**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN**

**HỌC PHẦN: XỬ LÝ ẢNH VÀ THỊ GIÁC MÁY TÍNH**

**Đề tài số 07 : Xây dựng hệ thống tạo chú thích hình ảnh với CNN và LSTM**

**Giảng viên hướng dẫn: Lương Thị Hồng Lan**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Mã sinh viên** | **Sinh viên thực hiện** | **Lớp hành chính** |
| **1** | 20210256 | Chu Quốc Việt | DC.CNTT.12.10.1 |
| **2** | 20210063 | Nguyễn Thành Vinh | DC.CNTT.12.10.1 |
| **3** | 20210061 | Nguyễn Nhật Linh | DC.CNTT.12.10.1 |

**Bắc Ninh, năm 2024**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÀI TẬP LỚN**

**HỌC PHẦN: XỬ LÝ ẢNH VÀ THỊ GIÁC MÁY TÍNH**

**Đề tài số 07 : Xây dựng hệ thống tạo chú thích hình ảnh với CNN và LSTM**

**Giảng viên hướng dẫn: Lương Thị Hồng Lan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Mã sinh viên** | **Sinh viên thực hiện** | | **Lớp hành chính** |
| **1** | 20210256 | Chu Quốc Việt | | DC.CNTT.12.10.1 |
| **2** | 20210063 | Nguyễn Thành Vinh | | DC.CNTT.12.10.1 |
| **3** | 20210061 | Nguyễn Nhật Linh | | DC.CNTT.12.10.1 |
|  | | |  | | |

**Bắc Ninh, năm 2024**

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á  **KHOA CÔNG NGHỆ THÔNG TIN** | **KỲ THI KẾT THÚC HỌC PHẦN**  **HỌC KỲ 1, NĂM HỌC 2024** – **2025** |

|  |  |
| --- | --- |
| **PHIẾU CHẤM THI BÀI TẬP LỚN KẾT THÚC HỌC PHẦN**  **Mã đề thi: 07**  **Tên học phần:**  **Xử lý ảnh và thị giác máy tính**  **Lớp Tín chỉ:**  **XATGMT.03.K12.01.LH.C04.1\_LT** | |
| **Cán bộ chấm thi 1**  *(Ký và ghi rõ họ tên)*  Lương Thị Hồng Lan | **Cán bộ chấm thi 2**  *(Ký và ghi rõ họ tên)* |

| **TT** | **TIÊU CHÍ** | **THANG ĐIỂM** | **CHU QUỐC VIÊT** | **NGUYỄN THÀNH VINH** | **NGUYỄN NHẬT LINH** |
| --- | --- | --- | --- | --- | --- |
| 20210256 | 20210063 | 20210061 |
| **1** | **Nội dung báo cáo trên Word đầy đủ** | **3.5** |  |  |  |
| 1.1 | Có bố cục rõ ràng (mục lục, phần mở đầu, nội dung chính, kết luận). | 0,5 |  |  |  |
| 1.2 | Nội dung phân tích rõ ràng, logic. | 0,5 |  |  |  |
| 1.3 | Có dẫn chứng, số liệu minh họa đầy đủ. | 0,5 |  |  |  |
| 1.4 | Ngôn ngữ và trình bày chuẩn, không lỗi chính tả. | 0,5 |  |  |  |
| 1.5 | Có trích dẫn tài liệu tham khảo đúng quy cách. | 0,5 |  |  |  |
| 1.6 | Được trình bày chuyên nghiệp (canh lề, font chữ, khoảng cách dòng hợp lý). | 0,5 |  |  |  |
| 1.7 | Tài liệu đầy đủ, bám sát yêu cầu của đề bài. | 0,5 |  |  |  |
| **2** | **Nội dung thuyết trình đầy đủ** | **1.0** |  |  |  |
| 2.1 | Trình bày tự tin, phát âm rõ ràng, mạch lạc. | 0,5 |  |  |  |
| 2.2 | Nội dung thuyết trình đúng trọng tâm, không lan man. | 0,5 |  |  |  |
| **3** | **Slides báo cáo đầy đủ nội dung + Hỏi đáp** | **3.0** |  |  |  |
| 3.1 | Slides có bố cục rõ ràng (mở đầu, nội dung, kết luận). | 0,5 |  |  |  |
| 3.2 | Thiết kế slides đẹp, chuyên nghiệp (màu sắc, hình ảnh minh họa). | 0,5 |  |  |  |
| 3.3 | Nội dung trên slides ngắn gọn, dễ hiểu, súc tích. | 0,5 |  |  |  |
| 3.4 | Nội dung slides phù hợp với nội dung báo cáo. | 0,5 |  |  |  |
| 3.5 | Trả lời câu hỏi đầy đủ, chính xác. | 0,5 |  |  |  |
| 3.6 | Trả lời câu hỏi tự tin, thuyết phục. | 0,5 |  |  |  |
| **4** | **Code đầy đủ** | **2.5** |  |  |  |
| 1.1 | Code được trình bày rõ ràng, có chú thích đầy đủ. | 0,5 |  |  |  |
| 1.2 | Code chạy đúng, không lỗi. | 0,5 |  |  |  |
| 1.3 | Code tối ưu, không dư thừa. | 0,5 |  |  |  |
| 1.4 | Đáp ứng đầy đủ các yêu cầu chức năng theo đề bài. | 0,5 |  |  |  |
| 1.5 | Có tính sáng tạo hoặc cải thiện so với yêu cầu. | 0,5 |  |  |  |
| **TỔNG ĐIỂM BẰNG SỐ:** | | **10** |  |  |  |
| **TỔNG ĐIỂM BẰNG CHỮ:** | | *Mười tròn* |  |  |  |

**Mục Lục**

[**LỜI MỞ ĐẦU** 7](#_Toc184651422)

[**CHƯƠNG 1 : TỔNG QUAN** 9](#_Toc184651423)

[**1.1 Thị giác máy tính** 9](#_Toc184651424)

[1.1.1. Khái niệm 9](#_Toc184651425)

[1.1.2 Vai trò của thị giác máy tính 9](#_Toc184651426)

[1.1.3 Nguyên tắc vận hành của hệ thống thị giác máy tính 10](#_Toc184651427)

[1.1.4 Ứng dụng của thị giác máy tính 11](#_Toc184651428)

[1.1.5 Những tác vụ phổ biến mà thị giác máy tính thực hiện 12](#_Toc184651429)

[**1.2 Học sâu và ứng dụng** 14](#_Toc184651430)

[1.2.1 Giới thiệu 14](#_Toc184651431)

[1.2.2 Các mô hình của học sâu 15](#_Toc184651432)

[1.2.2.1 Mô hình RNN - Recurrent Neural 16](#_Toc184651433)

[1.2.2.2 Mô hình ANN - Artificial Neural Network 17](#_Toc184651434)

[1.2.2.3 Mô hình DNN - Deep neural network 20](#_Toc184651435)

[1.2.2.4 Mô hình CNN - Convolution Neural Network 21](#_Toc184651436)

[1.2.2.5 Mô hình DBN - Deep Belief Net 22](#_Toc184651437)

[1.2.3 Một số kiến trúc học sâu áp dụng cho vấn đề mô tả ảnh 24](#_Toc184651438)

[1.2.3.1 Kiến trúc mã hóa-giải mã (Encoder-Decoder) 24](#_Toc184651439)

[1.2.3.2 Kiến trúc Multi-Modal 24](#_Toc184651440)

[1.2.3.3 Kiến trúc Object Detection backbone 25](#_Toc184651441)

[1.2.3.4 Kiến trúc Encoder-Decoder với Attention 25](#_Toc184651442)

[1.2.3.5 Kiến trúc Encoder-Decoder với Transformers 26](#_Toc184651443)

[1.2.3.6 Kiến trúc Dense Captioning 27](#_Toc184651444)

[**1.3 Ngôn ngữ lập trình** 28](#_Toc184651445)

[1.3.1 Python 28](#_Toc184651446)

[1.3.2 Thư viện 28](#_Toc184651447)

[**CHƯƠNG 2 : XÂY DỰNG HỆ THỐNG TẠO CHÚ THÍCH ẢNH CNN VỚI LSTM** 31](#_Toc184651448)

[**2.1 Yêu cầu bài toán** 31](#_Toc184651449)

[**2.2. Các mô hình chú thích ảnh** 33](#_Toc184651450)

[2.2.1 Bài toán sinh mô tả ảnh 33](#_Toc184651451)

[2.2.2 Các kĩ thuật Deeplearning được dùng trong bài toán mô tả ảnh 33](#_Toc184651452)

[2.2.2.1 Mạng nơ ron tích chập - CNN 34](#_Toc184651453)

[2.2.2.2 LSTM 37](#_Toc184651454)

[**2.3 Chi tiết mô hình CNN – LSTM** 39](#_Toc184651455)

[**CHƯƠNG 3: THỰC NGHIỆM** 43](#_Toc184651456)

[**3.1 Dữ liệu** 43](#_Toc184651457)

[3.1.1 Mô tả dữ liệu đầu vào 43](#_Toc184651458)

[3.1.2 Xử lý dữ liệu 43](#_Toc184651459)

[**3.2 Kết quả mô hình** 45](#_Toc184651460)

[3.2.1 Kết quả huấn luyện mô hình 45](#_Toc184651461)

[3.2.2 Kết quả sinh mô tả ảnh 46](#_Toc184651462)

[**KẾT LUẬN** 47](#_Toc184651463)

[**TÀI LIỆU THAM KHẢO** 48](#_Toc184651464)

# **LỜI MỞ ĐẦU**

Trong thời đại công nghệ 4.0, với sự phát triển mạnh mẽ của trí tuệ nhân tạo (AI) và học sâu (Deep Learning), việc tự động hóa các tác vụ phức tạp đã trở nên khả thi hơn bao giờ hết. Một trong những lĩnh vực nhận được sự quan tâm đặc biệt là xử lý ngôn ngữ tự nhiên (Natural Language Processing - NLP) và xử lý ảnh (Computer Vision), hai lĩnh vực quan trọng với tiềm năng ứng dụng lớn. Sự kết hợp giữa hai lĩnh vực này đã mở ra khả năng xây dựng các hệ thống tạo chú thích hình ảnh (Image Captioning), nơi máy tính có thể hiểu và mô tả nội dung hình ảnh bằng ngôn ngữ tự nhiên.

Hệ thống tạo chú thích hình ảnh là một trong những ứng dụng nổi bật, thể hiện khả năng kết hợp sức mạnh của mạng nơ-ron tích chập (Convolutional Neural Networks - CNN) trong trích xuất đặc trưng hình ảnh và mạng nơ-ron hồi tiếp dài ngắn hạn (Long Short-Term Memory - LSTM) trong việc xử lý chuỗi dữ liệu ngôn ngữ. Các mô hình này không chỉ được sử dụng trong lĩnh vực giải trí mà còn có vai trò quan trọng trong hỗ trợ người khuyết tật, xây dựng hệ thống tìm kiếm thông minh và nâng cao trải nghiệm người dùng trong các nền tảng thương mại điện tử.

Mục tiêu của nghiên cứu này là xây dựng một hệ thống tạo chú thích hình ảnh dựa trên mô hình CNN và LSTM, với khả năng tự động phân tích nội dung hình ảnh và tạo ra các chú thích tương ứng. Qua đó, nghiên cứu không chỉ góp phần vào sự phát triển của lĩnh vực AI, mà còn đề xuất các giải pháp ứng dụng thực tiễn trong nhiều lĩnh vực đời sống.

