Shape, square

Description automatically generated

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN KỸ THUẬT DỮ LIỆU**

Logo

Description automatically generated

**NGUYỄN THỊ MỸ LINH – 19133032**

**VÕ THỊ NGỌC THẮM – 19133051**

Đề tài:

**ỨNG DỤNG HỌC SÂU CHO BÀI TOÁN**

**PHÂN TÍCH QUAN ĐIỂM**

**KHÓA LUẬN TỐT NGHIỆP KỸ SƯ KỸ THUẬT DỮ LIỆU**

GIẢNG VIÊN HƯỚNG DẪN

**ThS. QUÁCH ĐÌNH HOÀNG**

**Hồ Chí Minh, 2022**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN KỸ THUẬT DỮ LIỆU**

Logo

Description automatically generated

**NGUYỄN THỊ MỸ LINH – 19133032**

**VÕ THỊ NGỌC THẮM – 19133051**

Đề tài:

**ỨNG DỤNG HỌC SÂU CHO BÀI TOÁN**

**PHÂN TÍCH QUAN ĐIỂM**

**KHÓA LUẬN TỐT NGHIỆP KỸ SƯ KỸ THUẬT DỮ LIỆU**

GIẢNG VIÊN HƯỚNG DẪN

**ThS. QUÁCH ĐÌNH HOÀNG**

**Hồ Chí Minh, 2023**

|  |
| --- |
| **ĐH SƯ PHẠM KỸ THUẬT TP.HCM KHOA CNTT** \*\*\*\*\*\*\* |

**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 1: Nguyễn Thị Mỹ Linh MSSV: 19133032

Họ và tên Sinh viên 2: Võ Thị Ngọc Thắm MSSV: 19133051

Ngành: Kỹ thuật dữ liệu

Tên đề tài: ỨNG DỤNG HỌC SÂU CHO BÀI TOÁN PHÂN TÍCH QUAN ĐIỂM

Họ và tên Giáo viên hướng dẫn: ThS. Quách Đình Hoàng  
**NHẬN XÉT**1. Về nội dung đề tài khối lượng thực hiện:

..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
2. Ưu điểm:  
..................................................................................................................................................  
.................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
3. Khuyết điểm

..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
4. Đánh giá loại:

5. Điểm:

Tp*. Hồ Chí Minh,* ngày tháng năm 2023

Giáo viên hướng dẫn

*(Ký & ghi rõ họ tên)*

|  |
| --- |
| **ĐH SƯ PHẠM KỸ THUẬT TP.HCM KHOA CNTT** \*\*\*\*\*\*\* |

**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 1: Nguyễn Thị Mỹ Linh MSSV: 19133032

Họ và tên Sinh viên 2: Võ Thị Ngọc Thắm MSSV: 19133051

Ngành: Kỹ thuật dữ liệu

Tên đề tài: ỨNG DỤNG HỌC SÂU CHO BÀI TOÁN PHÂN TÍCH QUAN ĐIỂM

Họ và tên giáo viên phản biện: TS. Lê Thiên Bảo

**NHẬN XÉT**

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

..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
2. Ưu điểm:

..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
3. Khuyết điểm:

..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
..................................................................................................................................................  
4. Đề nghị cho bảo vệ hay không:

5 Đánh giá loại:

6 Điểm:

Tp*. Hồ Chí Minh,* ngày tháng năm 2023

Giáo viên hướng dẫn

*(Ký & ghi rõ họ tên)*

# LỜI CẢM ƠN

Trong quá trình nghiên cứu đề tài, chúng tôi xin bày tỏ lòng biết ơn chân thành đến Quý Thầy Cô giảng viên đã luôn hỗ trợ và chỉ dẫn chúng tôi với sự tận tâm và tôn trọng. Chúng tôi cũng muốn gửi lời cảm ơn sâu sắc đến Ban giám hiệu trường Đại học Sư phạm Kỹ Thuật Thành phố Hồ Chí Minh đã tạo điều kiện và môi trường học tập chất lượng, giúp chúng tôi có thể phát triển tối đa trong quá trình nghiên cứu đề tài.

Chúng tôi muốn gửi lời cảm ơn đặc biệt đến Ban chủ nhiệm khoa Công nghệ Thông tin và các Thầy Cô trong khoa đã cung cấp một môi trường học tập và làm việc chuyên nghiệp, nhiệt tình và hỗ trợ chúng tôi trong việc thực hiện đề tài. Điều này không chỉ đối với việc hoàn thành đề tài mà còn áp dụng cho tất cả sinh viên trong khoa Công nghệ Thông tin trong suốt quá trình học tập và làm việc tại trường.

