt để vấn đề không xử lý được câu hỏi dài mà những mô hình như chatbox Skype đang gặp phải.

3.3 Các kỹ thuật được sử dụng trong đồ án

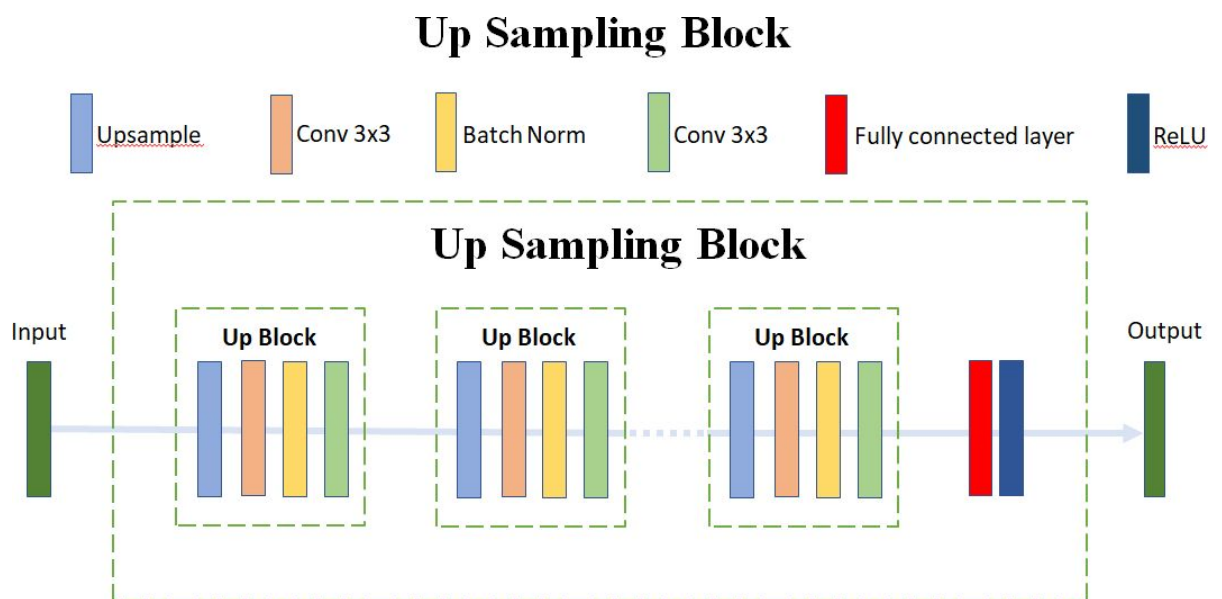
3.3.1 Upsampling

Đây là một khối giúp tăng phân giải kích thước của một ảnh. Và là một khối quan trọng trong giai đoạn Generate.

Upsampling sử dụng các lớp tương tự như mạng tích chập CNNs. Trong khối upsampling sẽ gồm các khối Unblock nhỏ hơn. Mỗi một khối unblock sẽ có các tầng như sau:

- Tầng Upsample
- 2 Tầng tích chập có kích thước 3x3
- Tầng BatchNorm

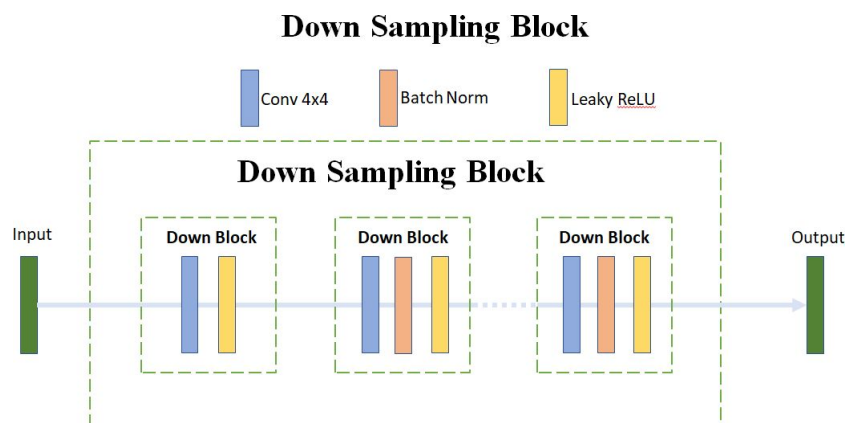
Và hai lớp cuối cùng của khối upsampling là lớp Fully Connected và lớp ReLU dùng để tạo ra hình ảnh.



Hình 3.19: Cấu trúc của Upsampling

3.3.2 Down-sampling

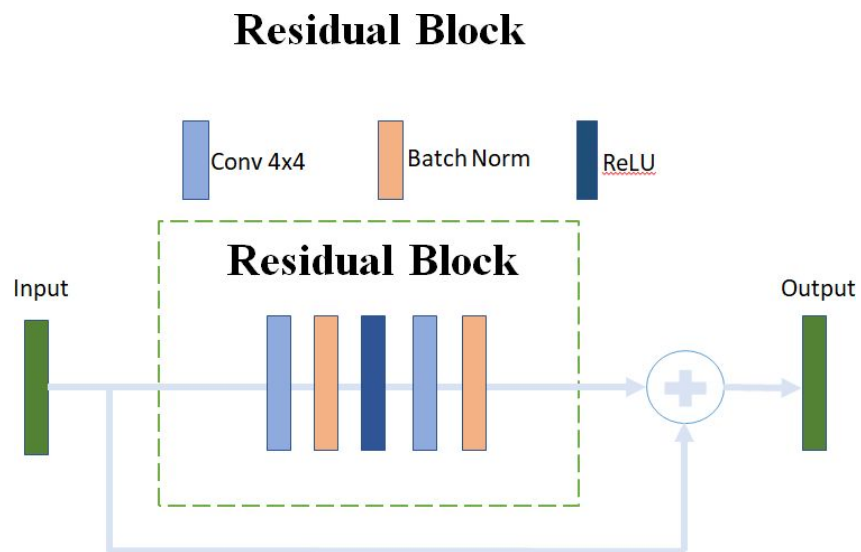
- Đây là một khối giúp giảm phân giải của hình ảnh. Đây là một khối chính trong giai đoạn Down-sampling.
- Down-sampling gồm một chuỗi các mạng tích chập 4x4, sau mỗi mạng tích chập là lớp Batch Norm (ngoại trừ mạng tích chập đầu tiên) và LeakyReLU



Hình 3.20: Cấu trúc của Down-sampling

3.3.3 Residual block

- Residual block gồm có các lớp tích chập 4x4 kết hợp với lớp batch norm và một lớp ReLU