Bài tiểu luận sẽ tập trung trình bày các cơ sở lý thuyết, quy trình triển khai hệ thống, cùng với kết quả thực nghiệm để minh họa hiệu quả của mô hình. Đồng thời, chúng tôi sẽ thảo luận về các thách thức và tiềm năng phát triển của hệ thống trong tương lai.

|  |  |  |
| --- | --- | --- |
| **STT** | **Hình** | **Trang** |
| **1** | *Hình 1. Phát hiện đối tượng* | 12 |
| **2** | *Hình 2.Trích xuất ảnh dựa trên nội dung* | 14 |
| **3** | *Hình 3.Các mô hình học sâu* | 15 |
| **4** | *Hình 4. Mô hình RNN* | 16 |
| **5** | *Hình 5. Cấu trúc mạng noron nhân tạo* | 17 |
| **6** | *Hình 6. Kiến trúc Artificial Neural Network* | 19 |
| **7** | *Hình 7. Kiến trúc mạng DNN* | 20 |
| **8** | *Hình 8. Kiến trúc mạng CNN* | 22 |
| **9** | *Hình 9. Kiến trúc mạng DBN* | 23 |
| **10** | *Hình 10. Kiến trúc mã hóa-giải mã (Encoder-Decoder)* | 24 |
| **11** | *Hình 11. Kiến trúc Multi-Modal* | 25 |
| **12** | *Hình 12. Kiến trúc Object Detection backbone* | 25 |
| **13** | *Hình 13. Kiến trúc Encoder-Decoder với Attention* | 26 |
| **14** | *Hình 14. Kiến trúc Encoder-Decoder với Transformers* | 27 |
| **15** | *Hình 15. Kiến trúc Dense Captioning* | 28 |
| **16** | *Hình 16: Ví dụ minh họa về bài toán sinh mô tả cho ảnh* | 33 |
| **17** | *Hình 17: Kiến trúc mạng tổng quát cho bài toán phân lớp* | 36 |
| **18** | *Hình 18: Cấu trúc LSTM* | 38 |
| **19** | *Hình 19: Mô hình Encoder-decoder cho vấn đề mô tả/chú thích ảnh* | 39 |
| **20** | *Hình 20: Mô hình đề xuất sinh mô tả cho ảnh* | 40 |
| **21** | *Hình 21: Kết quả chú thích ảnh* | 40 |
| **22** | *Hình 22 Mô hình Bộ mã hóa đặc trưng của ảnh* | 41 |
| **23** | *Hình 23 Mô hình Bộ giải mã trình tự* | 42 |
| **24** | *Hình 24: Hình ảnh dữ liệu thô được thu thập chưa qua xử lý* | 43 |
| **25** | *Hình 25: Các tiêu đề tương ứng cho ảnh* | 43 |

# **CHƯƠNG 1 : TỔNG QUAN**

# **1.1 Thị giác máy tính**

1.1.1. Khái niệm

Thị giác máy tính (tiếng Anh: Computer vision), là một lĩnh vực trong khoa học máy tính và trí tuệ nhân tạo (AI) tập trung vào việc phát triển phương pháp và công nghệ để máy tính có khả năng nhận diện, hiểu và xử lý hình ảnh và video một cách tự động. Thị giác máy tính sử dụng các thuật toán và mô hình học máy để phân tích và rút trích thông tin từ dữ liệu hình ảnh, như các đặc trưng, đối tượng, mối quan hệ và bối cảnh.

Các ứng dụng của thị giác máy tính có sự đa dạng đáng kể. Ví dụ, trong lĩnh vực xe tự vận hành, thị giác máy tính được sử dụng để nhận diện và phân loại các đối tượng trên đường, như ô tô, người đi bộ và biển báo giao thông, từ đó giúp xe tự động đưa ra quyết định an toàn. Trong công nghiệp sản xuất, thị giác máy tính có thể được áp dụng để kiểm tra chất lượng sản phẩm, phát hiện lỗi và tự động hóa quy trình kiểm tra. Ngoài ra, thị giác máy tính còn có thể được sử dụng trong y tế để phát hiện bệnh lý từ hình ảnh y học, trong an ninh để nhận diện khuôn mặt và phân loại hành vi đáng ngờ, và trong nhiều lĩnh vực khác.

Thông qua việc phân tích và hiểu hình ảnh và video, thị giác máy tính mang lại nhiều lợi ích và tiềm năng trong cuộc sống hàng ngày và các ngành công nghiệp khác nhau. Tuy nhiên, việc xử lý dữ liệu hình ảnh và video đòi hỏi tính toán phức tạp và sự phát triển của các thuật toán và mô hình học máy, để đảm bảo độ chính xác và hiệu quả trong việc nhận diện và phân tích hình ảnh.

### 1.1.2 Vai trò của thị giác máy tính

Thị giác máy tính quan trọng vì nó mang lại sự tự động hóa và hiệu quả trong việc xử lý và hiểu dữ liệu hình ảnh và video. Trước đây, việc xử lý thông tin hình ảnh yêu cầu sự can thiệp thủ công và tốn nhiều thời gian, dẫn đến khó khăn và lỗi trong quy trình. Với sự phát triển của thị giác máy tính và tăng cường sức mạnh tính toán, quy trình này đã được tự động hóa và cải thiện đáng kể.

Trước đây, việc triển khai hệ thống nhận diện khuôn mặt đòi hỏi con người phải gắn thẻ hàng ngàn hình ảnh bằng cách xác định các điểm dữ liệu chính như sống mũi và khoảng cách giữa hai mắt. Tuy nhiên, với sự phát triển của thị giác máy tính, các nhiệm vụ này có thể được tự động hóa, giúp tiết kiệm thời gian và công sức. Đồng thời, sự phát triển của công nghệ điện toán đám mây đã làm cho thị giác máy tính trở nên dễ tiếp cận và sử dụng cho mọi tổ chức.

Hiện nay, thị giác máy tính có nhiều ứng dụng rộng rãi và quan trọng. Ví dụ, nó có thể áp dụng để xác thực danh tính, duyệt nội dung, phân tích video trực tuyến và phát hiện lỗi. Các tổ chức từ khắp nơi đều có thể tận dụng công nghệ thị giác máy tính này để nâng cao hiệu quả và đáng tin cậy trong xử lý dữ liệu hình ảnh và video.

## 1.1.3 Nguyên tắc vận hành của hệ thống thị giác máy tính

**Học sâu (Deep Learning)**

Học sâu là một phương pháp máy học sử dụng mạng nơ-ron. Mạng nơ-ron học sâu bao gồm nhiều lớp mô-đun gọi là nơ-ron nhân tạo, hoạt động cùng nhau trong máy tính. Bằng cách sử dụng phép tính toán học, mạng nơ-ron tự động xử lý các khía cạnh khác nhau của dữ liệu hình ảnh và dần dần xây dựng sự hiểu biết về hình ảnh.

**Mạng nơ-ron tích chập (CNN)**

Mạng nơ-ron tích chập (CNN) được sử dụng để phân loại và hiểu dữ liệu hình ảnh. CNN phân tích hình ảnh theo từng điểm ảnh và gán nhãn cho mỗi điểm ảnh. Bằng cách sử dụng phép nhân chập, nó tạo ra các dự đoán về hình ảnh. Tương tự như cách con người nhận biết đối tượng từ xa, CNN đầu tiên xác định các đường nét và hình dạng đơn giản trước khi xem xét các chi tiết như màu sắc, cấu trúc bên trong và kết cấu. Quá trình dự đoán được lặp lại để nâng cao độ chính xác.

**Mạng nơ-ron hồi quy (RNN)**

Mạng nơ-ron hồi quy (RNN) có khả năng xử lý chuỗi hình ảnh và tìm hiểu mối liên kết giữa chúng. Trong khi Mạng Nơ-ron tích chập (CNN) được áp dụng cho việc phân tích từng hình ảnh riêng lẻ, Mạng Nơ-ron Hồi Quy (RNN) có khả năng xử lý dữ liệu video và hiểu các liên kết giữa các hình ảnh.

## 1.1.4 Ứng dụng của thị giác máy tính

**Lĩnh vực y tế:**

Thị giác máy tính đóng vai trò quan trọng trong việc hỗ trợ chẩn đoán và điều trị bệnh thông qua phân tích hình ảnh y khoa. Các công nghệ tiên tiến cho phép phát hiện sớm các bệnh lý như ung thư, tổn thương nội tạng, hoặc bệnh về tim mạch bằng cách phân tích hình ảnh X-quang, CT, hoặc MRI. Hệ thống tự động này giúp tăng độ chính xác trong chẩn đoán, giảm nguy cơ bỏ sót và hỗ trợ các bác sĩ xử lý số lượng lớn bệnh nhân một cách hiệu quả.

**Xe tự hành:**

Trong ngành công nghiệp xe tự hành, thị giác máy tính là công nghệ cốt lõi giúp xe "nhìn thấy" và hiểu môi trường xung quanh. Nó có thể nhận diện chướng ngại vật, biển báo giao thông, làn đường và thậm chí dự đoán hành vi của các phương tiện khác hoặc người đi bộ. Những khả năng này giúp xe đưa ra các quyết định lái xe an toàn và hiệu quả, đồng thời mở ra tiềm năng giảm thiểu tai nạn giao thông do lỗi con người.

**An ninh và bảo mật:**

Các hệ thống giám sát thông minh ngày nay dựa vào thị giác máy tính để phân tích hình ảnh từ camera an ninh, nhận diện khuôn mặt và phát hiện các hành vi bất thường. Tại các sân bay, nhà ga, hoặc cơ sở quan trọng, công nghệ này hỗ trợ việc kiểm soát truy cập, tìm kiếm đối tượng khả nghi, và đảm bảo an toàn một cách tự động và nhanh chóng. Ngoài ra, trong các ứng dụng thương mại như ngân hàng, thị giác máy tính còn giúp ngăn chặn gian lận và bảo mật dữ liệu khách hàng.

**Thương mại điện tử:**

Thị giác máy tính được sử dụng để cải thiện trải nghiệm mua sắm trực tuyến. Các nền tảng thương mại điện tử sử dụng công nghệ này để phân tích hình ảnh sản phẩm do người dùng tải lên, từ đó đề xuất các sản phẩm tương tự hoặc phù hợp với nhu cầu của họ. Điều này không chỉ tăng hiệu quả bán hàng mà còn nâng cao sự hài lòng của khách hàng.

**Giải trí và truyền thông:**

Trong lĩnh vực giải trí, thị giác máy tính giúp tối ưu hóa xử lý và chỉnh sửa hình ảnh, tạo hiệu ứng đặc biệt trong phim ảnh, hoặc nhận diện cử chỉ trong trò chơi điện tử. Người dùng có thể tương tác với thiết bị thông qua chuyển động tay hoặc cơ thể, mang lại trải nghiệm sống động và chân thực hơn.

Thị giác máy tính không chỉ là công cụ hữu ích mà còn mở ra nhiều cơ hội sáng tạo, thúc đẩy sự phát triển của các ngành công nghiệp trong thời đại số hóa.

## 1.1.5 Những tác vụ phổ biến mà thị giác máy tính thực hiện

**Phân loại hình ảnh:**

Phân loại hình ảnh là quá trình mà máy tính có khả năng quan sát một hình ảnh và xác định loại hình ảnh đó thuộc về nhóm nào. Thị giác máy tính có khả năng hiểu và đánh dấu các loại hình ảnh khác nhau, ví dụ như cây cối, máy bay, hoặc tòa nhà. Một ví dụ cụ thể là camera có thể nhận diện khuôn mặt trong ảnh và tập trung vào lấy nét chính xác của khuôn mặt.

A cat and dog with text

Description automatically generated

*Hình 1. Phát hiện đối tượng*

**Phát hiện đối tượng:**

Nhận diện vật thể và phát hiện đối tượng là một tác vụ trong thị giác máy tính nhằm phát hiện và xác định vị trí của các đối tượng trong hình ảnh. Tác vụ này sử dụng phân loại để nhận biết, sắp xếp và tổ chức hình ảnh. Việc phát hiện vật thể được áp dụng trong các ứng dụng tự động và giám sát quy trình sản xuất trong lĩnh vực công nghiệp và sản xuất. Các nhà cung cấp dịch vụ và nhà sản xuất camera gia đình cũng sử dụng công nghệ phát hiện vật thể để xử lý luồng video trực tiếp từ camera, nhằm nhận biết và thông báo cho người dùng về sự xuất hiện của con người và các vật thể trong thời gian thực.