Đặc biệt, chúng tôi muốn gửi lời cảm ơn chân thành nhất đến Thầy Quách Đình Hoàng - Giảng viên hướng dẫn khóa luận chuyên ngành – Khoa Công nghệ Thông tin – Trường Đại học Sư phạm Kỹ thuật Thành phố Hồ Chí Minh. Thầy đã dành thời gian và công sức để hướng dẫn, quan tâm và góp ý cho chúng tôi trong những giai đoạn khó khăn nhất của đề tài.

Tuy nhiên, do thời gian hoàn thành đề tài hạn chế, không thể tránh khỏi những sai sót và hạn chế nhất định. Chúng tôi mong nhận được phản hồi, đóng góp ý kiến và sự chỉ bảo từ Quý Thầy Cô để chúng tôi có thể thu thập thêm kiến thức hữu ích và nâng cao trình độ để phục vụ tốt hơn trong sự nghiệp sau này.

Chúng tôi xin chân thành cảm ơn!

**KẾ HOẠCH THỰC HIỆN**

|  |  |  |
| --- | --- | --- |
| Tuần 1, 2 | 20/03 – 02/04 | Tìm hiểu các công nghệ áp dụng vào project, tìm hiểu sơ lược các thuật toán cần dùng để xây dựng model. |
| Tuần 3, 4 | 03/04 - 16/04 | Tìm hiểu về sentiment analysis, NLP. Tiến hành thu thập và dán nhãn dữ liệu  . |
| Tuần 5, 6 | 17/04 - 28/04 | Tìm hiểu về word embedding (word2vec) và thuật toán LSTM, RNN, BERT |
| Tuần 7, 8 | 03/05 - 14/05 | Tiến hành dữ liệu và vector hóa dữ liệu với word embedding |
| Tuần 9, 10 | 15/05 - 28/05 | Tạo mô hình phân tích cảm xúc và ngôn ngữ sử dụng BERT, sử dụng kết hợp CNN và LSTM |
| Tuần 11, 12 | 29/05 – 11/06 | Sử dụng Python kết hợp với các ngôn ngữ HTML, CSS và JavaScript để xây dựng ứng dụng Phân tích ý kiến người dùng theo khía cạnh. |
| Tuần 13, 14 | 12/06 - 25/06 | Tiến hành quá trình kiểm thử, điều chỉnh tham số để cải thiện mô hình và tiếp tục viết báo cáo |
| Tuần 15, 16 | 26/06 – 09/07 | Kiểm tra lần cuối, hoàn thành báo cáo. Trao đổi với thầy hướng dẫn về nội dung cần trình bày. |

MỤC LỤC

[Phần 1: MỞ ĐẦU 1](#_Toc140037171)

[1. TÍNH CẤP THIẾT CỦA ĐỀ TÀI 1](#_Toc140037172)

[2. MỤC TIÊU VÀ NHIỆM VỤ NGHIÊN CỨU 2](#_Toc140037173)

[3. CÁCH TIẾP CẬN VÀ PHƯƠNG PHÁP NGHIÊN CỨU 2](#_Toc140037174)

[4. KẾT QUẢ DỰ KIẾN ĐẠT ĐƯỢC 4](#_Toc140037178)

[Phần 2: NỘI DUNG 5](#_Toc140037179)

[Chương 1: CƠ SỞ LÝ THUYẾT 5](#_Toc140037180)

[1.1. TỔNG QUAN VỀ PHÂN TÍCH CẢM XÚC VÀ QUAN ĐIỂM (SENTIMENT ANALYSIS) 5](#_Toc140037181)

[1.1.1. Khái niệm phân tích quan điểm 5](#_Toc140037182)

[1.1.2. Lợi ích của phân tích cảm xúc và quan điểm 5](#_Toc140037183)

[1.1.3. Các cấp độ phân tích cảm xúc và quan điểm 6](#_Toc140037184)

[1.1.4. Phân tích quan điểm trên cấp độ khía cạnh 7](#_Toc140037185)

[1.1.5. Những trường hợp sử dụng phân tích cảm xúc và quan điểm 9](#_Toc140037186)

[1.1.6. Thách thức của bài toán phân tích cảm xúc và quan điểm 10](#_Toc140037187)

[1.2. TỔNG QUAN VỀ XỬ LÍ NGÔN NGỮ TỰ NHIÊN (NLP) 11](#_Toc140037188)

[1.2.1. Xử lí ngôn ngữ tự nhiên 11](#_Toc140037189)

[1.2.2. Tầm quan trọng của xử lí ngôn ngữ tự nhiên 12](#_Toc140037190)