Hình 3.21: Cấu trúc của Residual block

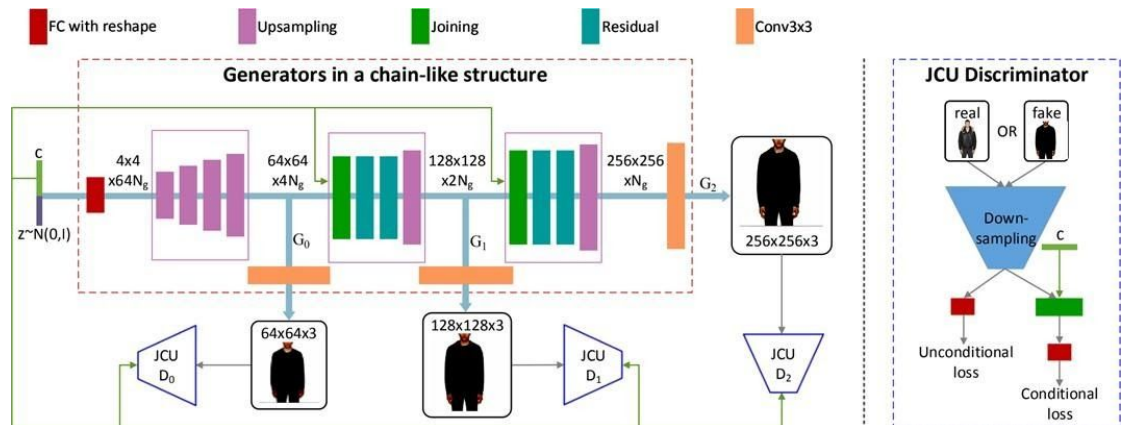
Chương 4

MÔ HÌNH ĐỀ XUẤT

4.1 StackGAN-v2

4.1.1 Giới thiệu

Khác với mô hình StackGAN-v1 [8] được chia làm 2 mạng riêng biệt là GAN giai đoạn 1 được sử dụng để phát sinh cấu trúc ảnh có độ phân giải thấp và GAN giai đoạn 2 được sử dụng để phát thông tin chi tiết ảnh ở mức cao thì StackGAN-v2 [9] sẽ bao gồm nhiều mạng phát sinh (Gs) và phân biệt (Ds) ở cấu trúc dạng cây. Ảnh sẽ được khởi tạo với độ phân giải từ thấp đến cao ở những nhánh khác nhau của cây. Tại mỗi nhánh, mạng phát sinh sẽ được huấn luyện để khởi tạo ảnh sinh ra ở một tỉ lệ nhất định và mạng phân biệt sẽ được huấn luyện để phân biệt ảnh thật và giả ở tỉ lệ đó. Những mạng phát sinh cũng được huấn luyện cùng lúc để xấp xỉ đa phân phối, và những mạng phát sinh và phân biệt sẽ được huấn luyện xen kẽ với nhau.



Hình 4.1: Mô hình StackGAN-v2 được tiểu luận sử dụng cho bài toán phát sinh ảnh thời trang

Có 2 loại đa phân phối:

- Phân phối của ảnh ở những tỉ lệ khác nhau.
- Phân phối của ảnh có điều kiện và không có điều kiện.

4.1.2 Xấp xỉ phân phối

StackGAN-v2 có cấu trúc dạng cây, nó sẽ nhận những vector nhiễu $z \sim p_{noise}$ làm đầu vào và những mạng phát sinh sẽ sinh ra các hình ảnh ở những tỉ lệ khác nhau. Trong đó p_{noise} là một phân phối xác suất cho trước và thường được chọn là phân phối chuẩn. Biến ẩn z được chuyển đổi thành các đặc trưng ẩn thông qua các tầng. Đặc trưng ẩn h_i cho bộ phát sinh G_i được tính bằng cách chuyển đổi tuyến tính sau:

$$h_0 = F_0(z); h_i = F_i(h_{i-1}, z), i = 1, 2, \dots, m-1 \quad (4.1.1)$$

Trong đó, h_i là đặc trưng ẩn của nhánh i^{th} , m là tổng số nhánh, và F_i là mô hình mạng nơron. Để có thể học được những thông tin bỏ sót ở nhánh trước đó, vector nhiễu z được nối với đặc trưng ẩn h_{i-1} để trở thành đầu vào cho mạng F_i nhằm mục đích tính đặc trưng h_i . Dựa vào đặc trưng ẩn ở các tầng khác nhau (h_0, h_1, \dots, h_{m-1}), mạng phát sinh sẽ sinh ra ảnh có tỉ lệ (độ phân giải) từ thấp đến cao (s_0, s_1, \dots, s_{m-1}).

$$s_i = G_i(h_i), i = 0, 1, \dots, m-1 \quad (4.1.2)$$

Trong đó, G_i là mạng phát sinh tại nhánh i^{th} . Theo sau mỗi mạng phát sinh G_i là một mạng phân biệt D_i , mạng D_i sẽ nhận đầu vào là ảnh thật x_i hoặc ảnh s_i được sinh ra bởi G_i , sau đó D_i sẽ được huấn luyện để phân loại ảnh thật hoặc phát sinh bằng cách tối thiểu hàm *cross-entropy loss* sau:

$$LG = \sum_{i=1}^m L_{Gi} \quad (4.1.3)$$

$$L_{Gi} = -E_{s_i \sim p_{Gi}} [\log D_i(s_i)]$$

Trong đó, L_{Gi} là hàm mất mát cho việc xấp xỉ phân phối của ảnh thật tại tỉ lệ của nhánh i^{th} . Trong suốt quá trình huấn luyện, mạng phân biệt D_i và mạng phát sinh G_i sẽ được tối ưu một cách xen kẽ nhau cho đến khi chúng hội tụ. Việc khởi tạo ảnh ở những tỉ lệ khác nhau có thể khiến cho việc huấn luyện toàn bộ mạng StackGAN-v2 được ổn định và khởi tạo ra ảnh có độ phân giải cao được chi tiết hơn. Thật vậy, sau khi khởi tạo ảnh tỉ lệ (độ phân giải) thấp với những hình ảnh và màu sắc cơ bản ở nhánh đầu

tiên của cấu trúc cây, những mảng phát sinh ở những nhánh tiếp theo sẽ chỉ quan tâm đến việc bổ sung và hoàn thành các chi tiết còn thiếu để khởi tạo hình ảnh có độ phân giải cao hơn.

4.1.3 Phân phối ảnh có điều kiện và không điều kiện

Đối với ảnh được khởi tạo không có điều kiện, mạng phân biệt của StackGAN-v2 sẽ được huấn luyện để phân biệt xem ảnh là thật hay phát sinh, điều này sẽ giúp cho mảng phân biệt xấp xỉ được với phân phối của ảnh không có điều kiện. Đối với ảnh được khởi tạo có điều kiện, ảnh và biến điều kiện tương ứng sẽ là đầu vào của mạng phân biệt để xác định xem tập ảnh và đoạn văn mô tả của ảnh có khớp với nhau hay không, điều này sẽ giúp cho mạng phân biệt xấp xỉ được với phân phối của ảnh có điều kiện. Đối với việc khởi tạo ảnh có điều kiện, mảng F_0 và F_i của các mảng phát sinh sẽ được chuyển đổi để có thể nhận thêm vector điều kiện c làm đầu vào như sau $h_0 = F_0(c, z)$ và $h_i = F_i(h_{i-1}, c)$. Trong mảng F_i , vector điều kiện c sẽ thay thế vector nhiễu z để làm cho mảng phát sinh có thể sinh ra ảnh có nhiều chi tiết hơn dựa vào đoạn văn mô tả cho trước. Do đó, những ảnh có tỉ lệ khác nhau được khởi tạo bởi G_i sẽ như sau $s_i = G_i(h_i)$ và hàm mục tiêu của việc huấn luyện phân biệt D_i sẽ cho việc phân biệt ảnh có điều kiện sẽ bao gồm 2 phần là *hàm mất mát có điều kiện (conditional loss)* và *hàm mất mát không có điều kiện (unconditional loss)*.