**Theo dõi đối tượng:**

Theo dõi đối tượng là quá trình sử dụng mô hình học sâu để xác định và theo dõi các đối tượng từ một danh sách đã được chỉ định. Chức năng này có nhiều ứng dụng thực tiễn rộng rãi trên nhiều lĩnh vực khác nhau. Theo dõi đối tượng bao gồm việc phát hiện đối tượng, tạo ra một khung bao quanh đối tượng, gán một ID cho đối tượng và tiếp tục theo dõi qua các khung hình. Ví dụ, việc theo dõi đối tượng có thể được áp dụng để giám sát giao thông trong thành phố, theo dõi hoạt động của con người hoặc thực hiện chụp ảnh y tế.

**Phân đoạn:**

Phân đoạn là một kỹ thuật trong thị giác máy tính được sử dụng để nhận diện các đối tượng trong hình ảnh bằng cách chia nó thành các phần khác nhau dựa trên các đặc điểm của điểm ảnh. Qua việc đơn giản hóa hình ảnh, phân đoạn có thể đặt một hình dạng hoặc đường nét để xác định đối tượng. Điều này giúp phân đoạn nhận biết nếu có nhiều hơn một đối tượng trong hình ảnh hoặc khung hình.

Ví dụ: Có một hình ảnh vườn hoa với nhiều loại hoa khác nhau. Ta có thể sử dụng phân đoạn để xác định và tách riêng các bông hoa hồng, hoa cúc và hoa tulip trong hình ảnh. Thuật toán phân đoạn sẽ xác định các vùng tương ứng với từng loại hoa và tạo ra đường viền hoặc mặt nạ cho từng đối tượng. Nhờ đó, chúng ta có thể phân biệt và phân loại các loại hoa một cách tự động và chính xác.

**Trích xuất hình ảnh dựa trên nội dung:**

Trích xuất hình ảnh dựa trên nội dung là một ứng dụng của thị giác máy tính, cho phép tìm kiếm hình ảnh cụ thể trong một cơ sở dữ liệu lớn. Kỹ thuật này dựa trên việc phân tích các siêu dữ liệu như thẻ, mô tả, nhãn và từ khóa. Truy xuất ngữ nghĩa cho phép người dùng tìm kiếm hình ảnh theo yêu cầu cụ thể, ví dụ như "tìm ảnh cây xăng", để truy xuất những hình ảnh có nội dung liên quan.

A group of people in a busy city

Description automatically generated

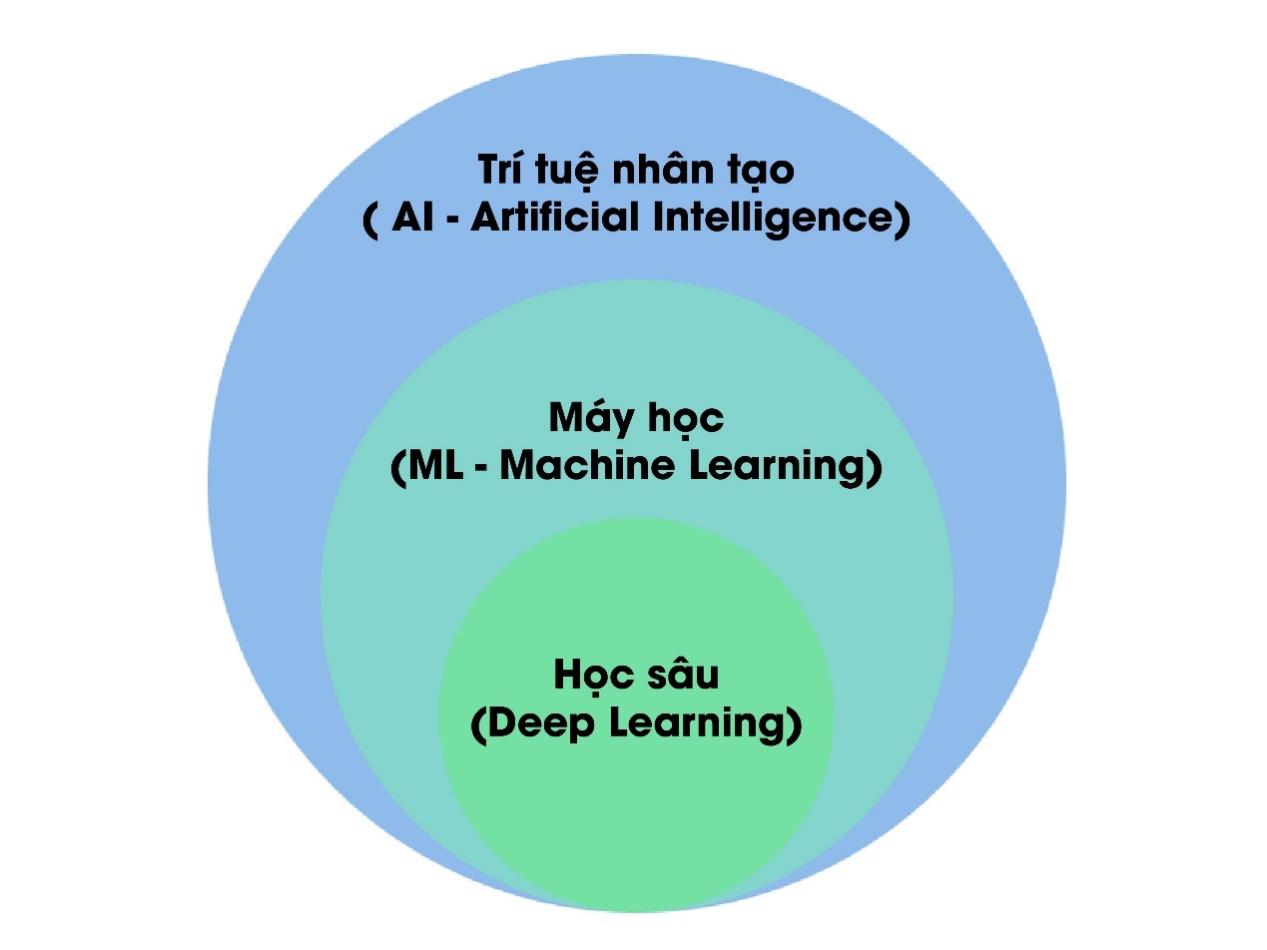
*Hình 2.Trích xuất ảnh dựa trên nội dung*

# **1.2 Học sâu và ứng dụng**

## 1.2.1 Giới thiệu

Học sâu (Deep Learning) là một nhánh của trí tuệ nhân tạo, dựa trên việc sử dụng mạng nơ-ron nhân tạo với nhiều lớp để xử lý dữ liệu và giải quyết các bài toán phức tạp. Với khả năng học từ dữ liệu lớn và không cần sự can thiệp chi tiết của con người, học sâu đã trở thành công cụ mạnh mẽ trong các ứng dụng như nhận diện hình ảnh, xử lý ngôn ngữ tự nhiên, và tạo chú thích tự động cho ảnh.

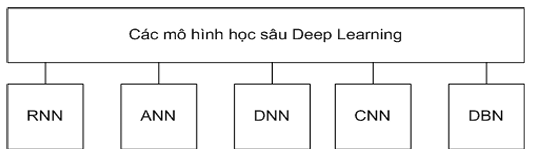
Học sâu dựa trên nguyên lý của mạng nơ-ron nhân tạo nhiều lớp (Deep Neural Networks). Mỗi lớp trong mạng học cách trích xuất các đặc trưng khác nhau từ dữ liệu đầu vào, từ các đặc trưng cơ bản ở các lớp đầu đến các đặc trưng trừu tượng hơn ở các lớp sau. Điều này cho phép học sâu hiểu sâu sắc các cấu trúc phức tạp trong dữ liệu như hình ảnh hoặc văn bản.



Trong hệ thống tạo chú thích tự động cho ảnh, học sâu đóng vai trò quan trọng trong việc nhận diện nội dung ảnh và ánh xạ chúng với văn bản phù hợp. Quá trình này thường bao gồm hai bước chính:

1. **Trích xuất đặc trưng ảnh:** Sử dụng các mô hình mạng nơ-ron tích chập (Convolutional Neural Networks - CNNs) để trích xuất các đặc trưng từ ảnh.
2. **Sinh văn bản:** Sử dụng các mô hình xử lý ngôn ngữ tự nhiên như LSTM hoặc Transformer để tạo ra mô tả văn bản từ các đặc trưng ảnh đã trích xuất.

## 1.2.2 Các mô hình của học sâu

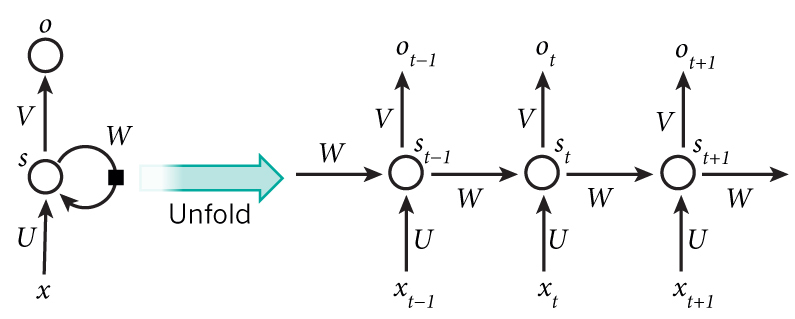


*Hình 3.Các mô hình học sâu*

### 1.2.2.1 Mô hình RNN - Recurrent Neural

Network Ý tưởng chính của RNN [2] (Recurrent Neural Network) là sử dụng chuỗi các thông tin. Trong các mạng nơ-ron truyền thống tất cả các đầu vào và cả đầu ra là độc lập với nhau. Tức là chúng không liên kết thành chuỗi với nhau. Nhưng các mô hình này không phù hợp trong rất nhiều bài toán. Ví dụ, nếu muốn đoán từ tiếp theo có thể xuất hiện trong một câu thì ta cũng cần biết các từ trước đó xuất hiện lần lượt thế nào chứ nhỉ? RNN được gọi là hồi quy (Recurrent) bởi lẽ chúng thực hiện cùng một tác vụ cho tất cả các phần tử của một chuỗi với đầu ra phụ thuộc vào cả các phép tính trước đó. Nói cách khác, RNN có khả năng nhớ các thông tin được tính toán trước đó. Trên lý thuyết, RNN có thể sử dụng được thông tin của một văn bản rất dài, tuy nhiên thực tế thì nó chỉ có thể nhớ được một vài bước trước đó mà thôi.

Về cơ bản một mạng RNN có dạng như sau:

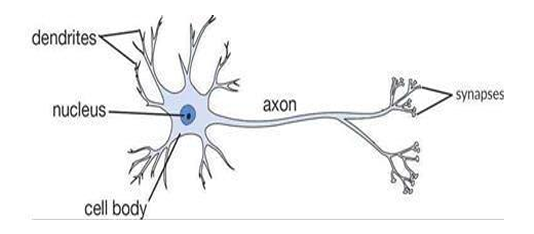


*Hình 4. Mô hình RNN*

Mô hình trên mô tả phép triển khai nội dung của một RNN. Triển khai ở đây có thể hiểu đơn giản là ta vẽ ra một mạng nơ-ron chuỗi tuần tự. Việc tính toán bên trong RNN được thực hiện như sau: − 𝑥𝑡 là đầu vào tại bước t. − 𝑠𝑡 là trạng thái ẩn tại bước t. Nó chính là bộ nhớ của mạng. 𝑠𝑡 được tính toán dựa trên cả các trạng thái ẩn phía trước và đầu vào tại bước đó: 𝑠𝑡 = 𝑓 (𝑈𝑥𝑡 + 𝑊𝑠𝑡−1). Hàm f thường là một hàm phi tuyến tính như tang hyperbolic (tanh) hay ReLu. Để làm phép toán cho phần tử ẩn đầu tiên ta cần khởi tạo them 𝑠−1, thường giá trị khởi tạo được gắn bằng 0. − 𝑂𝑡 là đầu ra tại bước t. Ứng dụng của RNN: thường được sử dụng trong Mô hình hóa ngôn ngữ và sinh văn bản; Dịch máy; Nhận dạng giọng nói; và mô tả hình ảnh.