[1.2.3. Một vài ứng dụng của xử lí ngôn ngữ tự nhiên 12](#_Toc140037191)

[1.3. WORD EMBEDDING 13](#_Toc140037192)

[1.3.1. Term Frequency – Inverse Document Frequency (TF-IDF) 14](#_Toc140037193)

[1.3.2. Word2Vec 16](#_Toc140037194)

[1.3.3. Glove 19](#_Toc140037195)

[1.3.4. FastText 21](#_Toc140037196)

[1.3.5. BERT 22](#_Toc140037197)

[1.4. TỔNG QUAN VỀ DEEP LEARNING 23](#_Toc140037198)

[1.4.1. Deep Learning 23](#_Toc140037199)

[1.4.2. Một vài ứng dụng của Deep Learning 24](#_Toc140037200)

[1.4.3. Cách thức hoạt động của Deep Learning 25](#_Toc140037201)

[1.4.4. Convolutional Neural Network (CNNs) 25](#_Toc140037202)

[1.4.5. Recurrent neural network (RNNs) 28](#_Toc140037203)

[1.4.6. Long short-term memory (LSTMs) 29](#_Toc140037204)

[1.4.7. Transformer 30](#_Toc140037205)

[1.4.8. Mô hình BERT dựa trên kiến trúc Transformer 33](#_Toc140037206)

[Chương 2: XÂY DỰNG MÔ HÌNH PHÂN TÍCH QUAN ĐIỂM 37](#_Toc140037207)

[2.1. BÀI TOÁN 37](#_Toc140037208)

[2.2. DỮ LIỆU 37](#_Toc140037209)

[2.3. XÂY DỰNG MÔ HÌNH 39](#_Toc140037210)

[2.3.1. Tiền xử lý dữ liệu 39](#_Toc140037211)

[2.3.2. Vector hóa dữ liệu 40](#_Toc140037212)

[2.3.3. Xây dựng và huấn luyện mô hình mạng neural xử lý ngôn ngữ tự nhiên 41](#_Toc140037213)

[2.4. XÂY DỰNG ỨNG DỤNG CHO MÔ HÌNH PHÂN TÍCH QUAN ĐIỂM DỰA TRÊN KHÍA CẠNH 45](#_Toc140037214)

[2.4.1. Thiết kế 45](#_Toc140037215)

[2.4.2. Thực hiện 47](#_Toc140037216)

[Phần 3: KẾT LUẬN 49](#_Toc140037217)

[1. KẾT QUẢ ĐẠT ĐƯỢC 49](#_Toc140037218)

[2. ƯU ĐIỂM VÀ NHƯỢC ĐIỂM 49](#_Toc140037219)

[3. HƯỚNG PHÁT TRIỂN 50](#_Toc140037220)

[4. LỜI KẾT 51](#_Toc140037221)

[TÀI LIỆU THAM KHẢO 53](#_Toc140037222)

# DANH MỤC HÌNH ẢNH

[Hình 1.1: Kiến trúc CNN phân loại câu [19] 23](#_Toc139926926)

[Hình 1.2: Kiến trúc mạng RNN [20] 24](#_Toc139926928)

[Hình 1.3: Kiến trúc của một node trong mạng LSTM [20] 26](#_Toc139926930)

[Hình 1.4. Kiến trúc mô hình Transformer [21] 27](#_Toc139926932)

[Hình 2.1: Minh họa về tập dữ liệu 35](#_Toc139926937)

[Hình 3.1: Minh họa các đánh giá sau khi tiền xử lý 36](#_Toc139926940)

[Hình 3.2. Kiến trúc mô hình CNN\_LSTM 39](#_Toc139926943)

[Hình 3.2: Kiến trúc mô hình PhoBert 41](#_Toc139926944)

[Hình 3.3: Kết quả mô hình PhoBert trên tập test 41](#_Toc139926945)

[Hình 3.4: Kiểm tra kết quả trên tập test 41](#_Toc139926946)

[Hình 4.1. Phân tích thiết kế giao diện 43](#_Toc139926949)

[Hình 4.2. Phân tích thiết kế chức năng 43](#_Toc139926950)

[Hình 4.3. Giao diện trang phân tích bình luận 44](#_Toc139926952)

[Hình 4.4. Giao diện trang thống kê dữ liệu 44](#_Toc139926954)