$$L_{D^i} = - \frac{1}{2} \mathbb{E}_{x^i \sim p_{data_i}} [\log D_i(x_i)] - \frac{1}{2} \mathbb{E}_{s^i \sim p_{G_i}} [\log(1 - D_i(s_i))] +$$

$$\frac{1}{2} \mathbb{E}_{x^i \sim p_{data_i}} [\log D_i(x_i, c)] - \frac{1}{2} \mathbb{E}_{s^i \sim p_{G_i}} [\log(1 - D_i(s_i, c))]$$

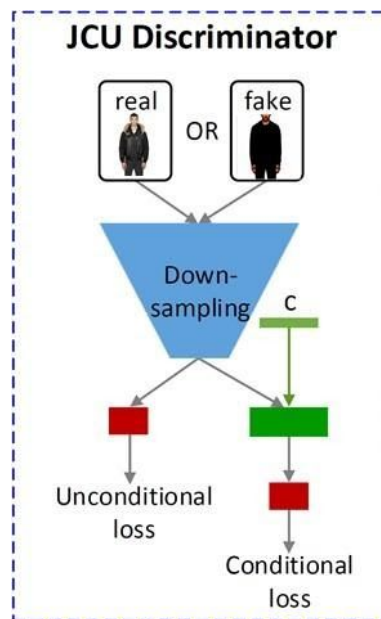
(4.1.4)

Hàm mất mát không điều kiện sẽ giúp xác định rằng ảnh là ảnh thật hay giả, hàm mất mát có điều kiện sẽ giúp xác định ảnh được khởi tạo với đoạn văn bản mô tả có trùng khớp với nhau hay không.

Hàm mất mát của các mạng phát sinh G_i sẽ được chuyển đổi thành như sau:

$$L_{G^i} = - \underbrace{2 \mathop{\mathbb{E}}_{s_i \sim p_{G^i}} [\log(1 - D_i(s_i))] + \mathop{\mathbb{E}}_{s_i \sim p_{G^i}} [\log(1 - D_i(s_i, c))] + \mathop{\mathbb{E}}_{s_i \sim p_{G^i}} [\log(1 - D_i(s_i, c))] + \mathop{\mathbb{E}}_{s_i \sim p_{G^i}} [\log(1 - D_i(s_i, c))]}_{\text{conditional loss}} \quad (4.1.5)$$

Các mạng phát sinh G_i sẽ đồng thời xấp xỉ phân phối của ảnh có điều kiện và không có điều kiện.



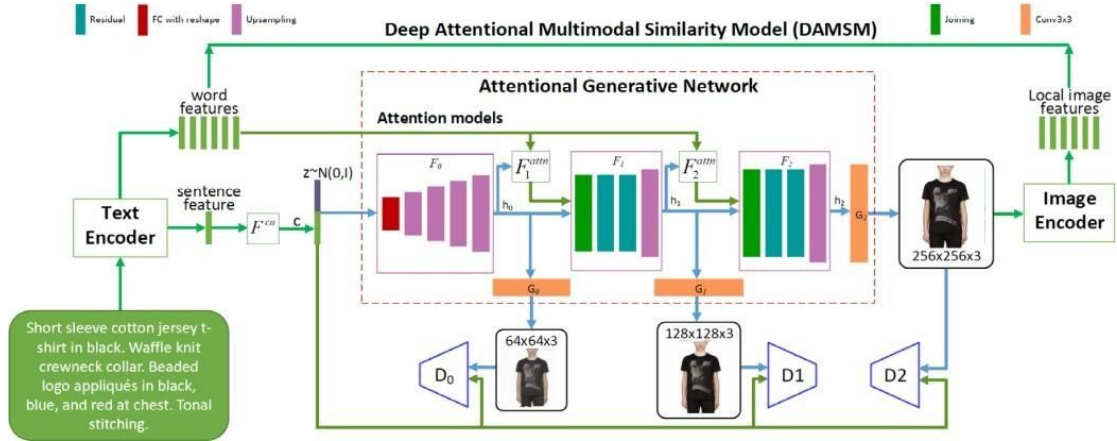
Hình 4.2: Mô hình xấp xỉ phân phối hình ảnh

4.2 AttnGAN

4.2.1 Giới thiệu

AttnGAN [7] là một mô hình được cải tiến từ StackGAN-v2. Mô hình giải quyết được các vấn đề chỉ phát sinh ra ảnh từ cấp độ câu mà chưa thể phát sinh cả

ảnh ở cấp độ từ. Điều này làm cho hình ảnh được tạo ra chưa được chi tiết hết sức có thể. Mô hình trước tiên sẽ tạo ra hình ảnh ở cấp độ câu nhằm tạo ra những nét cơ bản của đối tượng. Sau đó sẽ dựa vào những từ ngữ của câu trên để chia những vùng nhỏ trên hình ảnh thô sơ của đối tượng và tập trung tạo những chi tiết màu sắc của phân vùng đó.



Hình 4.3: Mô hình của AttnGAN được tiểu luận sử dụng cho bài toán phát sinh ảnh thời trang

AttnGAN sẽ gồm hai mảng chính:

- Attentional Generative Network
- Deep attentional multimodal similarity model

4.2.2 Cấu trúc

Mô hình chính kế thừa từ mô hình StackGAN-v2 nên cũng sẽ mang những nét tương đồng với StackGAN-v2. Bên cạnh đó sẽ có những chi tiết khác biệt như sau:

• Attentional Generative Network

Từ hình 4.3, giả sử có mô hình mảng attentional generative có m bộ khởi tạo (G_0, G_1, \dots, G_{m-1}) có các tầng ẩn sau (h_0, h_1, \dots, h_{m-1}) coi như là đầu vào và hình ảnh khởi tạo có kích thước từ nhỏ tới lớn ($\hat{x}_0, \hat{x}_1, \dots, \hat{x}_{m-1}$)

Cụ thể qua công thức sau:

$$\begin{aligned}
h_0 &= F_0(z, F^{ca}(e)) \\
h_i &= F_i(h_{i-1}, F^{attn}(e, h_{i-1})), \text{ với } i = \\
&1, 2, \dots, m-1 \\
(4.2.1) \quad \hat{x} &= G_i(h_i)
\end{aligned}$$

Trong đó, z là vector nhiễu được lấy từ phân phối chuẩn. e là vector của toàn câu, e là ma trận của vector từ, F^{ca} là Conditional Augmentation dùng để chuyển vector e thành vector điều kiện c . F^{attn}_i là model tập trung ở tầng thứ i của AttnGAN. F^{ca} , F^{attn}_i , F_i , G_i đều là mạng neural.

