**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**



**ĐỒ ÁN TỐT NGHIỆP**

**TÌM HIỂU BÀI TOÁN GỢI Ý PHỐI TRANG PHỤC DÙNG HỌC SÂU**

**SVTH 1: HUỲNH DUY ANH**

**MSSV: 15110004**

**SVTH 2: NGUYỄN DANH NGHI**

**MSSV: 15110087**

**Khóa: 2015**

**Ngành: CÔNG NGHỆ THÔNG TIN**

**GVHD: TS NGUYỄN THIÊN BẢO**

Tp. Hồ Chí Minh, tháng 7 năm 2020

|  |  |
| --- | --- |
|  | 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: Huỳnh Duy Anh  Họ và tên sinh viên: Nguyễn Danh Nghi | MSSV: 15110004  MSSV: 15110087 |
| Ngành: Công Nghệ Thông Tin | Lớp: 15110CL2 |
| Giảng viên hướng dẫn: Nguyễn Thiên Bảo | ĐT: 0927613761 |
| Ngày nhận đề tài: 24/02/2020 | Ngày nộp đề tài: 01/07/2020 |
| 1. Tên đề tài: Tìm hiểu bài toán phối trang phục dùng học sâu 2. Các số liệu, tài liệu ban đầu:    * Bài báo: Yujie Lin, Pengjie. Ren, Zhumin Chen, Zhaochun Ren, Jun Ma, and Maarten de Rijke, 2018. Explainable Fashion Recommendation with Joint Outfit Matching and Comment Generation (2018).    * Data: Tập hình ảnh và dataset trong Jay Ren fashion recommendation tkde2018 code dataset. 3. Nội dung thực hiện đề tài:    * Tìm hiểu về học sâu.    * Tìm hiểu về mạng nơ ron tích chập (CNN).    * Tìm hiểu cơ chế mutual attention.    * Tìm hiểu về bài toán phối trang phục dùng Convolutional Neural Network (CNN). 4. Sản phẩm: Source code giải thuật cho bài toán về phoi trang phục | |
| TRƯỞNG NGÀNH  Nguyễn Đăng Quang | GIẢNG VIÊN HƯỚNG DẪN  TS. Nguyễn Thiên Bảo |

|  |  |
| --- | --- |
|  | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập – Tự do – Hạnh phúc**  **\*\*\*\*\*\*\*** |

# PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên sinh viên: Huỳnh Duy Anh MSSV: 15110004

Họ và tên sinh viên: Nguyễn Danh Nghi MSSV: 15110087  
Ngành: Công nghệ thông tin  
Tên đề tài: Tìm hiểu bài toán phối trang phục dùng học sâu  
Họ và tên giáo viên hướng dẫn: Nguyễn Thiên Bảo

NHẬN XÉT1. Về nội dung đề tài & khối lượng thưc 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:

* Nắm đươc kiến thức về học máy, học sâu như CNN, RNN, cơ chế Attention
* Tìm hiểu bài toán gợi ý phối trang phục dùng học sâu
* Sinh viên nắm được kiến trúc của mô hình gợi ý trang phục dùng học sâu từ đó áp dụng vào thực tế.

+ Về thực hành:

Sinh viên chạy được demo về gợi ý trang phục dùng CNN với cơ chế Attention trên tập dataset FashionCV.

2. Ưu điểm:

* Sinh viên chịu khó tìm hiểu, cố gắng tìm kiếm các tài liệu liên quan về đề tài
* Sinh viên thực hiện tốt các công việc và đúng deadline
* Thái độ làm việc: làm việc nghiêm túc, chủ động liên hệ với GVHD

3. Khuyết điểm:

* Hạn chế về nguồn lực phần cứng
* Chưa có kinh nghiệm về chạy deep learning trên môi trường google colab
* Chưa đưa ra được độ đo đánh giá cuối cùng khi chạy thực nghiệm
* Kết quả về mặt trực quan cũng cần cải thiện thêm

4. Đề nghị cho bảo vệ hay không?

5. Đánh giá loại: Khá

6. Điểm: (Bằng chữ: )

*Tp. Hồ Chí Minh, ngày tháng năm 2020*

Giảng viên hướng dẫn

*(Ký & ghi rõ họ tên)*

TS. NGUYỄN THIÊN BẢO

|  |  |
| --- | --- |
|  | **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: Huỳnh Duy Anh MSSV: 15110004

Họ và tên sinh viên: Nguyễn Danh Nghi MSSV: 15110087

Ngành: Công nghệ thông tin  
Tên đề tài: Tìm hiểu bài toán phối trang phục dùng học sâu

Họ và tên Giáo viên phản biện:

NHẬN XÉT1. Về nội dung đề tài & khối lượng thưc hiện:

2. Ưu điểm:

3. Khuyết điểm:

4. Đánh giá loại:

5. Điểm: (Bằng chữ: )

*Tp. Hồ Chí Minh, ngày tháng năm 2020*

Giảng viên phản biện

*(Ký & ghi rõ họ tên)*

# **LỜI CẢM ƠN**

Nhóm chúng em xin chân thành cảm ơn thầy Nguyễn Thiên Bảo hướng dẫn, giảng giải, cô Võ Hoàng Anh, và các anh/chị/bạn khóa trước đã nhiệt tình giúp đỡ, và đưa ra những ý kiến hữu ích để chúng em có thể hoàn thành tốt báo cáo này.

Em xin chân thành cảm ơn quý thầy cô khoa Công nghệ thông tin trường Đại học Sư phạm Kỹ thuật thành phố Hồ Chí Minh đã tận tình truyền đạt kiến thức trong những năm em học tập ở trường. Với vốn kiến thức được tiếp thu trong quá trình học tập không chỉ là nền tảng cho quá trình nghiên cứu khóa luận mà còn là hành trang để em bước vào đời một cách vững chắc.

Với vốn kiến thức hạn hẹp và thời gian thực hiện có hạn nên em không tránh khỏi những thiếu sót. Rất mong nhận được những ý kiến đóng góp của quý thầy cô để em có thể tiếp tục phát triển luận văn này lên thành đồ án tốt nghiệp.

Xin chân thành cảm ơn!

# **LỜI MỞ ĐẦU**

Có cái nhìn tổng quan về AI, Machine Learning. Tìm hiểu sâu hơn về những kiến thức và những công trình nghiên cứu để ứng dụng của lĩnh vực này vào thực tế, đặc biệt là Deep Learning.