### 1.2.2.2 Mô hình ANN - Artificial Neural Network

Artificial Neural Network [3] là một mạng nơ-ron nhân tạo, một tập hợp các thuật toán tạo ra để tìm các mối quan hệ cơ bản trong một tập dữ liệu bằng cách nắm bắt các bước trong cách thức hoạt động não bộ của con người. Có thể hiểu một mạng nơ-ron nhân tạo là một hệ thống các nơ-ron nhân tạo. Mạng nơ-ron có thể thích ứng với bất kỳ thay đổi nào với đầu vào, vì vậy có thể nhận được kết quả tốt nhất mà không nhất thiết kể lại các tiêu chí đầu ra. Khái niệm mạng nơ-ron nhân tạo ngày càng được sử dụng nhiều hơn trong sự phát triển của toàn xã hội. Hệ thống thần kinh của con người chứa các tế bào, được gọi là tế bào thần kinh (nơ ron). Các nơ-ron được kết nối với nhau bằng cách sử dụng sợi trục (axons) và sợi nhánh (dendrites), và các vùng kết nối giữa sợi trục và sợi nhánh được gọi là khớp thần kinh 16 (synapses). Các kết nối này thường thay đổi để đáp ứng với các kích thích bên ngoài. Sự thay đổi này là cách học tập diễn ra trong các sinh vật sống.



*Hình 5. Cấu trúc mạng noron nhân tạo*

Cơ chế sinh học này được mô phỏng trong các mạng nơ ron nhân tạo, chứa các đơn vị tính toán được gọi là nơ ron. Các đơn vị tính toán này được kết nối với nhau thông qua các trọng số, có tác dụng như các khớp thần kinh trong cơ chế sinh học. Mỗi đầu vào của một nơron được tương ứng với một trọng số, tác động đến chức năng được tính toán tại nơ ron đó.

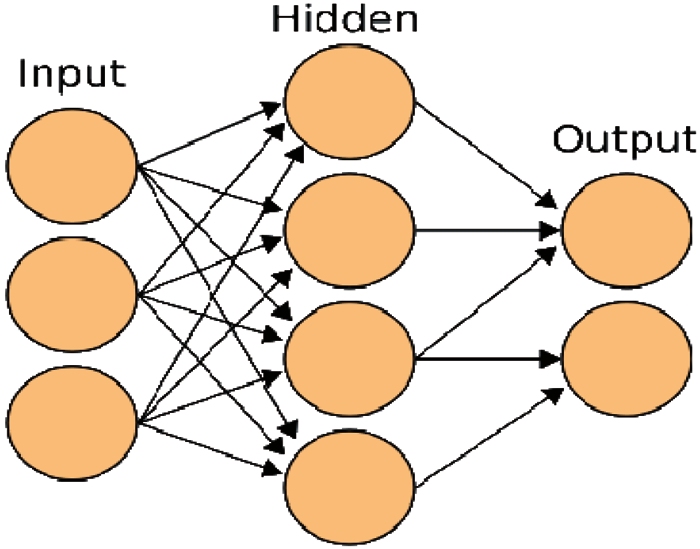
Mỗi nơ ron là một kiến trúc tính toán gồm nhiều đầu vào. Mạng nơ ron là một tập hợp các nơ ron đơn lẻ được kết nối với nhau nhằm thực hiện việc tính toán một hàm của các đầu vào bằng cách truyền các giá trị được tính toán từ các nơ ron đầu vào đến các nơ ron đầu ra và sử dụng các trọng số làm tham số trung gian. Quá trình học tập của mạng nơ ron thực hiện bằng cách thay đổi các trọng số liên kết của các nơ ron. Các kích thích bên ngoài là thứ cần thiết để hình thành các thay đổi trên mạng nơ ron sinh học, và trong mạng nơ ron nhân tạo các kích thích này được biểu diễn bởi các dữ liệu huấn luyện. Dữ liệu huấn luyện là các cặp đầu vào - đầu ra của hàm cần học. Ví dụ: dữ liệu huấn luyện có thể chứa các giá trị biểu diễn pixel của hình ảnh (đầu vào) và nhãn chú thích (đầu ra) tương ứng của chúng (ví dụ: chó, mèo, quả táo, quả cam...). Các cặp dữ liệu huấn luyện này được đưa vào mạng nơ ron bằng cách sử dụng các biểu diễn đầu vào để đưa ra dự đoán về nhãn đầu ra.

Khi xảy ra các dự đoán sai, mạng nơ ron sẽ tạo ra các cơ chế phản hồi để điều chỉnh lại tham số dự đoán để đáp ứng các lỗi dự đoán nhằm đưa mô hình trở lên chính xác hơn. Mục tiêu của việc thay đổi các trọng số là sửa đổi hàm được tính toán để làm cho các 17 dự đoán lặp lại chính xác hơn. Do đó, các trọng số được thay đổi cẩn thận theo cách hợp lý về mặt toán học để giảm sai số trong tính toán trên mẫu huấn luyện đó. Bằng cách điều chỉnh liên tục các trọng số giữa các nơ-ron qua nhiều cặp đầu vào-đầu ra, hàm được tính toán bởi mạng nơ-ron được tinh chỉnh theo thời gian để cung cấp dự đoán chính xác hơn.

Do đó, nếu mạng nơ ron được huấn luyện với đủ nhiều dữ liệu, nó sẽ cho kết quả dự đoán chính xác. Chẳng hạn với nhiều hình ảnh khác nhau của con mèo, thì cuối cùng nó sẽ có thể nhận ra đúng một con mèo trong hình ảnh mà nó chưa từng thấy trước đây. Đây là lí do mạng nơ ron có thể xấp xỉ các hàm số tốt và không cần biết trước rằng hàm số đó có dạng gì, có là đa thức hay không. Và kết quả của quá trình huấn luyện sẽ là một quan hệ được biểu diễn bằng một hàm phức tạp thể hiện bằng các tham số của mạng nơ ron. Trong thực tế, các đơn vị tính toán cơ bản nhất trong mạng nơ ron được lấy cảm hứng từ các thuật toán học máy truyền thống như SVM hoặc hồi quy logistic.

Ngoài việc xấp xỉ các hàm số, hồi quy, mạng nơ ron còn có khả năng tính toán chính xác giá trị đầu ra dự đoán của các đầu vào không nhìn thấy trong dữ liệu huấn luyện hữu hạn cho trước. Khả năng này được gọi là khả năng tổng quát hóa mô hình. Đây là tính chất quan trọng của các mô hình machine learning nói chung và deep learning nói riêng.

Kiến trúc của mạng noron Mạng nơ-ron nhân tạo là sự gắn kết giữa các tầng perceptron hoặc cũng có thể gọi là perceptron đa tầng. Với mỗi mạng neural nói chung bao gồm 3 tầng: Tầng vào (input layer): Tầng này nằm ở ngoài cùng bên trái, đại diện cho giá trị đầu vào của mạng; Tầng ra (output layer): Tầng này nằm ở ngoài cùng bên phải, đại diện cho đầu ra của mạng; Tầng ẩn (hidden layer): Tầng này là tầng nằm giữa hai tầng vào và tầng ra, đại diện cho phần logic của mạng.



*Hình 6. Kiến trúc Artificial Neural Network*

Đối với mạng nơ-ron nhân tạo, thường số lượng tần ẩn sẽ có số lượng nơ-ron khác nhau. Nói cách khác, các tế bào thần kinh trong các tầng sẽ có liên kết với nhau để tạo thành một hệ thống mạng hoàn chỉnh.

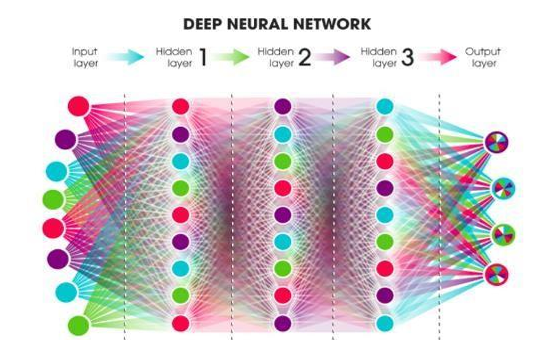
Ứng dụng của ANN:

• Tài chính - Ngân hàng: ANN hỗ trợ dự đoán những sự việc trong tương lai bằng cách sử dụng dữ liệu thu thập được từ trước. Thẩm định bất động sản, cho vay, kiểm tra tài sản thế chấp, đánh giá sự hợp tác (mức độ hợp tác), chương trình thương mại thông qua giấy tờ, dự báo tỷ giá tiền tệ, phân tích hạn mức tín dụng và thanh toán thẻ tín dụng là một số ứng dụng mà doanh nghiệp tài chính - ngân hàng sử dụng ANN.

• Hoạt ảnh và các hiệu ứng đặc biệt được sử dụng trong ngành giải trí. • Điện tử: Dự đoán mã tuần tự, điều khiển quá trình, sơ đồ chip IC, phân tích nguyên nhân lỗi chip, nhận dạng giọng nói và mô hình hóa phi tuyến là tất cả các ví dụ về chủ đề điện tử.

• Nhận dạng vũ khí, phát hiện mục tiêu, xử lý ảnh radar, nhận dạng nét mặt, cảm biến thế hệ mới, v.v. là các ứng dụng của ANN đối với quốc phòng. • Hệ thống định vị, hệ thống điều khiển chuyến bay và máy dò lỗi đều được sử dụng trong ngành hàng không.

### 1.2.2.3 Mô hình DNN - Deep neural network



*Hình 7. Kiến trúc mạng DNN*

Deep neural network (DNN) [4] là hệ thống cấu trúc thần kinh phức tạp gồm nhiều đơn vị neural network mà trong đó, ngoài các lớp nguồn vào (input), nguồn ra (output) thì có hơn một lớp ẩn (hidden layer). Mỗi lớp này sẽ thực hiện một kiểu phân loại và sắp xếp riêng trong một quá trình ta gọi là “phân cấp tính năng” và mỗi lớp đảm nhiệm một trọng trách riêng, output của lớp này sẽ là input của lớp sau. DNN được xây dựng với mục đích mô phỏng hoạt động não bộ phức tạp của con người và được áp dụng vào nhiều lĩnh vực khác nhau, mang lại thành công và những hiệu quả đáng kinh ngạc cho con người.

Ứng dụng của DNN: Dịch thuật, Phương tiện không người lái, Trợ lý ảo, chatbots/bots, Y học.

### 1.2.2.4 Mô hình CNN - Convolution Neural Network

Mạng nơ-ron nhân tạo CNN [9] là một trong những mô hình Deep learning được đánh giá có nhiều ưu điểm trong các tác vụ xử lý hình ảnh. Đây cũng là cơ sở thực hiện để xây dựng mô hình trong đề tài này.

Convolution Neural Network-CNN [5] được xem là một trong những mô hình của Deep Learning – tập hợp các thuật toán để có mô hình dữ liệu trừu tượng hóa ở mức cao bằng cách sử dụng nhiều lớp xử lý cấu trúc phức tạp. Hiểu đơn giản, CNN là một lớp của mạng nơ-ron sâu, được áp dụng phổ biến nhất để phân tích hình ảnh trực quan. Xét về cơ bản, khi xem một hình ảnh mới, thuật toán CNN sẽ không nhận biết được nó ở vị trí nào, các Feature sẽ khớp với nhau ở đâu? Chính vì vậy, Convolutional sẽ thử chúng với tất cả các vị trí khác nhau và tạo thành một bộ lọc gọi là Filter. Quá trình này được thực hiện thông qua phần toán nơ ron tích chập.