Mô hình AttnGAN $F^{attn}(e, h)$ sẽ có 2 đầu vào: vector đặc trưng $e \in \mathbb{R}^{D \times T}$ và đặc trưng hình ảnh từ tầng ẩn trước đó $h \in \mathbb{R}^{D \times N}$. Đặc trưng từ sẽ được chuyển đổi về không gian chung của đặc trưng hình ảnh bằng cách thêm vào 1 lớp perceptron với $e = Ue$ mà $U = \mathbb{R}^{D \times D}$. Sau đó vector nghĩa từ sẽ được tính toán cho mỗi vùng con của hình ảnh dựa trên tầng ẩn h . Mỗi 1 cột của h là vector đặc trưng cho 1 vùng con của ảnh. Với vùng con thứ j , vector nghĩa từ của vùng đó sẽ được đại diện nghĩa từ liên quan tới h_j và được tính toán như sau:

$$c_j = \sum_{i=0}^{T-1} \beta_{j,i} e'_i, \text{ trong đó } \beta_{j,i} = \frac{\exp(s'_{j,i})}{\sum_{l=0}^{T-1} \exp(s'_{j,l})}, s'_{j,i} = h_j^T e'_i \text{ và } \beta_{j,i}$$

là trọng số của mô hình tại chữ thứ i khi khởi tạo ra vùng con thứ j của ảnh. Sau đó sẽ thêm 1 ma trận nghĩa từ cho đặc trưng hình ảnh h bằng $F^{attn}(e, h) = (c_0, c_1, \dots, c_{N-1}) \in \mathbb{R}^{D \times N}$. Cuối cùng, hình ảnh đặc trưng và đặc trưng nghĩa từ tương ứng sẽ được kết hợp để tạo ra những hình ảnh ở trạng thái kể. Để khởi tạo ra ảnh thực với những điều kiện của những cấp độ (cấp độ câu và cấp độ từ), hàm đối tượng cuối cùng của mạng khởi tạo tập trung được định nghĩa như sau:

$$\mathcal{L} = \mathcal{L}_G + \lambda \mathcal{L}_{DAMSM} \quad (4.2.2)$$

Trong đó:

$$\mathcal{L}_G = \sum_{i=0}^{m-1} \mathcal{L}_{G_i}$$

λ là siêu tham số để cân bằng 2 vế của công thức 4.2.2. Vế đầu là hàm mất mát GAN để phân chia phân phối có điều kiện và không điều kiện. Tại tầng thứ i của AttnGAN, bộ khởi tạo G_i có một bộ phân biệt D_i tương ứng. Hàm mất mát khứ cho G_i được định nghĩa như sau:

$$\mathcal{L}_{G_i} = \underbrace{-\frac{1}{2} \mathbb{E}_{\hat{x}_i \sim p_{G_i}} [\log D_i(\hat{x}_i)]}_{\text{unconditional loss}} - \underbrace{\frac{1}{2} \mathbb{E}_{\hat{x}_i \sim p_{G_i}} [\log D_i(\hat{x}_i, \bar{e})]}_{\text{conditional loss}} \quad (4.2.3)$$

Trong đó hàm mất mát không điều kiện sẽ quyết định ảnh là thật hay giả trong khi hàm mất mát có điều kiện sẽ quyết định ảnh và câu đó có khớp với nhau hay không.

Mặt khác để huấn luyện cho G_i , mọi bộ phân biệt D_i được huấn luyện để phân lớp cho đầu vào là lớp thật hay lớp giả bằng cách tối thiểu hóa hàm mất mát cross-entropy:

$$\begin{aligned} \mathcal{L}_{D_i} = & \underbrace{-\frac{1}{2} \mathbb{E}_{x_i \sim p_{data_i}} [\log D_i(x_i)] - \frac{1}{2} \mathbb{E}_{\hat{x}_i \sim p_{G_i}} [\log(1 - D_i(\hat{x}_i))]}_{\text{unconditional loss}} \\ & + \\ & \underbrace{-\frac{1}{2} \mathbb{E}_{x_i \sim p_{data_i}} [\log D_i(x_i, \bar{e})] - \frac{1}{2} \mathbb{E}_{\hat{x}_i \sim p_{G_i}} [\log(1 - D_i(\hat{x}_i, \bar{e}))]}_{\text{conditional loss}} \end{aligned} \quad (4.2.4)$$

Trong đó x_i là từ phân phối hình ảnh p_{data} ở phân giải thứ i , và \hat{x}_i là từ phân phối mô hình p_G ở cùng phân giải. Bộ phân biệt của AttnGAN thì được phân chia 1 cách có cấu trúc, để nó có thể huấn luyện song song và mỗi một bộ phân biệt sẽ tập trung ở một phân giải hình ảnh đơn.

• Deep attentional multimodal similarity model

Mô hình DAMSM (Deep Attentional Multimodal Similarity Model) dùng 2 mạng neural để ánh xạ vùng con của ảnh và những từ ngữ của câu

sang một không gian ngữ nghĩa chung, sau đó sẽ đo độ tương tự của hình ảnh và văn bản ở cấp độ từ để cho việc tính hàm mất mát cho việc sinh ảnh.

– **Bộ mã hóa văn bản (text encoder)**

Là một mạng bi-directional Long Short-Term Memory (LSTM) được dùng để rút trích những vector ngữ nghĩa từ đoạn văn bản mô tả. Trong mạng bi-directional LSTM, mọi từ tương ứng với hai trạng thái ẩn theo mọi hướng. Vì vậy, hai trạng thái ẩn này sẽ được ghép lại với nhau để biểu diễn ngữ nghĩa của mỗi từ. Ma trận đặc trưng của tất cả các từ được kí hiệu bởi $e \in \mathbb{R}^{D \times T}$. Trong đó, cột thứ i là e_i là vector đặc trưng của từ thứ i , D là số chiều của vector và T là số lượng từ. Trong khi đó, trạng thái ẩn cuối cùng của bi-directional LSTM sẽ được ghép vào vector toàn cục của câu, kí hiệu $e \in \mathbb{R}^D$