Từ cơ sở lý thuyết và những kiến thức từ các công trình nghiên cứu đã được công bố mà nhóm tìm hiểu, từ đó nhóm muốn thấy được lợi ích của việc ứng dụng AI, Machine Learning vào thực tế nhằm đem đến những lợi ích trên nhiều lĩnh vực quan trọng như kinh tế, giáo dục, y tế, quốc phòng, xã hội…

# **MỤC LỤC**

[**NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP** i](#_Toc44472664)

[**PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN** ii](#_Toc44472665)

[**PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN** iv](#_Toc44472666)

[**LỜI CẢM ƠN** vi](#_Toc44472667)

[**LỜI MỞ ĐẦU** vii](#_Toc44472668)

[**MỤC LỤC** viii](#_Toc44472669)

[**DANH MỤC CÁC CHỮ VIẾT TẮT** x](#_Toc44472670)

[**DANH MỤC CÁC BẢNG BIỂU** xi](#_Toc44472671)

[**DANH MỤC HÌNH ẢNH** xii](#_Toc44472672)

[**CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI** 13](#_Toc44472673)

[**CHƯƠNG 2: BÀI TOÁN GỢI Ý TRANG PHỤC** 16](#_Toc44472674)

[**CHƯƠNG 3: TỔNG QUAN VỀ HỌC SÂU** 17](#_Toc44472675)

[**3.1. Deep Learning** 17](#_Toc44472676)

[**3.2. Convolutional Neural Network** 17](#_Toc44472677)

[**3.3. Recurrent Neural Network** 20](#_Toc44472678)

[**CHƯƠNG 4: GIẢI PHÁP ĐỀ XUẤT CHO BÀI TOÁN GỢI Ý TRANG PHỤC** 22](#_Toc44472679)

[**4.1. Tổng quan** 22](#_Toc44472680)

[**4.2. Bộ mã hóa hình ảnh phần trên và phần dưới** 24](#_Toc44472681)

[**4.3. Matching Decoder** 27](#_Toc44472682)

[**4.4. Hàm mất mát** 28](#_Toc44472683)

[**CHƯƠNG 5: CÀI ĐẶT THỬ NGHIỆM** 29](#_Toc44472684)

[**5.1 Ứng dụng vào việc phân vùng đối tượng trong ảnh thời trang** 29](#_Toc44472685)

[**5.2. Tập dữ liệu** 29](#_Toc44472686)

[**5.3. Môi trường** 29](#_Toc44472687)

[**5.4. Kết quả** 30](#_Toc44472688)

[**CHƯƠNG 6: KẾT LUẬN** 58](#_Toc44472689)

[**6.1. Tổng kết** 58](#_Toc44472690)

[**6.2. Ưu điểm, nhược điểm** 58](#_Toc44472691)

[**6.3. Công việc tương lai** 58](#_Toc44472692)

[**TÀI LIỆU THAM KHẢO** 59](#_Toc44472693)

# **DANH MỤC CÁC CHỮ VIẾT TẮT**

CNN: Convolutional Neural Network

AUC: Area Under the ROC Curve

MAP: Maximum A Posteriori

MRR: Mean Reciprocal Rank

RNN: Recurrent neural network

GPU: Graphics Processing Unit

RAM: Random Access Memory

# **DANH MỤC CÁC BẢNG BIỂU**

[**Bảng 5.1:** Kết quả khi train với tập gồm 400 hình ảnh thời trang 35](#_Toc44469911)

[**Bảng 5.2:** Kết quả khi train với tập gồm 5000 hình ảnh thời trang 40](#_Toc44469912)

[**Bảng 5.3**: Kết quả khi train với tập gồm 10000 hình ảnh thời trang 46](#_Toc44469913)

[**Bảng 5.4:** Kết quả khi train với tập gồm 18000 hình ảnh thời trang 52](#_Toc44469914)

[**Bảng 5.5:** Kết quả khi train với tập gồm 25000 hình ảnh thời trang 57](#_Toc44469915)

# **DANH MỤC HÌNH ẢNH**

[**Hình 1.1:** Trang phục từ Chictopia. Người dùng chia sẻ rộng rãi các thành phần trang phục của họ với công chúng. 13](#_Toc44453047)

[**Hình 1.2:** Mô hình gợi ý trang phục. Bao gồm một danh sách ngắn trang phục được gợi ý và điểm số phù hợp. 14](#_Toc44453048)

[**Hình 3.1:** Bộ lọc được sử dụng trong lớp tích chập đầu tiên là các ma trận kích thước 3x3 của -1, 0 và 1. 18](#_Toc44453054)

[**Hình 3.2:** Ví dụ về bộ lọc cạnh (đứng phải, đứng trái, ngang dưới, ngang trên) với đầu vào là ảnh số viết tay. 18](#_Toc44453055)

[**Hình 3.3:** Ví dụ pooling theo giá trị cực đại 19](#_Toc44453056)

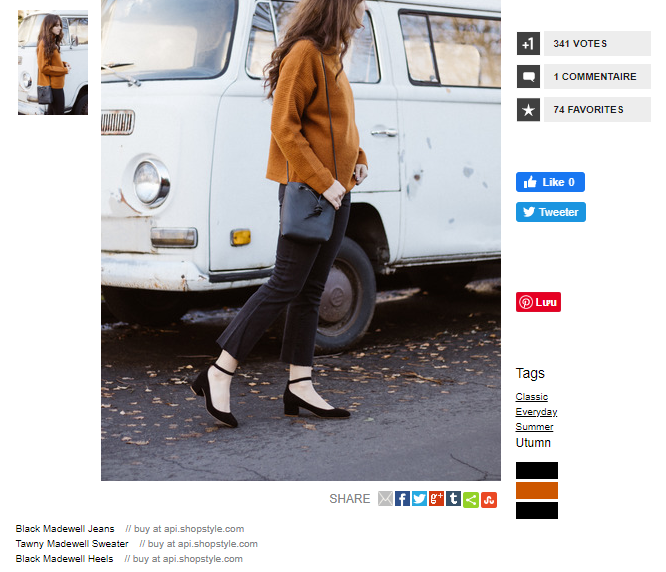
[**Hình 3.4:** Kiến trúc của mô hình RNN. 20](#_Toc44453057)

[**Hình 4.1:** (a) Bộ mã hóa hình ảnh trên và dưới trích xuất các tính năng hình ảnh và từ hình ảnh. (b) Sử dụng cơ chế chú ý lẫn nhau, chúng tôi chuyển đổi các tính năng trực quan sang các biểu diễn tiềm ẩn và . Sau đó, Matching Decoder dự đoán chỉ số phù hợp . 23](#_Toc44453059)

[**Hình 4.2:** 4x4x3 RGB Image. 24](#_Toc44453060)