*Hình 8. Kiến trúc mạng CNN*

Trong ba lớp của Convolutional Neural N etwork, Convolutional Layer được xem là lớp có vai trò quan trọng nhất. Bởi vì Convolutional Layer sẽ đại diện CNN thực hiện mọi phép toán. Ứng dụng của CNN [6]

• Công nghiệp xe hơi: Dự đoán góc đánh lái của vô lăng. • Phân loại ảnh: là tiền đề cho rất nhiều ngành khác.

• An ninh bảo mật: Nhận diện vật thể trong camera an ninh, từ đó đưa ra các dự đoán và cảnh báo đến chủ nhà.

• Xử lý ngôn ngữ tự nhiên: Phân tích ngữ nghĩa, mô hình câu, dự đoán cũng như phân loại.

• Nhận dạng giọng nói: Gần đây, các công ty như Google đã và đang sử dụng CNN cùng với Neuron Network định kỳ và LSTM để nhận dạng giọng nói.

### 1.2.2.5 Mô hình DBN - Deep Belief Net

Deep Belief Net – DBN [7] là một mô hình mạng nơ-ron nhân tạo nhiều lớp. Quá trình huấn luyện mạng DBN gồm hai pha: tiền huấn luyện và hiệu chỉnh trọng số. Mô hình thuộc supervised và cả unsupervised learning, trong pha tiền huấn luyện, máy học Boltzmann được sử dụng để khởi tạo trọng số tốt nhất cho mô hình với dữ liệu không cần được gán nhãn. Trong pha hiệu chỉnh trọng số, DBN tiếp tục được huấn luyện bằng phương pháp lan truyền ngược cổ điển với dữ liệu được gán nhãn. Đối với IDS-Intrusion Detection System (Hệ thống phát hiện xâm nhập), DBN được sử dụng cho các nhiệm vụ trích xuất đặc trưng và phân lớp.



*Hình 9. Kiến trúc mạng DBN*

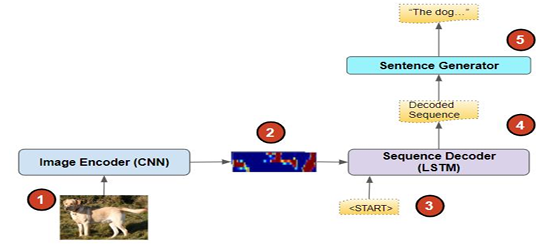
Nó là một ngăn xếp của Máy Boltzmann bị hạn chế (RBM) hoặc Máy mã tự động. Máy Boltzmann bị hạn chế (RBM) hoặc Autoencoders. Hai lớp trên cùng của DBN là kết nối vô hướng, đối xứng giữa chúng tạo thành bộ nhớ liên kết. Các kết nối giữa tất cả các lớp thấp hơn được định hướng, với các mũi tên chỉ về phía lớp gần với dữ liệu nhất. Các lớp thấp hơn đã định hướng các kết nối xoay vòng chuyển đổi bộ nhớ liên kết thành các biến quan sát. Lớp thấp nhất hoặc các đơn vị hiển thị nhận dữ liệu đầu vào. Dữ liệu đầu vào có thể là nhị phân hoặc thực. DBN không có kết nối lớp nội bộ như RBM và các đơn vị 22 ẩn đại diện cho các tính năng nắm bắt các mối tương quan có trong dữ liệu. Hai lớp được nối với nhau bằng một ma trận có trọng lượng đối xứng W. Mọi đơn vị trong mỗi lớp được kết nối với mọi đơn vị trong mỗi lớp lân cận.

Ứng dụng của DBN: Nhận dạng hình ảnh; Chuỗi video; Dữ liệu chụp chuyển động; Nhận dạng giọng nói.

## 1.2.3 Một số kiến trúc học sâu áp dụng cho vấn đề mô tả ảnh

### 1.2.3.1 Kiến trúc mã hóa-giải mã (Encoder-Decoder)

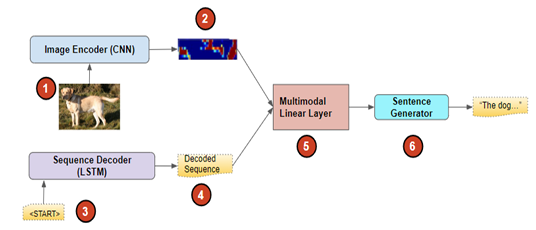
Kiến trúc học sâu phổ biến nhất cho Image Captioning đôi khi được gọi là “Inject”, kết nối trực tiếp Bộ mã hóa đặc trưng của ảnh với Bộ giải mã trình tự, tiếp theo là Bộ tạo câu, như đã mô tả ở phần trên.



*Hình 10. Kiến trúc mã hóa-giải mã (Encoder-Decoder)*

### 1.2.3.2 Kiến trúc Multi-Modal

Kiến trúc Inject là kiến trúc ban đầu cho Image Captioning và vẫn rất phổ biến. Tuy nhiên, một giải pháp thay thế được gọi là kiến trúc “Multi-Modal” đã được tìm thấy để tạo ra kết quả tốt hơn. Thay vì kết nối Bộ mã hóa hình ảnh làm đầu vào của Bộ giải mã trình tự, hai thành phần này hoạt động độc lập với nhau. Nói cách khác, ta không kết hợp hai dạng thức: hình ảnh với văn bản. Mạng CNN chỉ xử lý hình ảnh và mạng LSTM chỉ hoạt động trên chuỗi token được tạo ra.



*Hình 11. Kiến trúc Multi-Modal*

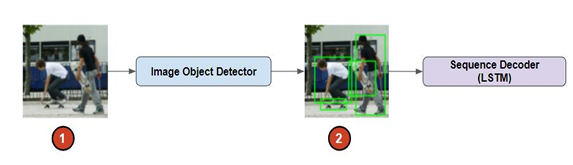
Đầu ra của hai mạng này sau đó được kết hợp với nhau bằng một lớp Đa phương thức (có thể là một lớp Linear và Softmax). Nó thực hiện công việc diễn giải cả hai kết quả đầu ra và sau đó, Trình tạo câu sẽ dự đoán chú thích cuối cùng.

Một ưu điểm khác của cách tiếp cận này là nó cho phép ta sử dụng transfer learning không chỉ cho Bộ mã hóa hình ảnh mà còn cho Bộ giải mã trình tự. Chúng ta có thể sử dụng mô hình ngôn ngữ được đào tạo trước cho Bộ giải mã trình tự.

### 1.2.3.3 Kiến trúc Object Detection backbone

Trước đó, ta đã nói về việc sử dụng “xương sống” từ mô hình Phân loại hình ảnh đào tạo trước cho Bộ mã hóa. Loại mô hình này giúp xác định một lớp duy nhất cho toàn bộ bức ảnh.

Tuy nhiên, trong hầu hết các bức ảnh, bạn có thể có nhiều đối tượng quan tâm. Thay vì sử dụng Phân loại hình ảnh, tại sao không sử dụng mô hình Phát hiện đối tượng được đào tạo trước để trích xuất các đặc trưng từ ảnh?

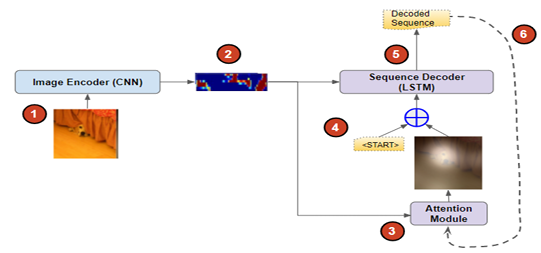


*Hình 12. Kiến trúc Object Detection backbone*

Mô hình Phát hiện Đối tượng tạo ra các hộp giới hạn xung quanh tất cả các đối tượng nổi bật trong ảnh. Nó không chỉ dán nhãn nhiều đối tượng mà còn xác định vị trí tương đối của chúng trong ảnh. Do đó, nó có thể cung cấp hình ảnh được mã hóa phong phú hơn, và đây là đầu vào để Bộ giải mã trình tự sử dụng để đưa vào chú thích của tất cả các đối tượng.

### 1.2.3.4 Kiến trúc Encoder-Decoder với Attention

Trong vài năm qua, việc sử dụng mô hình Attention với NLP đã thu hút được nhiều sự chú ý. Nó cho thấy khả năng cải thiện đáng kể hiệu suất của các ứng dụng NLP. Khi mô hình tạo ra từng từ của đầu ra, Attention giúp mô hình tập trung vào các từ trong chuỗi đầu vào có liên quan nhất đến từ đầu ra đó. Do đó, không có gì ngạc nhiên khi thấy rằng Attention cũng đã được áp dụng cho Chú thích ảnh. Khi Bộ giải mã trình tự tạo ra từng từ trong chú thích, Attention được sử dụng để giúp nó tập trung vào phần hình ảnh có liên quan nhất đến từ mà nó đang tạo.

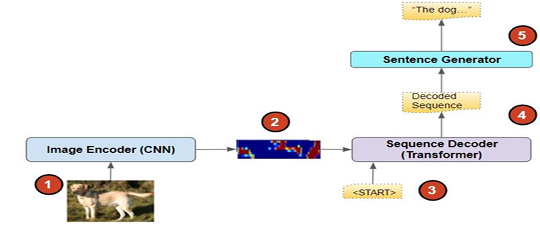


*Hình 13. Kiến trúc Encoder-Decoder với Attention*

Mô-đun Attention lấy vectơ hình ảnh được mã hóa cùng với mã token đầu ra hiện tại từ LSTM. Nó tạo ra Attention Score có trọng số. Khi điểm đó được kết hợp với hình ảnh, nó sẽ tăng trọng số của các pixel mà LSTM nên tập trung vào trong khi dự đoán mã token tiếp theo.

### 1.2.3.5 Kiến trúc Encoder-Decoder với Transformers

Khi nói về Attention, một gã khổng lồ khác chắc chắn là Transformer. Nó xoay quanh cốt lõi của Attention và không sử dụng Recurrent Network vốn là trụ cột chính của NLP trong nhiều năm. Kiến trúc rất giống với Encoder-Decoder, song Transformer thay thế cho LSTM.

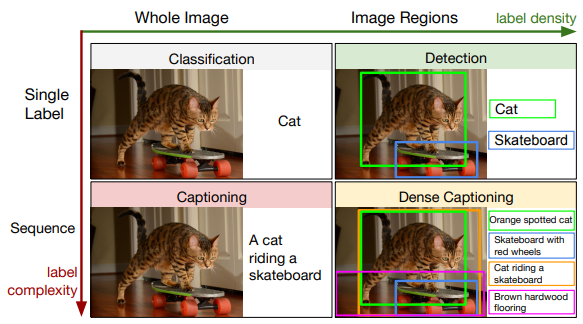


*Hình 14. Kiến trúc Encoder-Decoder với Transformers*

Một số biến thể khác nhau của kiến trúc Transformer đã được đề xuất để giải quyết bài toán Chú thích ảnh. Một cách tiếp cận mới đang cố gắng mã hóa không chỉ các đối tượng riêng lẻ trong ảnh mà còn cả các mối quan hệ không gian của chúng. Ví dụ: việc biết một đối tượng nằm dưới, phía sau hay bên cạnh một đối tượng khác sẽ cung cấp một ngữ cảnh hữu ích trong việc tạo mô tả hay chú thích.

### 1.2.3.6 Kiến trúc Dense Captioning

Một biến thể khác của phương pháp tiếp cận hướng Phát hiện Đối tượng được gọi là Dense Captioning. Ý tưởng gốc rễ đằng sau kiến trúc này là: một bức ảnh thường tập hợp phong phú các đối tượng và chúng hoạt động ở các vị trí khác nhau trong bức ảnh. Do đó, nó có thể đại diện không chỉ một chú thích đơn lẻ mà nhiều chú thích cho các vùng khác nhau của hình ảnh. Mô hình này giúp ghi lại tất cả các chi tiết trong hình ảnh.