– **Bộ mã hóa hình ảnh (image encoder)** Bộ mã hóa hình ảnh là một mạng tích chập (CNN) được dùng để ánh xạ hình ảnh thành vector ngữ nghĩa. Những lớp trung gian của mạng CNN sẽ rút trích các đặc trưng cục bộ của các vùng con khác nhau trên ảnh. Cụ thể, bộ mã hóa hình ảnh sẽ được xây dựng dựa trên mô hình Inception-v3 đã được huấn luyện trên ImageNet. Đầu tiên, hình ảnh sẽ được thay đổi về kích thước 299×299 pixels. Sau đó, ma trận đặc trưng cục bộ có kích thước $f \in \mathbb{R}^{768 \times 289}$ (được chuyển đổi từ $768 \times 17 \times 17$) sẽ được rút trích từ lớp “mixed_6e” của mạng Inception-v3. Trong đó, 768 là số chiều của vector đặc trưng cục bộ, và 289 là số vùng con của ảnh. Trong khi đó, vector đặc trưng toàn cục $f \in \mathbb{R}^{2048}$ sẽ được rút trích từ lớp average pooling cuối cùng của mạng Inception-v3. Cuối cùng, đặc trưng của ảnh sẽ được chuyển đổi sang không gian ngữ nghĩa chung của đặc trưng văn bản bằng việc được thêm vào một lớp perceptron như sau:

$$v = Wf, v = Wf, \quad (4.2.5)$$

Trong đó, $v \in \mathbb{R}^{D \times 289}$ và vector cột thứ i là vector đặc trưng cho vùng con thứ i của ảnh, $v \in \mathbb{R}^D$ là vector đặc trưng toàn cục của cả bức ảnh. D là số chiều của vector ảnh và văn bản trong không gian đặc trưng. Để hiệu quả hơn, các trọng số trong các lớp của mô hình Inception-v3 được giữ nguyên, và các trọng số

trong lớp được thêm vào sẽ được huấn luyện đồng thời cùng với các mảng còn lại.

- **The attention-driven image-text matching score** Được dùng để đo lường độ khớp của cặp ảnh và văn bản dựa vào attention model giữa ảnh và văn bản. Đầu tiên, ma trận tương đương của tất cả các cặp từ trong câu với các vùng con trong ảnh được tính như sau:

$$s = e^T v \quad (4.2.6)$$

Trong đó, $s \in \mathbb{R}^{T \times 289}$ và $s_{i,j}$ là tích vô hướng giữa vector đặc trưng của từ thứ i trong câu với vùng con j của ảnh. Ma trận tương đương sẽ được chuẩn hóa như sau:

$$\bar{s}_{j,i} = \frac{\exp(s_{i,j})}{\sum_{k=0}^{T-1} \exp(s_{k,j})} \quad (4.2.7)$$

Sau đó, một mô hình attention sẽ được tạo ra để tính vector vùng – nghĩa cho từng từ. Vector vùng – nghĩa c_i sẽ thể hiện vùng con của ảnh liên quan đến từ thứ i của câu và được tính bằng cách nhân tất cả vector đặc trưng của các vùng con của ảnh với trọng số, sau đó lấy tổng của các vector đó.

$$C_i = \sum_{j=0}^{288} \alpha_j v_j, \quad (4.2.8)$$

Trong đó:

$$\alpha = \frac{\exp(\gamma_1 \bar{s}_{i,j})}{\sum_{k=0}^{288} \exp(\gamma_1 \bar{s}_{i,k})}$$

γ_1 là giá trị quyết định mức độ chú ý của đặc trưng của vùng con tương ứng khi tính vector vùng-nghĩa cho một từ. Cuối cùng, *attention-driven image-text matching score* của toàn bộ hình ảnh (Q) và toàn bộ đoạn miêu tả (D) được tính như sau:

$$R(Q, D) = \log(\sum_{i=1}^{T-1} \exp(\gamma_2 R(c_i, e_j)))^{\frac{1}{\gamma_2}}, \quad (4.2.9)$$

Dựa vào thực nghiệm trên tập validation, ta xét các tham số như sau: $\gamma_1 = 5$, $\gamma_2 = 5$, $\gamma_3 = 10$ và $M = 50$. Mô hình DAMSM sẽ được huấn luyện bằng cách tối thiểu hàm mất mát nhằm sử dụng các cặp ảnh-văn bản thật. Vì kích thước của ảnh được xử lý bởi DAMSM không bị giới hạn bởi kích thước ảnh được khởi tạo nên kích thước ảnh thật được sử dụng là 299×299 . Bên cạnh đó, bộ mã hóa văn bản đã được huấn luyện trước trong mô hình DAMSM cũng cung cấp các vector từ có thể phân biệt trực quan được rút trích từ các cặp ảnh - văn bản để sử dụng cho mô hình *attentional generative network*. Để so sánh, các vector từ thông thường được xử lý trước trên dữ liệu văn bản thuần túy thường không phân biệt trực quan được, ví dụ: các vector từ khác nhau về các màu sắc, chẳng hạn như đỏ, xanh, vàng, v.v., thường được phân cụm cùng nhau trong không gian vector, do thiếu nền tảng là các tín hiệu hình ảnh thực tế.

Tổng kết lại, có hai mô hình attention là attentional generative network và DAMSM, hai mô hình này đóng vai trò khác nhau trong AttnGAN. Cơ chế attention trong mạng sinh (2) cho phép AttnGAN có khả năng tự động lựa chọn biến điều kiện ở cấp độ từ cho việc khởi tạo những vùng con khác nhau của ảnh. Với cơ chế attention (công thức 4.2.9), mạng DAMSM sẽ có khả năng tính độ khớp giữa ảnh-văn bản dựa vào hàm mất mát L_{DAMSM} . Chú ý, L_{DAMSM} chỉ được áp dụng cho generator cuối cùng là G_{m-1}