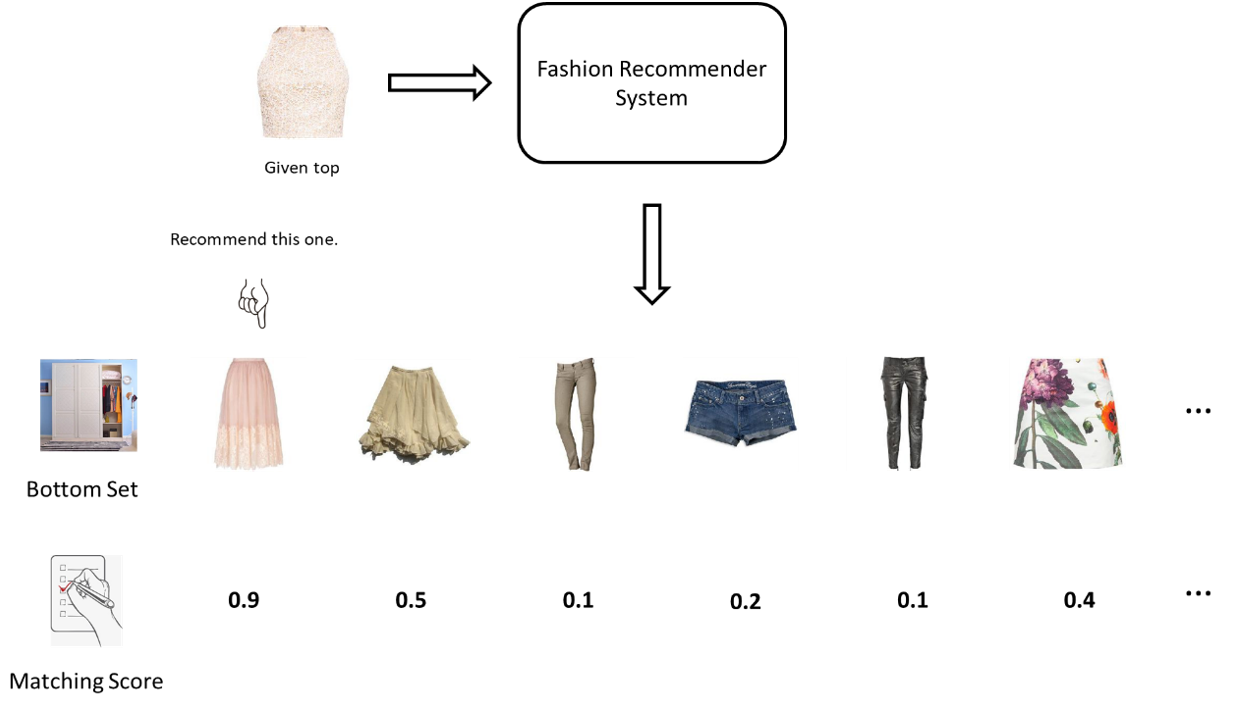
# **CHƯƠNG 1****:** **TỔNG QUAN VỀ ĐỀ TÀI**

Gợi ý phối trang phục đóng một vai trò ngày càng quan trọng trong thị trường bán lẻ trực tuyến. Mục đích của việc này là để thúc đẩy sự quan tâm và tham gia của mọi người vào mua sắm trực tuyến bằng cách gợi ý trang phục thời trang mà họ có thể quan tâm. Những nghiên cứu đầu tiên về đề tài gợi ý phối trang phục thường dựa trên các bộ dữ liệu nhỏ, điều này ngăn cản sự phát triển của các mô hình phức tạp cần các dataset lớn (ví dụ: các mô hình dựa trên học sâu). Trong những năm gần đây, với sự phát triển của các cộng đồng định hướng thời trang trực tuyến trực tuyến, ví dụ như Polyvore, Chictopia, mọi người có thể chia sẻ và nhận xét về các thành phần trang phục, như trong Hình 1. Các thông tin này rất có giá trị để xây dựng các hệ thống thông minh và chính xác hơn.



**Hình 1.1:** Trang phục từ Chictopia. Người dùng chia sẻ rộng rãi các thành phần trang phục của họ với công chúng.

Nhiệm vụ cần làm là đề xuất một mô hình gợi ý phối trang phục dùng học sâu. Bài toán đưa ra một phần trên (ví dụ: áo sơ mi), cần đề xuất một danh sách ngắn các phần dưới (ví dụ: quần, váy, vv…) phù hợp nhất với phần trên từ một bộ sưu tập lớn và ngược lại. Chúng ta cần làm là mô hình hóa khả năng tương thích của các yếu tố thời trang, ví dụ: màu sắc, chất liệu, hoa văn, hình dạng, v.v.



**Hình 1.2:** Mô hình gợi ý trang phục. Bao gồm một danh sách ngắn trang phục được gợi ý và điểm số phù hợp.

Để gợi ý trang phục, sử dụng mạng lưới thần kinh tích chập (CNN) với cơ chế chú ý lẫn nhau để trích xuất các đặc điểm hình ảnh của trang phục. Cụ thể, trước tiên, chúng ta sử dụng CNN để mô hình hóa phần trên và phần dưới thành các vectơ tiềm ẩn, sau đó chúng ta đề xuất một cơ chế chú ý lẫn nhau, trích xuất các đặc điểm hình ảnh nổi bật của cả phần trên và phần dưới bằng cách sử dụng các vectơ trên để khớp với các vectơ dưới và ngược lại. Các tính năng hình ảnh sau đó được giải mã thành một điểm đánh giá bằng Matching Decoder.

Đã có một số nghiên cứu về khuyến nghị trang phục tương tự. Song và các cộng sự, những người đầu tiên sử dụng mạng mã hóa tự động kép để tìm hiểu không gian tương thích tiềm ẩn, họ mô hình hóa mối quan hệ kết hợp giữa các tính năng trực quan (nghĩa là hình ảnh) và các tính năng theo ngữ cảnh (nghĩa là danh mục, thẻ). Sau đó, họ sử dụng xếp hạng cá nhân hóa Bayes (BPR) để khai thác các ưu tiên theo cặp giữa phần trên và phần dưới. Sự khác biệt của bài viết này và họ là chúng ta giới thiệu một cơ chế chú ý lẫn nhau và đa phương thức vào không gian tương thích tiềm ẩn thay vì mạng tự động mã hóa kép. Chúng tôi cùng đào tạo tính năng trích xuất và xếp hạng ưu tiên trong một sơ đồ truyền bá ngược.

Tập dữ liệu là bộ dữ liệu thực tế lớn từ Polyvore chứa thông tin đa phương thức, ví dụ: hình ảnh, siêu dữ liệu theo ngữ cảnh của các mục, v.v. Kết quả thử nghiệm mở rộng được thực hiện trên bộ dữ liệu này cho thấy rằng mô hình này đạt được hiệu suất tốt hơn so với các mô hình hiện đại về gợi ý phối trang phục, về mặt AUC, MAP và MRR. (Độ chính xác trung bình (MAP), Xếp hạng đối ứng trung bình (MRR) và Vùng dưới đường cong ROC (AUC)).

Tóm lại, những đóng góp của bài viết là:

* Khám phá ý kiến người dùng để cải thiện chất lượng đề xuất trang phục.
* Đưa ra các đề xuất về trang phục.
* Sử dụng sự chú ý lẫn nhau để mô hình hóa sự tương thích giữa các mặt hàng thời trang.
* Phương pháp đề xuất này được chứng minh là có hiệu quả trong các thử nghiệm trên bộ dữ liệu hiện có và bộ dữ liệu quy mô lớn được xây dựng có mục đích.