*Hình 15. Kiến trúc Dense Captioning*

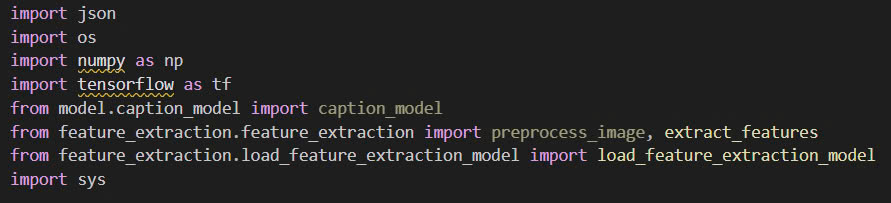
# **1.3 Ngôn ngữ lập trình**

## 1.3.1 Python

Python là một ngôn ngữ lập trình bậc cao, hướng đối tượng, được thiết kế với cú pháp đơn giản và dễ đọc, làm cho nó trở thành một trong những ngôn ngữ lập trình phổ biến nhất hiện nay. Python thường được sử dụng trong nhiều lĩnh vực, từ phát triển phần mềm, khoa học dữ liệu, trí tuệ nhân tạo, đến phát triển web và tự động hóa.

## 1.3.2 Thư viện

Các thư viện



**json**

• Chức năng: Thư viện json trong Python được sử dụng để đọc và ghi dữ liệu ở định dạng JSON. Trong đoạn mã trên, thư viện này được dùng để lưu trữ tokenizer (một đối tượng dùng để chuyển đổi văn bản thành các chỉ số số học) vào tệp JSON. Điều này giúp mô hình có thể dễ dàng tái sử dụng trong các lần chạy sau mà không cần phải huấn luyện lại tokenizer

**os**

• Chức năng: Thư viện os cung cấp các hàm giúp thao tác với hệ thống tệp và thư mục trong Python. Trong đoạn mã này, os.path được sử dụng để kiểm tra sự tồn tại của tệp và thư mục, xử lý các đường dẫn tệp, và tạo các thư mục mới nếu chúng chưa tồn tại

**numpy**

• Chức năng: numpy là thư viện phổ biến trong Python cho việc tính toán với các mảng n-dimensional. Trong đoạn mã trên, thư viện numpy được sử dụng để xử lý các mảng đặc trưng hình ảnh. Đặc trưng hình ảnh sau khi được trích xuất sẽ được chuyển đổi thành mảng numpy để phù hợp với mô hình học sâu. numpy giúp thực hiện các phép toán với mảng, như reshaping mảng hoặc xử lý các giá trị thiếu (missing values

**tensorflow**

• Chức năng: tensorflow là thư viện mã nguồn mở được sử dụng rộng rãi trong các ứng dụng học sâu (deep learning). Trong mã nguồn, tensorflow đóng vai trò quan trọng trong việc xây dựng mô hình học sâu, huấn luyện mô hình, và lưu trữ mô hình sau khi huấn luyện.

**model.caption\_model**

• Chức năng: Đây là một mô-đun có thể chứa định nghĩa mô hình captioning (tạo chú thích cho hình ảnh). Mặc dù trong đoạn mã này, caption\_model không được sử dụng trực tiếp, nhưng có thể nó chứa lớp mô hình được sử dụng để tạo chú thích cho hình ảnh trong dự án. Mô hình captioning thường bao gồm một mạng nơ-ron học sâu để dự đoán chú thích văn bản cho mỗi hình ảnh đầu vào.

**feature\_extraction.feature\_extraction**

• Chức năng: Thư viện này bao gồm các hàm dùng để tiền xử lý ảnh và trích xuất đặc trưng từ các bức ảnh. Cụ thể, preprocess\_image có thể thực hiện các bước như thay đổi kích thước ảnh, chuẩn hóa ảnh, hoặc chuyển đổi ảnh thành dạng có thể đưa vào mô hình học sâu. extract\_features là hàm trích xuất đặc trưng từ ảnh, giúp chuyển ảnh thành các vector đặc trưng, thường được sử dụng làm đầu vào cho các mô hình học sâu. Những đặc trưng này là các biểu diễn số học giúp mô hình hiểu được nội dung của ảnh.

**feature\_extraction.load\_feature\_extraction\_model**

• Chức năng: Thư viện này cung cấp các hàm để tải mô hình đã được huấn luyện từ trước, có thể là các mô hình học sâu như Inception, VGG, hoặc ResNet, dùng để trích xuất đặc trưng từ ảnh. Mô hình này thường đã được huấn luyện trên một lượng lớn dữ liệu và có khả năng trích xuất các đặc trưng có ý nghĩa từ hình ảnh đầu vào. Hàm load\_feature\_extraction\_model sẽ tải mô hình này để sử dụng cho việc trích xuất đặc trưng từ các bức ảnh trong tập dữ liệu.

**sys**

• Chức năng: Thư viện sys cung cấp các chức năng liên quan đến hệ thống và trình thông dịch Python. Sys được sử dụng để thay đổi sys.path, giúp thêm thư mục hiện tại vào danh sách tìm kiếm các mô-đun Python, từ đó giúp Python có thể tìm thấy các mô-đun mà người lập trình đã tạo ra trong cùng thư mục với mã nguồn. sys.stdout được thay đổi để in ra các ký tự Unicode, giúp hiển thị đúng các ký tự đặc biệt khi làm việc với văn bản tiếng Việt trong chương trình.

# **CHƯƠNG 2 : XÂY DỰNG HỆ THỐNG TẠO CHÚ THÍCH ẢNH CNN VỚI LSTM**

# **2.1 Yêu cầu bài toán**

Trong thời đại công nghệ số hiện nay, hình ảnh ngày càng trở nên phổ biến và đóng vai trò quan trọng trong mọi lĩnh vực, từ thương mại điện tử, truyền thông, đến các ứng dụng về sức khỏe và giáo dục. Với sự phát triển mạnh mẽ của các công nghệ trí tuệ nhân tạo, đặc biệt là trong lĩnh vực thị giác máy tính và xử lý ngôn ngữ tự nhiên, việc xây dựng hệ thống tự động tạo chú thích cho hình ảnh đã trở thành một bài toán thú vị và đầy thử thách. Hệ thống này không chỉ đơn giản là nhận diện các đối tượng trong hình ảnh mà còn phải xây dựng mối quan hệ giữa chúng, từ đó tạo ra các mô tả tự nhiên giúp người dùng hiểu được nội dung của bức ảnh mà không cần phải quan sát trực tiếp. Bài toán tạo chú thích hình ảnh, hay còn gọi là Image Captioning, là sự kết hợp giữa hai lĩnh vực quan trọng: thị giác máy tính (Computer Vision) và xử lý ngôn ngữ tự nhiên (Natural Language Processing).

Mục tiêu của bài toán này là xây dựng một hệ thống sử dụng hai mô hình học sâu phổ biến là CNN (Convolutional Neural Network) và LSTM (Long Short-Term Memory) để giải quyết vấn đề tạo chú thích hình ảnh tự động. Cụ thể, hệ thống sẽ thực hiện hai nhiệm vụ chính:

* Trích xuất đặc trưng từ hình ảnh: Sử dụng CNN để nhận diện và trích xuất các đặc trưng quan trọng từ hình ảnh. Các mô hình CNN đã được huấn luyện trước, như InceptionV3 hoặc ResNet, sẽ giúp hệ thống hiểu được các đối tượng, cảnh vật, và mối quan hệ trong bức ảnh.
* Sinh văn bản mô tả: Sau khi trích xuất đặc trưng hình ảnh, mô hình LSTM sẽ tạo ra các từ tiếp theo trong câu dựa trên đặc trưng hình ảnh và các từ đã sinh trước đó. LSTM, với khả năng duy trì thông tin qua các bước thời gian, giúp hệ thống tạo ra câu chú thích chính xác và mạch lạc.

Ví dụ, đối với một bức ảnh chứa hình ảnh một chú chó đang chơi với quả bóng trên cỏ, mô hình sẽ tạo ra câu chú thích như *"A dog is playing with a ball on the grass."*

A diagram of a model

Description automatically generated

Việc xây dựng hệ thống này không chỉ mang lại những ứng dụng thực tiễn trong việc cải thiện công cụ tìm kiếm hình ảnh, hỗ trợ người khiếm thị, hay phát triển các hệ thống sáng tạo nội dung tự động, mà còn mở ra các cơ hội ứng dụng trong nhiều lĩnh vực khác như thương mại điện tử và mạng xã hội. Tuy nhiên, bài toán này đòi hỏi phải giải quyết những thách thức lớn, như độ phức tạp của dữ liệu hình ảnh, sự đa dạng của các cảnh vật và đối tượng, cũng như yêu cầu mô hình phải tạo ra các câu chú thích tự nhiên và chính xác.

# **2.2. Các mô hình chú thích ảnh**

## 2.2.1 Bài toán sinh mô tả ảnh

Bài toán sinh mô tả/chú thích cho ảnh là phương pháp tự động tạo ra văn bản ngắn gọn để mô tả nội dung của bức ảnh đầu vào. Mục đích của chú thích hình ảnh là tự động mô tả hình ảnh bằng một hoặc nhiều câu ngôn ngữ tự nhiên. Đây là bài toán tích hợp cả hai lĩnh vực thị giác máy tính (Computer Vision) và xử lý ngôn ngữ tự nhiên (NLP), vì vậy những thách thức chính của nó phát sinh từ nhu cầu của cả 2 lĩnh vực.

A child climbing up stairs in a wooden house

Description automatically generated

*Hình 16: Ví dụ minh họa về bài toán sinh mô tả cho ảnh*

## 2.2.2 Các kĩ thuật Deeplearning được dùng trong bài toán mô tả ảnh

Mô tả ảnh sử dụng cơ chế encoder-decoder để tạo ra mô tả, chú thích cho ảnh. Cụ thể: giai đoạn mã hóa (encoder), mạng nơ-ron tích chập (CNN) thường được sử dụng để phát hiện các đối tượng trong hình ảnh do lớp tích chập cuối cùng của các mạng này cung cấp biểu diễn phong phú của hình ảnh, lớp này được sử dụng làm vectơ đặc trưng (hoặc nhiều vectơ đặc trưng thu được từ các vùng khác nhau trong ảnh). Trong giai đoạn giải mã (decoder), các mạng thần kinh hồi quy (RNN), mạng bộ nhớ ngắn dài (LSTM) thường được sử dụng để tạo câu mô tả, câu chú thích cho ảnh đầu vào.

### 2.2.2.1 Mạng nơ ron tích chập - CNN

Mạng nơ ron tích chập Mạng nơ-ron tích chập (Convolutional neural network- CNN) là một trong những mô hình deep learning tiên tiến được sử dụng phổ biến hiện nay.

a) Tích chập (Covolution)

Tích chập là phương pháp được sử dụng trong tín hiệu số. Dựa theo nguyên lý biến đổi thông tin, các nha khoa học đã sử dụng trong xử lý ảnh và video số. Convolution là kỹ thuật quan trọng trong xử lý ảnh, được sử dụng chính yếu trong các phép toán trên ảnh như: đạo hàm ảnh, làm trơn ảnh, trích xuất biên cạnh trong ảnh.

Convolution là toán tử mà ta thực hiện xoay cửa sổ 180 độ rồi sau đó áp dụng phép tương quan. Correlation là toán tử tìm sự tương quan của các vùng trên ảnh gốc. Vùng pixel trên ảnh gốc có mẫu càng giống với cửa sổ thì giá trị tại điểm tương ứng của ảnh đầu ra càng lớn.

Công thức tích chập giữa hàm ảnh f (x, y) và bộ lọc k (x, y) (kích thước mxn) được biểu diễn bởi công thức sau:

A number of mathematical symbols

Description automatically generated with medium confidence

Thành phần không thể thiếu của phép tích chập là ma trận kernel (bộ lọc). Điểm neo (anchor point) của kernel sẽ quyết định vùng ma trận tương ứng trên ảnh để tích chập, thông thường anchor point được chọn là tâm của kernel. Giá trị mỗi phần tử trên kernel được xem như là hệ số tổ hợp với lần lượt từng giá trị độ xám của điểm ảnh trong vùng tương ứng với kernel.