Chương 5

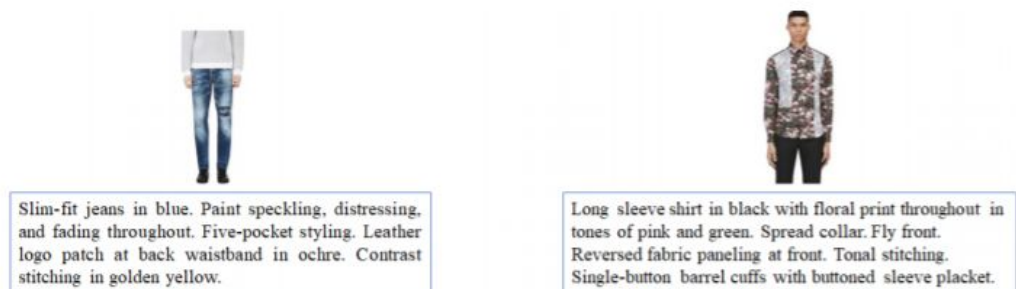
KẾT QUẢ NGHIÊN CỨU

5.1 Tập dữ liệu sử dụng

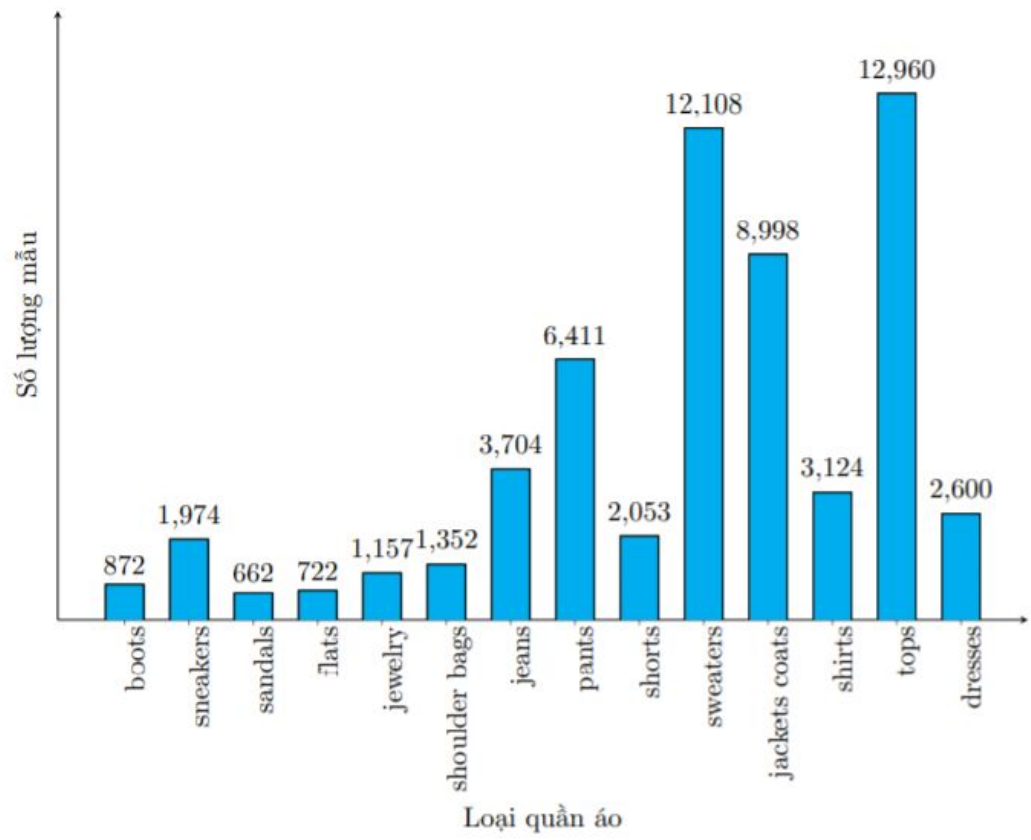
Đồ án sử dụng bộ dữ liệu Fashion-gen [5] đây là bộ dữ liệu chuẩn được sử dụng cho bài toán phát sinh ảnh thời trang. Bộ dữ liệu này có số lượng ảnh lớn và câu mô tả đi kèm như hình 5.5. Bộ dữ liệu Fashion-gen được chia thành nhiều loại quần áo khác nhau, tuy nhiên trong giới hạn của luận văn này, chỉ chọn ra 14 loại quần áo có số lượng ảnh dữ liệu và câu mô tả nhiều hơn 500 mẫu, với góc độ ảnh trực diện. Do đó số lượng dữ liệu được sử dụng cho việc huấn luyện và đánh giá mô hình còn lại như trong bảng 5.1. Các loại quần áo cũng như số lượng mẫu tương ứng cho từng loại quần áo trong tập huấn luyện và kiểm tra mà luận văn sử dụng được thể hiện lần lượt trên biểu đồ 5.2 và 5.3.

Tập dữ liệu	Fashion-gen	
	Huấn luyện	Kiểm tra
Số mẫu	52,174	6,523

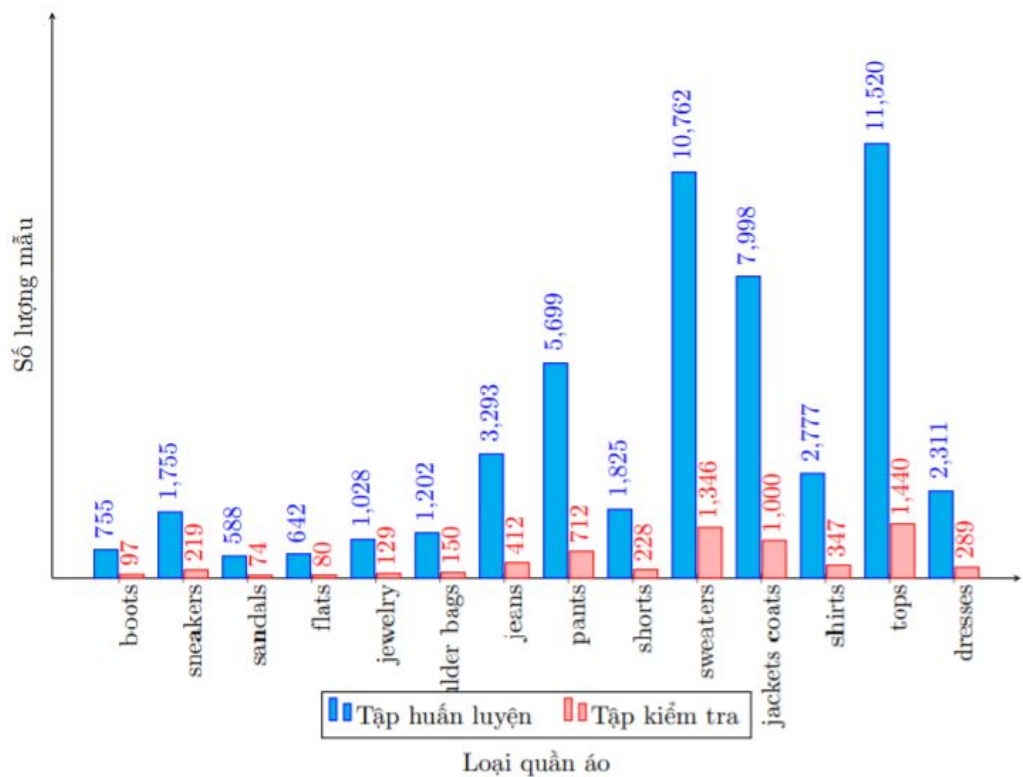
Bảng 5.1: thống kê số lượng mẫu phân bố trên tập huấn luyện và tập kiểm tra



Hình 5.1: Một số mẫu của tập dữ liệu Fashion-gen



Hình 5.2: Thống kê tập dữ liệu theo các loại quần áo



Hình 5.3: Thống kê tập dữ liệu theo tập huấn và kiểm tra.

5.2 Tiêu chí đánh giá

5.2.1 Inception Score (IS)

IS [6] là tiêu chí đánh giá tính thực tế của ảnh được phát sinh. Hai tiêu chuẩn đánh giá ảnh phát sinh bao gồm:

- Các ảnh phát sinh chứa các đối tượng có nghĩa, khi đó phân phối nhãn có điều kiện (the conditional class distribution) $p(y|x)$ có giá trị entropy thấp.
- Các ảnh phát sinh đa dạng, khi đó phân phối nhãn cận biên (the 52 marginal class distribution) $p(y) = \sum_x p(y|x)p_g(x)$ có giá trị entropy cao, trường hợp lý tưởng phân phối này có dạng phân phối đều. Kết hợp hai tiêu chuẩn trên ta có công thức tính IS như sau:

$$IS(G) = \exp(\mathbb{E}_{x \sim p_g} D_{KL}(p(y|x) || p(y))) \quad (5.2.1)$$

Trong đó:

- y là nhãn và x là ảnh được phát sinh.
- $DKL()$ là KL-divergence là một hàm được sử dụng để tính khoảng cách giữa $p(y|x)$ và $p(y)$.