# 

# **CHƯƠNG 2:** **BÀI TOÁN GỢI Ý TRANG PHỤC**

Đưa ra một bức ảnh của một mặt hàng thời trang (ví dụ: phần trên), một hệ thống đề xuất trang phục cố gắng đề xuất một bức ảnh của các mặt hàng thời trang khác (ví dụ: phần dưới). Đã có một số ít nỗ lực để giải quyết nhiệm vụ. Iwata và cộng sự đề xuất một mô hình chủ đề xác suất để đề xuất phần trên cho phần dưới bằng cách tìm hiểu thông tin về tọa độ từ các tính năng trực quan trong từng khu vực mặt hàng thời trang. Liu và cộng sự nghiên cứu cả hai vấn đề đề xuất trang phục và vật phẩm. Họ đề xuất một mô hình Máy Vector Hỗ trợ tiềm ẩn cho đề xuất trang phục theo định hướng, nghĩa là, cho một đầu vào của người dùng, đề xuất quần áo phù hợp nhất hoặc đề xuất các item để kết hợp với quần áo liên quan. Jagadeesh và cộng sự đề xuất hai lớp người giới thiệu trang phục, cụ thể là xác định và ngẫu nhiên, trong khi họ chủ yếu tập trung vào mô hình màu cho đề xuất trang phục.

Các nghiên cứu được liệt kê ở trên hầu hết dựa trên một bộ dữ liệu nhỏ, được chú thích thủ công, ngăn chặn sự phát triển của các mô hình phức tạp. Một số ấn phẩm gần đây đã sử dụng các nguồn khác, trong đó dữ liệu phong phú có thể được thu thập tự động, ví dụ: trong khu vực đề xuất toàn bộ trang phục được cá nhân hóa. Hu và cộng sự đề xuất một phương pháp nhân tố kéo căng chức năng để mô hình hóa các tương tác giữa người dùng và các mặt hàng thời trang qua bộ dữ liệu được thu thập từ Polyvore. McAuley và cộng sự sử dụng một khung chung để mô hình hóa sở thích trực quan của con người đối với một cặp đối tượng từ bộ dữ liệu đồng mua hàng của Amazon; họ trích xuất các tính năng trực quan với CNN và giới thiệu một số liệu tương tự để khám phá các mối quan hệ trực quan. Tương tự như vậy, He và McAuley giới thiệu một cách tiếp cận nhân tố ma trận kết hợp các tín hiệu thị giác vào các yếu tố dự đoán về ý kiến ​​của người dùng. Để xem xét thông tin theo ngữ cảnh (như tiêu đề và danh mục), Li và cộng sự phân loại một bộ trang phục nhất định là phổ biến hoặc không phổ biến thông qua hệ thống học tập sâu đa phương thức và đa thể. Để tổng hợp dữ liệu đa phương thức của các mặt hàng thời trang và thông tin theo ngữ cảnh, Song và cộng sự trước tiên sử dụng một bộ mã hóa tự động để khai thác không gian tương thích tiềm ẩn của chúng. Sau đó, họ sử dụng xếp hạng cá nhân hóa Bayes để khai thác các ưu tiên theo cặp giữa đỉnh và đáy. Kang và cộng sự sử dụng CNN để tìm hiểu các biểu diễn hình ảnh và cùng sử dụng bộ lọc cộng tác để đề xuất các mặt hàng thời trang cho người dùng. Han và cộng sự đề xuất để cùng tìm hiểu các nhúng nhúng trực quan và mối quan hệ tương thích giữa các mặt hàng thời trang theo cách từ đầu đến cuối. Họ huấn luyện một mô hình LSTM hai chiều để dự đoán tuần tự các mục tiếp theo dựa trên các mục trước để tìm hiểu các mối quan hệ tương thích của chúng. Song và cộng sự xem xét kiến ​​thức miền thời trang để kết hợp quần áo và đề xuất một kế hoạch tích hợp nó với các mạng lưới thần kinh. Và họ cũng giới thiệu một sơ đồ chu đáo để gán niềm tin quy tắc trong quy trình chưng cất kiến ​​thức.

# **CHƯƠNG 3:** **TỔNG QUAN VỀ HỌC SÂU**

## **3.1. Deep Learning**

Deep Learning là một kỹ thuật trong Machine Learning, liên quan đến các thuật toán lấy cảm hứng từ cấu trúc và hoạt động của bộ não động vật gọi là mạng thần kinh nhân tạo.

Mạng thần kinh nhân tạo là một hệ thống các chương trình và cấu trúc dữ liệu mô phòng cách vận hành của não người. Còn Machine Learning là chương trình chạy trên một mạng thần kinh nhân tạo, có khả năng huấn luyện máy tính “học” từ một lượng lớn dữ liệu được cung cấp để giải quyết những vấn đề cụ thể.

Deep Learning tập trung giải quyết các vấn đề liên quan đến mạng thần kinh nhân tạo nhằm nâng cấp các công nghệ như nhận diện giọng nói, tầm nhìn máy tính và xử lý ngôn ngữ tự nhiên. Deep Learning đã thúc đẩy tiến bộ trong đa dạng các lĩnh vực như nhận thức sự vật (object perception), dịch tự động (machine translation), nhận diện giọng nói, … những vấn đề từng rất khó khăn với các nhà nghiên cứu trí tuệ nhân tạo.

## **3.2. Convolutional Neural Network**

Convolutional Neural Network (CNNs – Mạng nơ-ron tích chập) là một trong những mô hình Deep Learning tiên tiến. Nó giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao. Như hệ thống xử lý ảnh lớn của Facebook, Google hay Amazon đã đưa vào sản phẩm của mình những chức năng thông minh như nhận diện khuôn mặt người dùng, phát triển xe hơi tự lái hay drone giao hàng tự động**.**

***Đặc trưng:***

Đặc trưng ảnh là những chi tiết xuất hiện trong ảnh, từ đơn giản như cạnh, hình khối, chữ viết tới phức tạp như mắt, mặt, chó, mèo, bàn, ghế, xe, đèn giao thông, v.v.. Bộ lọc phát hiện đặc trưng là bộ lọc giúp phát hiện và trích xuất các đặc trừng của ảnh, có thể là bộ lọc góc, cạnh, đường chéo, hình tròn, hình vuông, vv..

***Lớp tích chập (Convolutional Layers):***

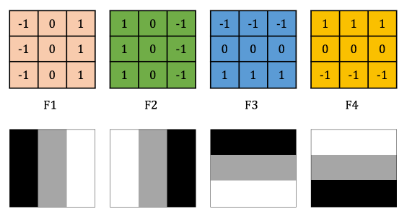
Lớp tích chập được dùng để phát hiện và trích xuất đặc trưng - chi tiết của ảnh.

Lớp tích chập lấy dữ liệu đầu vào, thực hiện các phép chuyển đổi để tạo ra dữ liệu đầu vào cho lớp kế tiếp (đầu ra của lớp này là đầu vào của lớp sau). Phép biến đổi được sử dụng là phép tính tích chập. Mỗi lớp tích chập chứa một hoặc nhiều bộ lọc - bộ phát hiện đặc trưng (filter - feature detector) cho phép phát hiện và trích xuất những đặc trưng khác nhau của ảnh.

Bộ lọc ở lớp tích chập càng sâu thì phát hiện các đặc trừng càng phức tạp.