Phép tích chập được hình dung thực hiện bằng việc dịch chuyển ma trận kernel lần lượt qua tất cả các điểm ảnh trong ảnh, bắt đầu từ góc bên trái trên của ảnh. Và đặt anchor point tương ứng tại điểm ảnh đang xét. Ở mỗi lần dịch chuyển, thực hiện tính toán kết quả mới cho điểm ảnh đang xét bằng công thức tích chập.

Ví dụ: A white background with black text and numbers

Description automatically generated

b) Kiến trúc mạng CNN

Kiến trúc mạng được chia thành 3 chiều: rộng, cao, và sâu. Các nơ-ron trong mạng có thể không liên kết hoàn toàn với toàn bộ nơ-ron kế tiếp theo. Cuối cùng, một tầng đầu ra được tối giản thành véc-tơ của giá trị xác suất. Mạng CNN là một tập hợp các lớp Convolution chồng lên nhau và sử dụng các hàm 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.

CNNs gồm hai thành phần:

* Phần tầng ẩn hay phần rút trích đặc trưng: trong phần này, mạng sẽ tiến hành tính toán hàng loạt phép tích chập và phép hợp nhất (pooling) để phát hiện các đặc trưng. Ví dụ: nếu ta có hình ảnh con ngựa vằn, thì trong phần này mạng sẽ nhận diện các sọc vằn, hai tai, và bốn chân của nó.
* Phần phân lớp: tại phần này, một lớp với các liên kết đầy đủ sẽ đóng vai trò như một bộ phân lớp các đặc trưng đã rút trích được trước đó. Tầng này sẽ đưa ra xác suất của một đối tượng trong hình.

A diagram of different layers

Description automatically generated

*Hình 17: Kiến trúc mạng tổng quát cho bài toán phân lớp*

**CNNs có khả năng trích rút đặc trưng**

Trong cấu trúc CNN, tích chập được thực hiện trên giá trị đầu vào của dữ liệu và kernel/filter để tạo ra một bản đồ đặc trưng của ảnh. Thực hiện phép tích chập bằng cách trượt kernel/filter theo dữ liệu đầu vào. Tại mỗi vị trí, tiến hành phép nhân ma trận và tính tổng các giá trị để đưa vào bản đồ đặc trưng. Tương tự như mạng nơ-ron thông thường, ta sử dụng một hàm kích hoạt để có đầu ra dưới dạng phi tuyến. Trong quá trình trượt kernel/filter trên dữ liệu đầu vào, ta sẽ quy định một bước nhảy (stride) với mỗi lần di chuyển. Thông thường ta lựa chọn thường chọn bước nhảy là 1. Nếu kích thước bước nhảy tăng, kernel/filter sẽ có ít ô trùng lặp. Kích thước đầu ra luôn nhỏ hơn đầu vào nên ta cần một phép xử lí đầu vào để đầu ra không bị co giãn bằng cách thêm một lề ở đầu vào. Một lề với giá trị 0 sẽ được thêm vào xung quanh đầu vào trước khi thực hiện phép tích chập. Thông thường, sau mỗi tầng tích chập, ta sẽ cho kết quả đi qua một tầng hợp nhất (pooling layer). Mục đích của tầng này là để nhanh chóng giảm số chiều. Việc này giúp giảm thời gian học và hạn chế việc overfitting. Pooling layer sẽ chọn được tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling). 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 convolution từ các filter.

**Khả năng phân lớp đối tượng**

Trong phần phân lớp, ta sử dụng một vài tầng với kết nối đầy đủ để xử lí kết quả của phần tích chập. Vì đầu vào của mạng liên kết đầy đủ là 1 chiều, ta cần làm phẳng đầu vào trước khi phân lớp. Tầng cuối cùng trong mạng CNN là một tầng liên kết đầy đủ, phần này hoạt động tương tự như mạng nơ-ron thông thường.

Kết quả thu được cuối cùng cũng sẽ là một véc-tơ với các giá trị xác suất cho việc dự đoán như mạng nơ-ron thông thường.

### 2.2.2.2 LSTM

Rcurrent Neural Network (Mạng nơron hồi quy) là một thuật toán thường được áp dụng trong bài toán xử lý ngôn ngữ tự nhiên. RNN có khả năng nhớ được các thông tin đã tính toán trước đó nhờ vào việc thực hiện cùng một tác vụ cho tất cả các phần tử của một chuỗi với đầu ra phụ thuộc và các tính toán trước đó.

Với RNN, ý tưởng của thuật toán sẽ là kết nối các thông tin trước đó để tính toán cho hiện tại. Tuy nhiên thực tế có thể sẽ chỉ cần sử dụng một số thông tin gần nhất để thực hiện các tác vụ hiện tại. Nhưng đôi khi cũng có những trường hợp RNN không thể học được các thông tin quá xa hiện tại do vấn đề vanishing gradient.

Long short term memory là một dạng đặc biệt của RNN, giải quyết một phần nào vanishing gradient cho RNN. Mô hình này lần đầu được giới thiệu bởi Hochreiter & Schmidhuber (1997). Tuy rằng RNN đã chậm và LSTM còn chậm hơn do khối lượng tính toán lớn, nhưng với các cải tiến cùng khả năng tương thích với nhiều bài toán nên đang được sử dụng nhiều hơn.

LSTM được thiết kế để giải quyết vấn đề phụ thuộc dài hạn (Long term Dependency). Với mô hình RNN các layer móc nối với nhau thành các phần trong mạng nơ-ron. LSTM cũng có cấu trúc mắt xích tương tự, tuy nhiên thay vì chỉ có một layer neural network thì LSTM có bốn layer.

A diagram of a tank

Description automatically generated

*Hình 18: Cấu trúc LSTM*

Điểm khác biệt của LSTM nằm ở c (cell state), c giống như một băng chuyền để lan truyền thông tin. Cổng vào i (Input gate) quyết định lượng thông tin được đi qua cổng và ảnh hưởng đến thông tin lan truyền, quyết định được đưa ra bởi hàm sigmoid với đầu ra của hàm nằm trong khoảng [0,1]. Nếu như kết quả là 1 thông tin đa số sẽ được giữ lại còn với 0, vector thông tin sẽ bị triệt tiêu hoàn toàn. Cổng quên f (Forget gate) quyết định sẽ bỏ đi bao nhiêu thông tin đến từ trạng thái trước đó. Cổng ra o (Output gate) điều chỉnh lượng thông tin có thể ra ngoài và truyền đến trạng thái tiếp theo. Với g, thực chất là một hidden state dựa trên đầu vào xt và trạng thái ℎ𝑡−1, thay vì dùng hàm sigmoid ta dùng hàm tanh. Cuối cùng, với ct được bắt đầu từ 𝑐𝑡−1, được lọc qua f (Forget gate), cộng với hidden state g đã được lọc bởi i (Input gate), sau đó sẽ đi qua o (Output gate) để lọc thông tin một lần nữa để thu được trạng thái mới ℎ𝑡. Nếu như kết quả của i (Input gate) là 1 và f (Forget gate) là 0, ta sẽ nhận được một cấu trúc tương tự như RNN. LSTM như là một đột phá từ mô hình RNN.

# **2.3 Chi tiết mô hình CNN – LSTM**

Xuất phát từ thành công của mô hình học Sequence-to-Sequence trong vấn đề dịch máy [8] nhiều mô hình tiên tiến nhất hiện nay sử dụng kiến trúc mã hóa-giải mã (encoder decoder architecture) cho vấn đề tự động sinh mô tả/chú thích cho hình ảnh. Đặc trưng của mô hình encoder-decoder kinh điển nhất cho bài toán sinh mô tả ảnh. Trong kiến trúc này, bộ mã hóa được sử dụng để ánh xạ đầu vào thành biểu diễn vectơ chiều cố định có giá trị thực của nó. Sau đó, bộ giải mã tạo đầu ra, dựa trên biểu diễn do bộ mã hóa tạo ra. Ưu điểm chính của một hệ thống như vậy là nó có thể được đào tạo từ đầu đến cuối, nghĩa là các tham số của toàn bộ mạng được học cùng nhau, do đó tránh được vấn đề sắp xếp một số thành phần độc lập.

A diagram of a diagram of a diagram

Description automatically generated

*Hình 19: Mô hình Encoder-decoder cho vấn đề mô tả/chú thích ảnh*

Mô tả/chú thích hình ảnh được hiểu là nhiệm vụ dịch một phương thức, tức là một hình ảnh, sang một phương thức khác, tức là mô tả của nó. Vì vậy kiến trúc bộ mã hóa-giải mã đã được sử dụng với mạng thần kinh tích chập (CNN) [9] ở phía bộ mã hóa và mạng thần kinh hồi quy (RNN) [10] ở phía bộ giải mã. 34 Dựa trên mô hình kinh điển encoder-decoder sử dụng cặp CNN-RNN thì trong bài toán sinh mô tả chú thích cho ảnh đề xuất thay thế một mô hình cải tiến của RNN là LSTM ở phía phần giải mã. Mô hình đề xuất có cấu trúc như trong hình sau:

**A diagram of a sequence of code

Description automatically generated**

*Hình 20: Mô hình đề xuất sinh mô tả cho ảnh*

Mô hình đề xuất sử dụng cấu trúc encoder-decoder với phần CNN trong bộ encoder và LSTM trong decoder. CNN hoạt động như một trình trích xuất đặc trưng thường được đào tạo trước trên một tập dữ liệu lớn cho nhiệm vụ phân loại [11]. Sau đó, một bản đồ đặc trưng các feature hình ảnh từ lớp tích chập hoặc biểu diễn vectơ từ lớp được kết nối đầy đủ được sử dụng làm biểu diễn hình ảnh. Các features hình ảnh sẽ đưa vào mô hình mạng bộ nhớ ngắn hạn dài (LSTM) [12], được sử dụng để lập mô hình ngôn ngữ và sinh câu chú thích.

A screenshot of a computer

Description automatically generated

*Hình 21: Kết quả chú thích ảnh*

Như vậy, mô hình đề xuất cho bài toán mô tả/chú thích ảnh bao gồm ba thành phần chính: Bộ mã hóa đặc trưng ảnh, Bộ giải mã tuần tự và Trình tạo câu. Cụ thể như sau:

a)Bộ mã hóa đặc trưng của ảnh

Bộ mã hóa này có nhiệm vụ lấy ảnh làm nguồn đầu vào và biểu diễn các đặc điểm cốt lõi của ảnh đó.

A diagram of a diagram of a computer

Description automatically generated with medium confidence

*Hình 22 Mô hình Bộ mã hóa đặc trưng của ảnh*

Tại đây, mô hình kiến trúc CNN và học chuyển giao được sử dụng phổ biến trong bộ mã hóa để từ đó có thể trích xuất ra những đặc trưng của hình ảnh. Đối với thành phần mã hóa này có thể kể đến một số mô hình tiền huấn luyện, phục vụ việc phân loại ảnh như VGGNet, ResNet và Inception. Tuy nhiên trong đồ án này, em sử dụng mô hình VGGNet để thực hiện phân loại ảnh.

Nền tảng của bộ mã hóa này là các cấu trúc mạng CNN, từng bước trích xuất các đặc trưng khác nhau từ ảnh đầu vào và tạo ra một tóm tắt mô tả nhỏ gọn để thể hiện rõ những yếu tố quan trọng, trọng tâm nhất trong bức ảnh.

Ví dụ: Bộ mã hóa bắt đầu bằng cách trích xuất các dạng đặc trưng hình học đơn giản như ngang, dọc, đường cong và bán nguyệt trong các lớp ban đầu, sau đó tiến dần đến các cấu trúc cấp cao hơn như mũi, mắt và bàn tay, và cuối cùng xác định các yếu tố như khuôn mặt và bánh xe.