Việc sử dụng hàm mũ exp giúp giá trị có thể dễ dàng để so sánh, khi lấy $\ln(\text{IS}(G))$ mà không làm mất đi tính tổng quát. Nếu cả hai tiêu chuẩn trên được thỏa mãn thì giá trị KL-divergence giữa hai phân phối $p(y)$ và $p(y|x)$ sẽ lớn dẫn đến kết quả IS lớn. Có thể xấp xỉ công thức tính IS thực tế từ các mẫu $x^{(i)}$ như sau:

$$\text{IS}(G) \approx \exp\left(\frac{1}{N} \sum_{i=1}^N D_{KL}(p(y|x^{(i)}) \parallel \hat{p}(y))\right) \quad (5.2.2)$$

Với N là số lượng mẫu và thường được khuyến khích chọn là 5000 [6]. Một số giới hạn của độ đo IS:

- Độ đo này bị giới hạn bởi bộ phân loại Inception, nó phụ thuộc vào dữ liệu huấn luyện ImageNet 2014, do đó nếu mô hình phát sinh các đối tượng không được thể hiện trong tập huấn luyện của ImageNet 2014 thì sẽ có thể luôn luôn nhận được giá trị IS thấp mặc dù ảnh phát sinh có chất lượng cao bởi vì ảnh không thể phân loại một cách chính xác.
- Mạng phân lớp không thể phân loại các đặc trưng liên quan đến khái niệm chất lượng ảnh do CNN chủ yếu tập trung vào thông tin cục bộ hơn là thông tin kết cấu hình dáng, vì vậy những ảnh có chất lượng kém vẫn có thể đạt được điểm số IS cao.
- Độ đo này phụ thuộc vào giá trị trọng số của mô hình huấn luyện, trong số mô hình mạng Inception được huấn luyện trên các framework khác nhau có thể dù kết quả phân loại không có tác động đáng kể. Tuy nhiên, trọng số khác biệt lại dẫn đến điểm số khác biệt trên cùng tập mẫu [1].
- Một khuyết điểm của IS là nó có thể làm sai lệch hiệu suất của mô hình nếu trường hợp mô hình chỉ phát sinh duy nhất một ảnh trên một lớp, giá trị IS vẫn có thể cao do độ đo này không đo đặc sự đa dạng trong từng lớp.
- Nếu bộ phát sinh có thể ghi nhớ dữ liệu huấn luyện và sao chép nó thì điểm số có thể cao.

5.2.2 Fréchet Inception Distance (FID)

Để khắc phục một số hạn chế của IS. Độ đo FID [4] được sử dụng để đo đặc chất lượng mẫu được phát sinh dựa vào khoảng cách giữa phân phối của ảnh thật và phân phối của ảnh phát sinh. Để tính được FID giữa ảnh thật và ảnh được phát sinh, ta sử dụng công thức sau:

$$\text{FID}(r, g) = \|\mu_r - \mu_g\|^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{\frac{1}{2}}) \quad (5.2.3)$$

Trong đó:

- μ, Σ là giá trị trung bình và ma trận hiệp phương sai (covariance) của phân phối Gaussian.
- r, g lần lượt là ảnh thật và ảnh được phát sinh.
- T_r là tổng tất cả các phần tử trên đường chéo.

FID càng nhỏ nghĩa là ảnh phát sinh càng có chất lượng tốt(quality) và càng đa dạng(diversity) cũng như có nhiều sự tương tự giữa ảnh phát sinh và ảnh thật. Có thể nói FID thể hiện tốt trường hợp nhiều hơn so với IS. Vì vậy FID đo đạt rất tốt tính đa dạng của ảnh. Ngoài ra FID còn cho thấy giá trị bias cao nhưng variance thấp và rất nhạy cảm với hiện tượng mode collapse.

5.3 Kết quả mô hình phát sinh

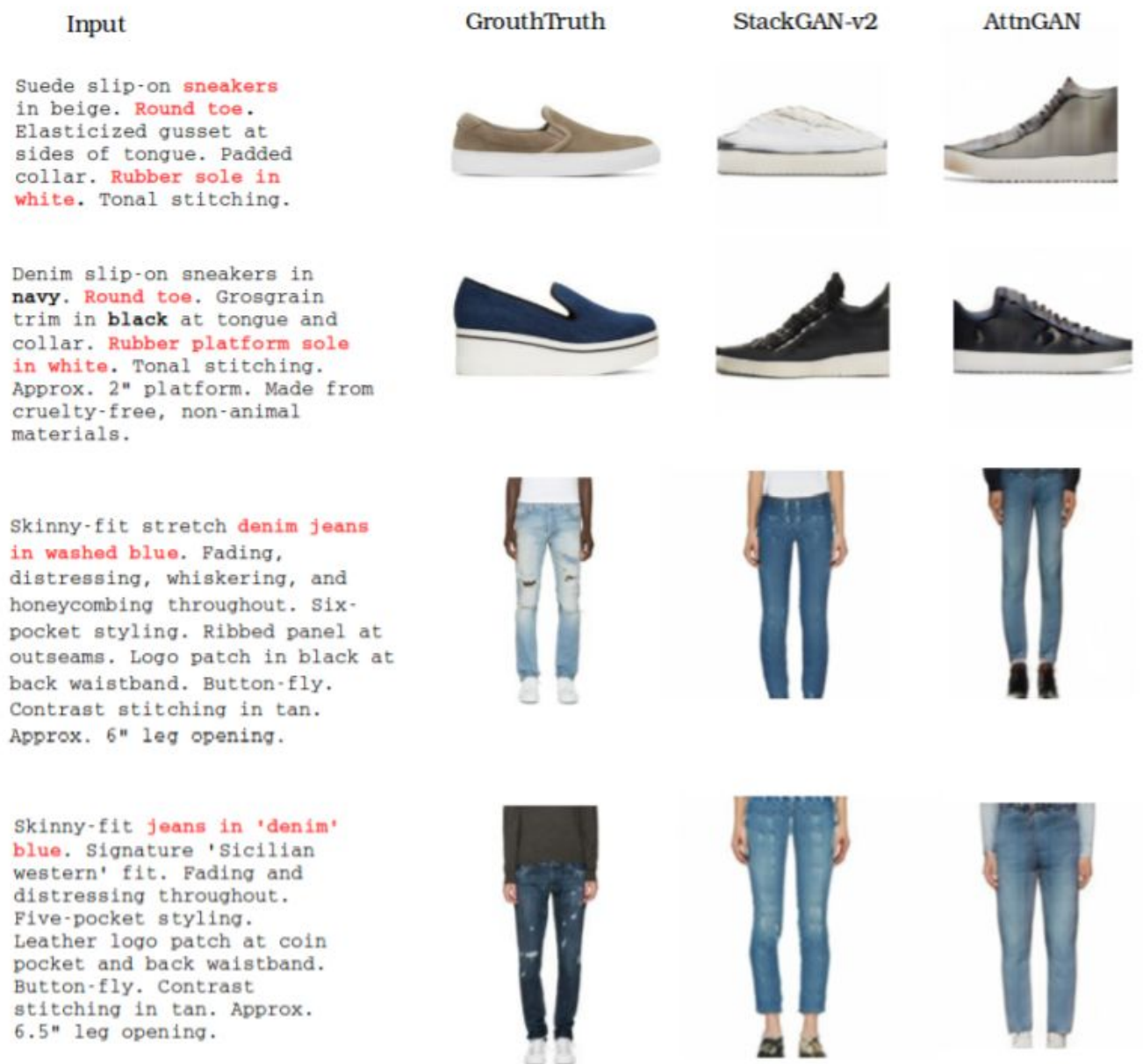
- Khi tiến hành huấn luyện và đánh giá hai mô hình StackGAN-v2, AttnGAN trên tập dữ liệu Fashion-gen luận văn tiến hành so sánh hai mô hình trên dựa vào hai độ đo phổ biến cho bài toán phát sinh ảnh là IS và FID kết quả được thể hiện như bảng 5.2.