Độ phức tạp của đặc trưng được phát hiện bởi bộ lọc tỉ lệ thuận với độ sâu của lớp tích chập mà nó thuộc về. Trong mạng CNN, những lớp tích chập đầu tiên sử dụng bộ lọc hình học (geometric filters) để phát hiện những đặc trưng đơn giản như cạnh ngang, dọc, chéo của bức ảnh. Những lớp tích chập sau đó được dùng để phát hiện đối tượng nhỏ, bán hoàn chỉnh như mắt, mũi, tóc, v.v. Những lớp tích chập sâu nhất dùng để phát hiện đối tượng hoàn hỉnh như: chó, mèo, chim, ô tô, đèn giao thông, v.v. Để hiểu cách thức hoạt động của lớp tích chập cũng như phép tính tích chập, hãy cùng xem ví dụ về bộ lọc phát hiện cạnh (edge filters/ detectors) dưới đây.

Trong ví dụ này, CNN được sử dụng để phân loại tập các ảnh viết tay của các số từ 0 tới 9. Đầu vào là những bức ảnh trắng đen (Gray Scale) và được biểu diễn bởi một ma trận các điểm ảnh với kích thước cố định h×w. Lớp tích chập đầu tiên của CNN sử dụng 4 bộ lọc kích thước 3×3 với giá trị tương ứng như trong hình 3. Các giá trị tại mỗi ô của các bộ lọc có thể được biểu diễn bởi màu sắc tương ứng với Đen (−1), Xám (0), Trắng (1) như trong hình dưới đây.



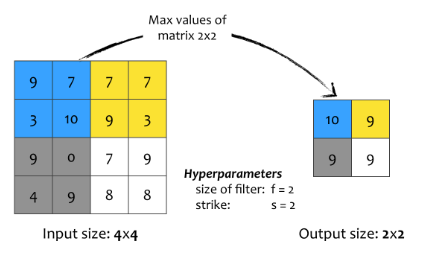
**Hình 3.1:** Bộ lọc được sử dụng trong lớp tích chập đầu tiên là các ma trận kích thước 3x3 của -1, 0 và 1.



**Hình 3.2:** Ví dụ về bộ lọc cạnh (đứng phải, đứng trái, ngang dưới, ngang trên) với đầu vào là ảnh số viết tay.

***Lớp Pooling:***

Lớp Pooling được sử dụng trong CNN để giảm kích thước đầu vào, tăng tốc độ tính toán và hiệu năng trong việc phát hiện các đặc trưng. Có nhiều hướng Pooling được sử dụng, trong đó phổ biến nhất là pooling theo giá trị cực đại (max pooling) và pooling theo giá trị trung bình (average pooling).

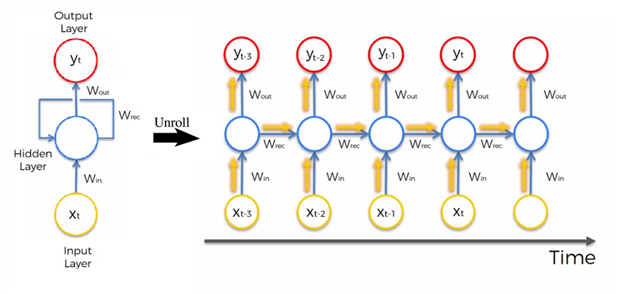


**Hình 3.3:** Ví dụ pooling theo giá trị cực đại

## **3.3. Recurrent Neural Network**

RNN (Recurrent Neural Network) là một mạng neural sử dụng chủ yếu để xử lý các bài toán ngôn ngữ tự nhiên (Natural Language) và những nhiệm vụ liên quan đến xử lý chuỗi. Ngoài ra do bản chất của RNN là tính toán tuần tự (sequencial computation) nên thích hợp cho những bài toán dự đoán về dữ liệu tuần tự như xử lý giọng nói (speech recognition), tạo ra nhạc (music generation), phân loại câu (sentiment classification), phân tích chuỗi DNA (DNA sequence analysis), dịch ngôn ngữ (Machine translation), Nhận diện hoạt động của video (Video activity recognition), vv..

Với cái nhìn tổng quát, ý tưởng chính của RNN là sử dụng chuỗi các thông tin (dữ liệu chuỗi đầu vào), thực hiện cùng một tác vụ cho tất cả các từ của một chuỗi với đầu ra phụ thuộc vào cả các phép tính trước đó. Ví dụ như khi xử lý dịch một chuỗi từ ngôn ngữ này sang ngôn ngữ khác. Các từ sẽ được lần lượt dịch từ trái sang phải, lúc này RNN sẽ dự đoán nghĩa của từ tại thời điểm đang thực hiện bằng cách căn cứ vào những từ đã được dịch trước đó. Cơ chế hoạt động này cho thấy rằng RNN có thể nhớ được thông tin qua từng giai đoạn xử lý trước đó, dựa vào đó để có thể tiếp tục xử lý cho giai đoạn tiếp theo.



**Hình 3.4:** Kiến trúc của mô hình RNN.

Tại mỗi thời điểm t, từ trong chuỗi  được đưa vào hidden layer với ký hiệu là để dự đoán ra được kết quả là . Quá trình này được thực hiện lặp lại cho đến khi dự đoán ra hết tất cả các từ trong câu. Hidden layer được xem như là “memory”, lưu lại những thông tin ở hidden layer trước đó để dự đoán cho từ, và sau đó cập nhật thông tin hiện tại để tiếp tục cho giai đoạn dự đoán tiếp theo. Trong đó hidden layer () tại thời điểm t được tính bằng công thức.

Với là các ma trận trọng số. Lấy tổng tích vô hướng (dot-product) của từ với ma trận trọng số và hidden layer ở thời điểm t-1 với ma trận trọng số . Tổng này sẽ đưa qua hàm f, đây là một hàm phi tuyến tính như tanh hay ReLU, nhằm giúp cho tinh chỉnh lại sự chênh lệch quá lớn giữa các trọng số của hidden layer httại thời điểm t sau khi thực hiện phép tính về với miền từ khoảng (-1,1).

Từ dự đoán () tại thời điểm t được tính bằng cách lấy hidden layer hiện tại nhân tích vô hướng với ma trận trọng số , đưa qua hàm softmax giúp chuẩn hóa (normalization) lại phân bổ xác suất về miền giá trị [0,1]

=Softmax()

Tại mỗi thời điểm t, điểm mất mát (loss) giữa từ thực sự (actual output) và từ được dự đoán (predicted output) được tính và tổng hợp để cập nhật lại trọng số cho các ma trận . Mất mát (loss) này được lan truyền ngược lại cập nhật những trọng số để tối giản mất mát, kỹ thuật này gọi là Back propagation through time (BPTT).