Trong cấu trúc của mô hình Phân loại hình ảnh, sau khi trích xuất được đặc trưng của ảnh thì bản đồ đặc trưng (feature map) tiếp theo sẽ được đưa đến Trình phân loại (Classifier), nhằm dự đoán đầu ra cuối cùng của lớp (ví dụ: mèo hoặc ô tô) của đối tượng chính trong hình ảnh. Tuy nhiên, khi áp dụng mô hình này cho Mô tả/chú thích ảnh, ta chỉ chú trọng đến vấn đề biểu diễn bản đồ đặc trưng của ảnh chứ không cần dự đoán phân loại. Do đó, đối với Bộ mã hóa thì các nhà phát triển thường giữ lại “khung xương” tức là khối CNN và loại bỏ đi thành phần các lớp phân loại.

b) Bộ giải mã trình tự (Sequence Decoder)

Thành phần khối giải mã tuần tự thực hiện việc biểu diễn được mã hóa của ảnh và xuất ra một chuỗi các mã tokens mô tả bức ảnh đó. Kĩ thuật thường được áp dụng đối với Bộ 36 giải mã trình tự là sử dụng mô hình Recurrent Network bao gồm các lớp LSTM (Long short term memory) được cung cấp bởi một lớp nhúng (Embedding).

A diagram of a process

Description automatically generated

*Hình 23 Mô hình Bộ giải mã trình tự*

Đầu vào của mô hình là vectơ mã hóa hình ảnh cùng mã token duy nhất: ‘Start’. Từ đây, mô hình ‘giải mã’ vector hình ảnh đầu vào và xuất ra một chuỗi các mã tokens. Quy trình này xảy ra theo dạng vòng lặp, mã token được xuất ra sau đó được đưa trở lại mạng làm đầu vào cho lần lặp tiếp theo. Cuối cùng, mô hình trả ra mã token ‘End’ hoàn thành chuỗi.

# **CHƯƠNG 3: THỰC NGHIỆM**

## **3.1 Dữ liệu**

### 3.1.1 Mô tả dữ liệu đầu vào

Tập dữ liệu được sử dụng là Flickr8k được thu thập trên trang Kaggle. Bộ dữ liệu bao gồm 2 phần chính: Ảnh và tiêu đề của các bức ảnh đó.

Mỗi một ảnh sẽ có 5 tiêu đề tương ứng cho bức ảnh đó. Trong đó bao gồm:

* Ảnh: 8091 ảnh.
* Tiêu đề: 40456 tiêu đề.

A person in a white shirt

Description automatically generated with medium confidence

*Hình 24: Hình ảnh dữ liệu thô được thu thập chưa qua xử lý*

A screen shot of a computer screen

Description automatically generated

*Hình 25: Các tiêu đề tương ứng cho ảnh*

# 3.1.2 Xử lý dữ liệu

Tiền xử lý dữ liệu là một bước rất quan trọng trong việc giải quyết các vấn đề về học máy. Hầu hết các bộ dữ liệu được sử dụng trong các vấn đề liên quan đến học máy cần được xử lý, làm sạch và chuyển đổi trước khi đào tạo các thuật toán học máy trên các bộ dữ liệu đó. Các kỹ thuật tiền xử lý dữ liệu hiện tại bao gồm: Xử lý dữ liệu bị thiếu, mã hóa các biến phân loại, chuẩn hóa dữ liệu, chia tỷ lệ dữ liệu, v.v.

Hiện nay chúng ta đang có những model lớn về xử lý ảnh như VGG, Resnet, Inception. Đây đều là những model đã được training trên tập dữ liệu cực kỳ lớn và đem lại kết quả rất tốt. Chúng ta sẽ tận dụng lại những model này để khắc phục việc dữ liệu trong bài toán này quá ít để có thể training trích xuất đặc trưng của ảnh. Kỹ thuật này gọi là Transfer learning, nó sẽ giúp chúng ta giải quyết vấn đề về thiếu dữ liệu được gán nhãn. Có thể thực hiện một công việc mới với kinh nghiệm đã học được từ những công việc cũ. Chúng ta sẽ cho ảnh trong tập dữ liệu mẫu đi qua model và thu lại đặc trưng của từng bức ảnh đó.

Ở trong đồ án này chúng ta sẽ sử dụng InceptionV3 là một trong những mô hình CNN tiên tiến, thuộc dòng Inception, được đề xuất bởi Google. Đây là mô hình đã được huấn luyện trước trên tập dữ liệu ImageNet, thường được sử dụng rộng rãi trong các bài toán Transfer Learning và trích xuất đặc trưng từ ảnh. Với cấu trúc tối ưu và khả năng học tốt cả thông tin cục bộ và toàn cục, InceptionV3 là lựa chọn lý tưởng cho nhiều bài toán xử lý ảnh phức tạp.

InceptionV3 sử dụng các khối Inception Modules, giúp tăng hiệu quả tính toán mà không làm tăng quá nhiều tham số. Các khối này áp dụng đồng thời nhiều phép lọc với kích thước kernel khác nhau (1x1, 3x3, 5x5), cho phép học được các đặc trưng ở cả phạm vi cục bộ lẫn toàn cục.

InceptionV3 sử dụng một số kỹ thuật đặc biệt giúp cải thiện hiệu suất, bao gồm:

* Factorized Convolutions: Tách các phép tích chập 5x5 thành hai phép chập liên tiếp 3x3, giảm số lượng tham số và thời gian tính toán.
* Auxiliary Classifiers: Thêm các lớp phân loại phụ trong quá trình huấn luyện để ngăn chặn overfitting.
* Label Smoothing: Sử dụng kỹ thuật làm mượt nhãn (label smoothing) để cải thiện tính tổng quát của mô hình.
* Efficient Grid Size Reduction: Sử dụng các phép chập stride 2 để giảm kích thước ma trận hiệu quả hơn thay vì dùng max-pooling.

Cụ thể, mỗi ảnh đầu vào được chuẩn hóa về kích thước 299x299 trước khi đưa vào mô hình. Mô hình sau đó thực hiện nhiều lớp tích chập, pooling và các phép toán phi tuyến tính để trích xuất đặc trưng.

Trong đồ án này, InceptionV3 được sử dụng như một mô hình pretrained, tức là sử dụng trọng số đã được huấn luyện trước trên tập dữ liệu ImageNet. Quá trình Transfer Learning diễn ra như sau:

1. Ảnh được chuẩn hóa về kích thước 299x299.
2. Dữ liệu được đưa qua mạng InceptionV3 để trích xuất các feature vectors đại diện cho các đặc trưng quan trọng.
3. Các feature vectors sau đó được sử dụng làm đầu vào cho mô hình tùy chỉnh của đồ án.

Với các bài toán kết hợp xử lý hình ảnh và văn bản (như mô tả ảnh), dữ liệu văn bản đi kèm được xử lý song song. Quy trình xử lý bao gồm:

1. Loại bỏ dấu câu, ký tự đặc biệt, và chữ số.
2. Tokenization: Chuyển đổi các từ trong văn bản thành số nguyên thông qua từ điển được tạo ra bằng Tokenizer từ Keras.
3. Padding: Đưa các câu văn về cùng độ dài (max\_length) để phù hợp với mạng nơ-ron.

## **3.2 Kết quả mô hình**

## 3.2.1 Kết quả huấn luyện mô hình

Đầu vào: 8091 ảnh và 40456 tiêu đề.

A black screen with white text

Description automatically generated

## 3.2.2 Kết quả sinh mô tả ảnh

**A screenshot of a screenshot of a caption

Description automatically generated**

# **KẾT LUẬN**

Trong tiểu luận này, chúng tôi đã nghiên cứu và triển khai thành công một hệ thống tạo chú thích hình ảnh tự động bằng cách kết hợp hai mô hình học sâu mạnh mẽ là Mạng Nơ-ron Tích Chập (CNN) và Mạng Nơ-ron Hồi tiếp Dài Ngắn Hạn (LSTM). Hệ thống sử dụng CNN để trích xuất các đặc trưng từ hình ảnh và LSTM để sinh ra các câu chú thích mô tả hình ảnh, qua đó thể hiện khả năng hiểu và mô tả một cách tự động các nội dung hình ảnh.

Kết quả thực nghiệm trên bộ dữ liệu MS COCO cho thấy hệ thống hoạt động tốt trong việc tạo ra các chú thích hình ảnh chính xác, với các chỉ số đánh giá như BLEU, METEOR và CIDEr đạt kết quả khả quan. Mô hình đã cải thiện đáng kể khả năng nhận diện các đối tượng trong các bối cảnh phức tạp, giúp sinh ra các chú thích có ngữ nghĩa và phù hợp với nội dung hình ảnh.

Tuy nhiên, hệ thống vẫn còn một số hạn chế. Đặc biệt, độ đa dạng trong việc sinh ra các câu chú thích còn hạn chế, khi hệ thống thường tạo ra các câu mô tả tương tự nhau cho những hình ảnh khác nhau. Bên cạnh đó, khả năng nhận diện các đối tượng phức tạp, đặc biệt là trong những hình ảnh có nhiều đối tượng hoặc có sự chồng lấn, vẫn chưa đạt được độ chính xác tối ưu. Mô hình cũng chưa được tối ưu hoàn toàn về tham số, và vẫn còn cơ hội để cải thiện hiệu suất bằng cách điều chỉnh các tham số của CNN hoặc LSTM.

Nhìn chung, hệ thống tạo chú thích hình ảnh với CNN và LSTM đã mở ra nhiều cơ hội ứng dụng trong các lĩnh vực như trợ lý ảo, tìm kiếm hình ảnh, và nhận diện đối tượng. Tuy nhiên, các vấn đề về độ đa dạng chú thích và khả năng nhận diện đối tượng phức tạp cần được giải quyết trong các nghiên cứu tiếp theo. Việc tích hợp các phương pháp như Attention Mechanism hay Transformer có thể là hướng đi triển vọng để nâng cao hiệu quả và khả năng ứng dụng của hệ thống này.

# **TÀI LIỆU THAM KHẢO**

[1].Tài liệu bài giảng Xử lý ảnh và thị giác máy tính , Lương Thị Hồng Lan, Đại học Công Nghệ Đông Á.

[2] A. Mathew, P. Amudha and S. Sivakumari, "Deep Learning Techniques: An Overview," in Advanced Machine Learning Technologies and Applications , 2021, pp. 599-608.

[3] P. J. Braspenning, F. Thuijsman and A. J. M. M. Weijters, Artificial Neural Networks: An Introduction to ANN Theory and Practice, Springer Berlin, Heidelberg, 1995.

[4] R. M. Cichy and D. Kaiser, " Deep Neural Networks as Scientific Models," Trends in Cognitive Sciences, vol. 23, no. 4, pp. 305-317, 2019.

[5] Z. Li, F. Liu, W. Yang, S. Peng and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 12, pp. 6999-7019, 2022.

[6] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278 - 2324, 1998.

[7] G. E. Hinton, "Deep belief networks," Scholarpedia, vol. 4, no. 5, p. 5947, 2009.

[8] I. Sutskever, O. Vinyals and L. V. Quoc, "Sequence to Sequence Learning with Neural Networks," in Advances in Neural Information Processing Systems, Curran Associates Inc, 2014, pp. 3104-3112.

[9] Y. Lecun and Y. Bengio, "Convolutional Networks for Images, Speech, and Time-Series," in The Handbook of Brain Theory and Neural Networks, 1995.

[10] J. L. Elman, "Finding structure in time," Cognitive Science, vol. 14, no. 2, p. 179–211, 1990.

[11] O. Russakovsky, J. Deng, H. Su, et al, "ImageNet Large Scale Visual Recognition Challenge," International Journal of Computer Vision, vol. 115, no. 3, p. 211–252, 2015. 62

[12] K. Papineni, S. Roukos, T. Ward and W. Zhu, "BLEU: a method for automatic evaluation of machine translation," Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, p. 311–318, 2002.