Phương pháp	IS	FID
StackGAN-v2	3.383 \pm 0.197	33.53
AttnGAN	2.719 \pm 0.719	30.97













Bảng 5.2: Bảng kết quả so sánh giữa StackGAN-v2 và AttnGAN dựa vào phép đo IS, FID trên tập dữ liệu Fashion-Gen

- Dựa vào bảng kết quả so sánh, ta thấy rằng nếu xét về điểm IS thì StackGAN-v2 có xu hướng tốt hơn so với AttnGAN khi đạt điểm số là 3.383, điều này cho thấy ảnh được tạo ra bởi StackGAN-v2 thỏa mãn tiêu chuẩn về chất lượng khi ảnh được tạo ra có thể nhận dạng được và các đối tượng được tạo ra có sự khác biệt. Tuy nhiên, nếu xét về độ đo FID thì AttnGAN lại có xu hướng tốt hơn so với StackGAN-v2 với FID là 30.97 điều này cho thấy ảnh được tạo ra bởi AttnGAN có chất lượng tốt và gần giống với tập dữ liệu huấn luyện. Nếu xét về tổng thể, dựa trên kết quả thực tế khi phát sinh ảnh (hình 5.2, 5.3) AttnGAN dường như thể hiện được một cách chi tiết các thuộc tính quần áo hơn so với StackGAN-v2, ngoài ra các loại quần áo được tạo bởi AttnGAN chính xác hơn so với StackGAN-v2. Bởi vì độ đo IS không thể hiện được tính liên hệ giữa các mẫu thuộc cùng một loại quần áo nên điểm số

IS của StackGAN-v2 không bị ảnh hưởng. Mặc dù vậy, AttnGAN vẫn chưa giải quyết được vấn đề mode collapse khi vẫn còn nhiều mẫu phát sinh trùng lặp.



Hình 5.4: Kết quả so sánh giữa ảnh được StackGANv2 và AttnGAN tạo ra với quần và giày.

Input	GrouthTruth	StackGAN-v2	AttnGAN
<p>Long sleeve crewneck geometric print hoodie in deep khaki green. Drawstring closure at hood. Ribbed trim throughout. Zip closure at front. Geometric screen printed motifs in khaki green throughout. Side pockets. Terry loop interior. Tonal stitching.</p>			
<p>Long sleeve striped sweatshirt in navy and white. Skewed dot print throughout. Ribbed crewneck collar, sleeve cuffs, and hem. Raised velveteen appliqueacute; and embroidered logo at front in black and white. Fleece lining. Tonal stitching.</p>			
<p>Short sleeve cropped blouse in white. Embossed grid print throughout. Crewneck collar. Zip closure at back. Bonded backing in tonal neoprene. Tonal stitching.</p>			
<p>Short sleeve t-shirt in white. Crewneck collar. Floral graphic printed throughout yoke in black. Tonal stitching.</p>			

Hình 5.5: Kết quả so sánh giữa ảnh được StackGANv2 và AttnGAN tạo ra với các loại áo thun và áo khoác

Chương 6

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1 Kết Luận

Đề tài đồ án đã thực hiện được một số công việc sau:

- Tìm hiểu các mô hình phát sinh ảnh từ câu mô tả.
- Xây dựng mô hình phát sinh ảnh thời trang từ câu mô tả dựa trên các công trình nghiên cứu đã tìm hiểu với hai phương pháp phổ biến là StackGAN-v2 và AttnGAN.
- Tiến hành thực nghiệm trên bộ dữ liệu Fashion-gen và từ đó có những so sánh để thấy rằng độ đo IS thì StackGAN-v2 cho kết quả tốt hơn so với AttnGAN nếu xét về sự đa dạng của các ảnh phát sinh. Tuy nhiên, nếu xét về độ giống giữa ảnh phát sinh và ảnh thật thì AttnGAN cho kết quả khả quan hơn so với StackGAN-v2. Tuy nhiên, vấn đề mode collapse vẫn còn chưa được khắc phục trong mô hình AttnGAN trên tập dữ liệu Fashion-gen.

6.2 Hướng phát triển

Một số hạn chế của đồ án:

- Mặc dù hình ảnh được tạo ra với phân giải khá cao song vẫn còn một vài chi tiết chưa được tạo ra rõ ràng, cụ thể thể nhưng trong tương lai mô hình có thể được cải thiện được vấn đề trên.
- Mô hình vẫn chưa có thể tạo ra hình ảnh cụ thể, hoặc nền phức tạp.
- Vấn đề mode collapse vẫn chưa được giải quyết.
- Từ những hạn chế hiện tại, luận văn hy vọng sẽ có thể cải thiện được mô hình phát sinh ảnh thời trang từ câu mô tả đầu vào trong tương lai.

Tài liệu tham khảo

- [1] Shane Barratt and Rishi Sharma. A note on the inception score. *ArXiv*, abs/1801.01973, 2018.
- [2] Emily L. Denton, Soumith Chintala, Arthur Szlam, and Robert Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. *CoRR*, abs/1506.05751, 2015.
- [3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Ben-gio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672– 2680. Curran Associates, Inc., 2014.
- [4] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a nash equilibrium. *CoRR*, abs/1706.08500, 2017.
- [5] N. Rostam Zadeh, S. Hosseini, T. Bouquet, W. Stokowiec, Y. Zhang, C. Jauvin, and C. Pal. Fashion-Gen: The Generative Fashion Dataset and Challenge. *ArXiv e-prints*, June 2018.
- [6] Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *CoRR*, abs/1606.03498, 2016.

- [7] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. *CoRR*, abs/1711.10485, 2017.
- [8] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaolei Huang, Xiaogang Wang, and Dimitris N. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. *CoRR*, abs/1612.03242, 2016.
- [9] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N. Metaxas. Stackgan++: Re-alistic image synthesis with stacked generative adversarial networks. *CoRR*, abs/1710.10916, 2017.
- [10] [Fashion-Gen: The Generative Fashion Dataset and Challenge](#) by Negar Rostamzadeh, Seyedarian Hosseini, Thomas Boquet, Wojciech Stokowiec, Ying Zhang, Christian Jauvin, Chris Pal.