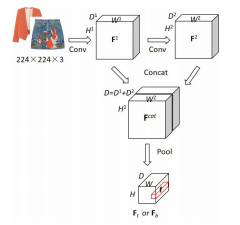
Mạng neural tái phát (RNN) có nhược điểm khi phải đối mặt với vấn đề hao hụt hoặc bùng nổ đạo hàm (Vanishing or Exploding Gradient). Khi xử ký chuỗi có kích thước lớn, lặp lại phép nhân của ma trận trọng số nhiều lần để dự đoán các từ ở chuỗi (long term) khiến cho RNN dễ đối mặt với hiện tường hao hụt hoặc bùng nổ đạo hàm (Vanisshing/Explodig gradient).

# **CHƯƠNG 4:** **GIẢI PHÁP ĐỀ XUẤT CHO BÀI TOÁN GỢI Ý TRANG PHỤC**

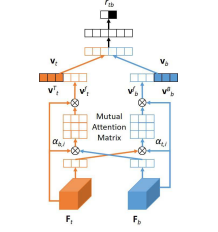
## **4.1. Tổng quan**

Cho một phần trêntừ một nhóm = {, ,. . . , }, nhiệm vụ đề xuất phần dưới là đề xuất một danh sách các phần dưới được xếp hạng từ nhóm ứng cử viên = {, ,. . . , }. Tương tự, nhiệm vụ đề xuất phần trên là đề xuất một danh sách các phần trên được xếp hạng cho một phần dưới nhất định.

Hình 4.1(a) trích xuất các đặc điểm hình ảnh từ các hình ảnh của một cặp (*t*,*b*) và chuyển đổi các đặc điểm hình ảnh thành các biểu diễn tiềm ẩn của *t* và *b* , tương ứng. Một cơ chế chú ý lẫn nhau được giới thiệu ở đây để đảm bảo rằng bộ mã hóa hình ảnh phần trên và phần dưới có thể mã hóa tính tương thích giữa *t* và *b* thành các biểu diễn tiềm ẩn của chúng. Trong hình 4.1(b), Matching Decoder là một tri giác đa lớp (MLP) để đánh giá điểm phù hợp giữa *t* và *b*.



(a) Bộ mã hóa hình ảnh trên và dưới.



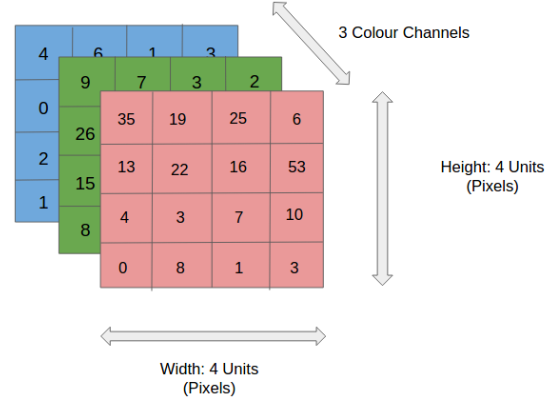
(b) Sự chú ý lẫn nhau và Matching Decoder.

**Hình 4.1:** (a) Bộ mã hóa hình ảnh trên và dưới trích xuất các tính năng hình ảnh và từ hình ảnh. (b) Sử dụng cơ chế chú ý lẫn nhau, chúng tôi chuyển đổi các tính năng trực quan sang các biểu diễn tiềm ẩn và . Sau đó, Matching Decoder dự đoán chỉ số phù hợp .

## **4.2. Bộ mã hóa hình ảnh phần trên và phần dưới**

Bộ mã hóa hình ảnh phía trên và bộ mã hóa hình ảnh phía dưới là CNN, được sử dụng rộng rãi trong khuyến nghị trang phục. Mặc dù có nhiều kiến trúc mạnh mẽ, như ResNet hoặc DenseNet, song việc đào tạo các kiến trúc này không dễ dàng, bởi vì chúng có nhiều tham số và cần nhiều dữ liệu và thời gian để đào tạo. Để cân bằng chi phí đào tạo và hiệu suất thử nghiệm, chúng ta thiết kế một CNN hai lớp với sự chú ý lẫn nhau là bộ mã hóa hình ảnh phần trên cùng và phần dưới, có ít tham số hơn và mang lại hiệu suất tốt. Chúng ta xác minh tính hiệu quả của kiến trúc mạng trong các thử nghiệm.

Input của CNN: Một bức ảnh sau khi số hoá có dạng width x height x depth (width: số lượng điểm ảnh trên chiều rộng, height: số lượng điểm ảnh trên chiều cao, depth: số lượng kênh chẳng hạn như RGB có 3 kênh đại diện cho mức độ của 3 màu Đỏ, Lục, Lam) nên input của CNN là 1 tensor 3 chiều



**Hình 4.2:** 4x4x3 RGB Image.

Cho một cặp hình ảnh (,), chúng tôi giả sử rằng hình ảnh và hình ảnh có kích thước 224 × 224 với 3 kênh màu. Như được hiển thị trong Hình 4.1 (a), chúng tôi trích xuất các tính năng trực quan từ hoặc thông qua CNN hai lớp. Cụ thể, trước tiên chúng tôi cung cấp hoặc cho một lớp chập để có các tính năng trực quan chính ∈ . Sau đó, chúng tôi đưa vào một lớp chập khác để có được các tính năng hình ảnh tiên tiến ∈ . Đưa ra các bài học với DensetNet để sử dụng các tính năng hình ảnh hiệu quả hơn trong các lớp CNN khác nhau, chúng tôi đảm bảo rằng = và = với các thao tác đệm để chúng ta có thể ghép nối và để có được ∈ . Cuối cùng, chúng tôi sử dụng nhóm tối đa trong để có được các tính năng hình ảnh cuối cùng F ∈ . Mục đích là để giảm kích thước đầu vào, tăng tốc độ tính toán và hiệu năng trong việc phát hiện các đặc trưng.

Sau đó, chúng tôi định hình lại F = [ ,. . . , ] bằng cách làm phẳng chiều rộng và chiều cao của F ban đầu, trong đó ∈ và L = W × H. Chúng ta có thể coi là các tính năng trực quan của khu vực địa phương thứ i của hình ảnh đầu vào. Cho một cặp bao gồm hình ảnh phần trên và hình phần dưới , chúng sẽ được đưa vào cùng một CNN, tức là, bộ mã hóa hình ảnh phần trên và phần dưới có cùng cấu trúc và các tham số chia sẻ. Đối với nó, các tính năng hình ảnh được trích xuất được ký hiệu như trong biểu thức. 1:

(1)

Tương tự, đối với các tính năng hình ảnh được trích xuất của hình ảnh , chúng ta có:

(2)

Các cơ chế chú ý trước đây không được thiết kế đặc biệt cho khuyến nghị trang phục, vì vậy chúng không phù hợp để mô hình hóa mối quan hệ phù hợp lẫn nhau giữa hình ảnh top và hình ảnh bottom. Chúng tôi đề xuất cơ chế chú ý lẫn nhau để đánh giá mối tương quan và sự liên kết giữa từng khu vực địa phương của và , như trong hình 7 (b). Bởi vì sự chú ý lẫn nhau có thể mô hình hóa mối quan hệ phù hợp từ hai phía, tức là từ hình ảnh dưới đến hình ảnh trên và từ hình ảnh trên đến hình ảnh dưới. Vì vậy, nó là phù hợp hơn cho khuyến nghị trang phục. Để tính trọng số chú ý từ trên xuống dưới, trước tiên chúng tôi thực hiện tổng hợp trung bình tổng quát ở , tổng hợp các tính năng hình ảnh từ tất cả các khu vực địa phương để có được các tính năng hình ảnh tổng quát của trong biểu thức. 3:

Sau đó, đối với khu vực địa phương thứ i của , chúng ta có thể tính trọng số chú ý với và như trong biểu thức. 4

(4)

Trong đó và và . Các trọng số chú ý được chuẩn hóa trong biểu thức. 5:

Sau đó, chúng tôi tính tổng trọng số của bằng để có được các attentive global visual features of :

Tương tự, chúng ta có thể tính toán các trọng số chú ý từ dưới lên trên và có được các attentive global visual features của

,

(7)

,

Sau đó, chúng tôi chiếu vào các vectơ đặc trưng trực quan và :

ReLU được dựa trên tư tưởng của việc loại bỏ bớt những tham số không quan trọng trong quá trình training và điều đó là cho mạng của chúng ta trở nên nhẹ hơn và việc training cũng nhanh chóng và có hiệu quả hơn. Hàm này thực hiện một việc rât đơn giản như sau: giữ nguyên những giá trị đầu vào lớn hơn 0, nếu giá trị đầu vào nhỏ hơn 0 thì coi là 0.

Cuối cùng, dựa trên những hiểu biết sâu sắc từ các phương pháp dựa trên yếu tố ma trận, chúng tôi cũng tìm hiểu các yếu tố tiềm ẩn top T và các yếu tố tiềm ẩn bottom B thông qua đó chúng tôi kết hợp thông tin lọc cộng tác để bổ sung cho các tính năng trực quan. Cụ thể, với mỗi đỉnh t và mỗi đáy b, chúng ta có các yếu tố tiềm ẩn và :

trong đó và . Và chúng tôi kết hợp các vectơ đặc trưng hình ảnh và các yếu tố tiềm ẩn để có được các biểu diễn tiềm ẩn và

Khi và .

## **4.3. Matching Decoder**

Như được hiển thị trong Hình 4.1(b), chúng tôi sử dụng mạng thần kinh nhiều lớp để tính xác suất khớp của t và b. Cho các đại diện tiềm ẩn và tính trong biểu thức. 10, đầu tiên chúng tôi ánh xạ và vào một không gian chung:

trong đó và và lần lượt là các ma trận ánh xạ cho và . Sau đó, chúng tôi ước tính xác suất khớp như sau:

*p()* = softmax(), (12)

trong đó cung cấp phân phối xác suất theo (tương ứng với *p()*) và = 0 (tương ứng với *p())*  . Ở đây, biểu thị rằng t và b khớp và biểu thị rằng t và b không khớp nhau. Cuối cùng, chúng tôi có thể đề xuất đỉnh hoặc đáy theo *p().*

## **4.4. Hàm mất mát**

Sử dụng negative log-likelihood (NIL) cho cả tác vụ khớp. Đối với tác vụ khớp, chúng tôi xác định hàm mất mát như sau:

trong là tập hợp các kết hợp dương, là các cặp top bottom được trích xuất từ các kết hợp trang phục trên Polyvore. là tập hợp các kết hợp âm, được hình thành bởi các đỉnh và đáy được lấy mẫu ngẫu nhiên. Ở đây, đối với các kết hợp dương, *p ()* có nghĩa là xác suất của *p ().*, tức là, cặp đã cho khớp; đối với các cặp âm, *p ().* có nghĩa là xác suất của *p ().*, tức là, cặp đã cho không khớp.

# **CHƯƠNG 5:** **CÀI ĐẶT THỬ NGHIỆM**

## **5.1 Ứng dụng vào việc phân vùng đối tượng trong ảnh thời trang**

Trong báo cáo này mô hình CNN trích xuất các đặc điểm hình ảnh của trang phục, sau đó chuyển đổi thành biểu diễn tiềm ẩn và dùng Matching Decoder dự đoán chỉ số phù hợp . Đầu vào là một ảnh thời trang, đầu ra sẽ là bức ảnh thời trang đầu vào nhưng đối tượng trong bức ảnh sẽ có một đa giác bao xung quanh.

## **5.2. Tập dữ liệu**

Tập dữ liệu được sử dụng để chạy thực nghiệm gồm hơn 28000 hình ảnh màu về trang phục.

Link database:

<https://bitbucket.org/Jay_Ren/fashion_recommendation_tkde2018_code_dataset/src/master/FashionVC/img.zip>

## **5.3. Môi trường**

Việc chạy thực nghiệm được thực hiện trên Google Colaboratory (một phiên bản Jupyter/iPython đến từ Google (think iPython + Google Drive), cung cấp cho chúng ta một môi trường notebook-based với backend Python 2/3 miễn phí), với GPU có RAM là 12,72 GB và Disk là 358,27 GB.

## **5.4. Kết quả**

* Với tập train 400 tấm hình:
  + Top:

|  |
| --- |
| Id: 177841780 |
| Id: 195136008 |
| Id: 144923126 |
| Id: 132556401 |
| Id: 160598911 |

* + Bottom:

|  |  |
| --- | --- |
|  | Score: |
| Id: 181510621 | 0.8096316 |
| Id: 197603446 | 0.6513389 |
| Id: 120648842 | 0.6276382 |
| Id: 119281562 | 0.57148856 |
| Id: 177304182 | 0.5637948 |
| Id: 175608170 | 0.55259895 |
| Id: 148069634 | 0.5450066 |
| Id: 131333384 | 0.5442911 |
| Id: 158480181 | 0.54052365 |
| Id: 183855665 | 0.5344074 |

**Bảng 5.1:** Kết quả khi train với tập gồm 400 hình ảnh thời trang

* Với tập train 5000 tấm hình:
  + Top:

|  |
| --- |
| Id: 177841780 |
| Id: 195136008 |
| Id: 144923126 |
| Id: 132556401 |
| Id: 160598911 |

* + Bottom:

|  |  |
| --- | --- |
|  | Score: |
| Id: 187130785 | 0.7296728 |
| Id: 186138466 | 0.68861955 |
| Id: 177304182 | 0.6823128 |
| Id: 197142144 | 0.6777388 |
| Id: 148069634 | 0.6675345 |
| Id: 200758526 | 0.6650808 |
| Id: 124605419 | 0.6562032 |
| Id: 172915735 | 0.6386481 |
| Id: 130754480 | 0.63174623 |
| Id: 169870788 | 0.62933105 |

**Bảng 5.2:** Kết quả khi train với tập gồm 5000 hình ảnh thời trang

* Với tập train 10000 tấm hình:
  + Top:

|  |
| --- |
| Id: 177841780 |
| Id: 195136008 |
| Id: 144923126 |
| Id: 132556401 |
| Id: 160598911 |

* + Bottom:

|  |  |
| --- | --- |
|  | Score: |
| Id: 186138466 | 0.7575436 |
| Id: 172915735 | 0.726945 |
| Id: 200359037 | 0.7176412 |
| Id: 124605419 | 0.7053268 |
| Id: 177304182 | 0.7044494 |
| Id: 119281562 | 0.69107157 |
| Id: 175608170 | 0.6907637 |
| Id: 197603446 | 0.6863883 |
| Id: 201123163 | 0.67574877 |
| Id: 183982528 | 0.66303986 |

**Bảng 5.3**: Kết quả khi train với tập gồm 10000 hình ảnh thời trang

* Với tập train 18000 tấm hình:
  + Top:

|  |
| --- |
| Id: 177841780 |
| Id: 195136008 |
| Id: 144923126 |
| Id: 132556401 |
| Id: 160598911 |

* + Bottom:

|  |  |
| --- | --- |
|  | Score: |
| Id: 177304182 | 0.7116264 |
| Id: 179538691 | 0.694852 |
| Id: 120648842 | 0.689612 |
| Id: 159267861 | 0.68788755 |
| Id: 187130785 | 0.6821649 |
| Id: 121097831 | 0.6817399 |
| Id: 183855665 | 0.674348 |
| Id: 197603446 | 0.6736614 |
| Id: 163787819 | 0.6736403 |
| Id: 105190003 | 0.66156185 |

**Bảng 5.4:** Kết quả khi train với tập gồm 18000 hình ảnh thời trang

* Với tập train 25000 tấm hình
  + Top:

|  |
| --- |
| Id: 177841780 |
| Id: 195136008 |
| Id: 144923126 |
| Id: 132556401 |
| Id: 160598911 |

* + Bottom:

|  |  |
| --- | --- |
|  | Score: |
| Id: 201189010 | 0.8316407 |
| Id: 158793076 | 0.82485515 |
| Id: 174494652 | 0.81329304 |
| Id: 18913740 | 0.7958464 |
| Id: 174193990 | 0.78493565 |
| Id: 191056758 | 0.75761986 |
| Id: 179538691 | 0.7541059 |
| Id: 177304182 | 0.74354494 |
| Id: 148069634 | 0.7343899 |
| Id: 153770786 | 0.7244516 |

**Bảng 5.5:** Kết quả khi train với tập gồm 25000 hình ảnh thời trang

# **CHƯƠNG 6:** **KẾT LUẬN**

## **6.1. Tổng kết**

Báo cáo nghiên cứu nhiệm vụ gợi ý trang phục bằng mô hình CNN với cơ chế chú ý lẫn nhau để trích xuất các đặc điểm hình ảnh của trang phục. Sử dụng bộ dữ liệu fashionCV để chứng minh sự cải tiến đáng kể về mặt MAP, MRR và AUC. Điều này chỉ ra rằng cơ chế chú ý lẫn nhau rất hữu ích cho việc gợi ý trang phục.

## **6.2. Ưu điểm, nhược điểm**

* Ưu điểm: Phương pháp đề xuất này được chứng minh là có hiệu quả trong các thử nghiệm trên bộ dữ liệu hiện có và bộ dữ liệu quy mô lớn được xây dựng có mục đích
* Nhược điểm: Mô hình còn hạn chế về sự đa dạng của các mặt hàng thời trang như các phụ kiện kính, mũ, giày, …

## **6.3. Công việc tương lai**

Trong tương lai, chúng em hi vọng có thể khám phá thêm nhiều sự kết hợp thời trang hơn nữa. Kết hợp nhiều mô hình khác để làm việc gợi ý trang phục càng ngày càng đơn giản và dễ dàng hơn.

# **TÀI LIỆU THAM KHẢO**

**Tiếng Việt**

[1] Bitcoin Vietnam News (12/2019) “Deep Learning là gì? Tiềm năng của Deep Learning”.

<https://bitcoinvietnamnews.com/deep-learning-la-gi>

[2] Deep Learning cơ bản (3/2019) “Convolutional neural network”.

<https://nttuan8.com/bai-6-convolutional-neural-network/>

[3] Do Duong (2018) “ Recurrent Neural Network”

<https://viblo.asia/p/recurrent-neural-networkphan-1-tong-quan-va-ung-dung-jvElaB4m5kw>

[4] “Mạng nơ-ron tích chập - Convolutional Neural Network (CNN)”

<https://dlapplications.github.io/2018-07-17-cnn-introduction/#2-m%E1%BA%A1ng-n%C6%A1-ron-t%C3%ADch-ch%E1%BA%ADp-convolutional-neural-network>

[5] “Nền tảng của deep learning - Multi-layer Perceptron” (15/6/2018)

<https://dlapplications.github.io/2018-06-15-MLP/>

**Tiếng Anh**

[1] (Intra-att)Ankur P. Parikh et…al (2016), A Decomposable Attention Model for Natural Language Inference”

<https://aclweb.org/anthology/D16-1244>

[2] Denny Britz (7/2015) “Understanding Convolutional Neural Networks for NLP”.

<http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/>

[3] Jason Brownlee (10/2017) “How Does Attention Work in Encoder-

Decoder Recurrent Neural Networks”

<https://machinelearningmastery.com/how-does-attention-work-in-encoder-decoder-recurrent-neural-networks/>

[4] Keitakurita (29 Dec 2017) “Explanin Attention Is All You Need”

<http://mlexplained.com/2017/12/29/attention-is-all-you-need-explained/>

[5] Lilian Weng (Jun 24 2018) “Attention? Attention!”

<https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html>

[6] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme (2009) “Bpr: Bayesian personalized ranking from implicit feedback”

<https://www.semanticscholar.org/paper/VBPR%3A-Visual-Bayesian-Personalized-Ranking-from-He-McAuley/bbb4a5e24d8b227e553b9efd6f1d62a56b7fba92>

[7] Tomoharu Iwata, Shinji Watanabe, and Hiroshi Sawada (2011) “Fashion coordinates recommender system using photographs from fashion magazines”

<https://dl.acm.org/citation.cfm?id=2283696.2283775>

[8] Xuemeng Song, Fuli Feng, Jinhuan Liu, Zekun Li, Liqiang Nie, and Jun Ma (2017) “NeuroStylist: Neural Compatibility Modeling for Clothing Matching”

<https://www.researchgate.net/publication/320542341_NeuroStylist_Neural_Compatibility_Modeling_for_Clothing_Matching>

[9] “Encoder-Decoder Architectures” (Mar 7, 2017)

<https://www.quora.com/What-is-an-Encoder-Decoder-in-Deep-Learning>

[10] SuperDataScience Team (Aug 23 2018) “Recurrent Neural Networks (RNN) - The Vanishing Gradient Problem”

<https://www.superdatascience.com/blogs/recurrent-neural-networks-rnn-the-vanishing-gradient-problem>

[11] “Understanding of RECURRENT NEURAL NETWORKS (LSTM, GRU)” (30/9/2018)

<https://mc.ai/understanding-of-recurrent-neural-networks-lstm-gru/>

[12] Wikipedia “Deep Learning”

<https://en.wikipedia.org/wiki/Deep_learning>