# 

# Phần 1: MỞ ĐẦU

## 1. TÍNH CẤP THIẾT CỦA ĐỀ TÀI

Trong những năm gần đây, chúng ta đã chứng kiến một sự phát triển mạnh mẽ của các trang mạng xã hội và các trang thương mại điện tử. Mua sắm trực tuyến đã trở nên phổ biến hơn bao giờ hết, nhờ đó con người có thể chia sẻ thông tin, diễn đạt quan điểm và thái độ của mình về sản phẩm và dịch vụ. Thông qua việc chia sẻ thông tin này, người dùng có thể tra cứu và tham khảo trước khi quyết định mua một sản phẩm hoặc sử dụng một dịch vụ. Các nhà cung cấp dịch vụ cũng có thể sử dụng thông tin này để có cái nhìn tổng quan về thái độ của khách hàng đối với sản phẩm và dịch vụ của mình, từ đó thực hiện các cải tiến nhằm đáp ứng nhu cầu của người dùng và thu hút và giữ chân khách hàng. Đặc biệt, khi một công ty tung ra sản phẩm mới, việc thu nhận phản hồi trở nên vô cùng quan trọng.

Việc thu thập phản hồi từ khách hàng là một phương pháp hiệu quả để các công ty hiểu rõ điểm mạnh và điểm yếu của sản phẩm và dịch vụ của mình. Đồng thời, điều này giúp họ nhanh chóng thấu hiểu tâm trạng và nhu cầu của khách hàng, từ đó mang đến cho họ những sản phẩm và dịch vụ hoàn hảo nhất. Đó chính là lý do tại sao nhóm chúng tôi đã chọn nghiên cứu và khai thác đề tài "Ứng dụng học sâu trong phân tích quan điểm".

Trên thực tế, việc áp dụng phương pháp học sâu trong phân tích quan điểm giúp chúng tôi khai thác hiệu quả các dữ liệu phản hồi từ khách hàng. Điều này cho phép chúng tôi xác định và hiểu rõ hơn về ý kiến, nhận định và cảm xúc của khách hàng đối với sản phẩm và dịch vụ. Kết quả của nghiên cứu này sẽ cung cấp cho công ty những thông tin quan trọng để cải thiện chất lượng sản phẩm và dịch vụ, từ đó nâng cao sự hài lòng của khách hàng và tăng tính cạnh tranh trên thị trường.

Chúng tôi tin rằng việc nghiên cứu và áp dụng học sâu cho bài toán phân tích quan điểm sẽ mang lại những kết quả giá trị và đóng góp tích cực cho lĩnh vực quản lý và phát triển sản phẩm và dịch vụ. Chúng tôi hy vọng rằng đề tài này sẽ góp phần thúc đẩy sự phát triển bền vững của doanh nghiệp và mang lại lợi ích cho khách hàng cũng như cộng đồng.

## 2. MỤC TIÊU VÀ NHIỆM VỤ NGHIÊN CỨU

Mục tiêu của đề tài là nghiên cứu và áp dụng các cơ sở lý thuyết của bài toán phân tích cảm xúc và quan điểm (sentiment analysis). Chúng tôi tập trung vào việc tìm hiểu các khía cạnh sau:

Tìm hiểu cơ sở lý thuyết: Chúng tôi nghiên cứu và tìm hiểu về các khái niệm, phương pháp và công cụ liên quan đến bài toán phân tích cảm xúc và quan điểm. Điều này bao gồm việc tìm hiểu về các khái niệm về cảm xúc, quan điểm, các loại dữ liệu ngôn ngữ như văn bản, và các phương pháp phân tích cảm xúc.

Tìm hiểu thuật toán neuron cơ bản và các thuật toán phân tích cảm xúc: Chúng tôi tìm hiểu về các thuật toán neuron cơ bản như mạng neural sâu (deep neural networks) và các thuật toán sử dụng trong phân tích cảm xúc như mạng neural tích chập (CNN) và mạng neural dài hạn ngắn (LSTM). Chúng tôi nghiên cứu cách áp dụng và điều chỉnh các thuật toán này để xử lí bài toán phân tích cảm xúc và quan điểm.

Xây dựng mô hình phân tích quan điểm: Chúng tôi tiến hành xây dựng một mô hình phân tích quan điểm dựa trên khía cạnh cụ thể, tập trung vào việc phân loại quan điểm văn bản dựa trên các khía cạnh cụ thể. Mục tiêu là hiểu và phân tích quan điểm của người dùng dựa trên các dữ liệu như bình luận, đánh giá hay phản hồi, và xác định các khía cạnh cụ thể trong quan điểm đó. Mô hình phân tích quan điểm dựa trên khía cạnh này sẽ cung cấp cái nhìn chi tiết hơn về những yếu tố cụ thể trong quan điểm của người dùng và giúp các doanh nghiệp hiểu rõ hơn về những mặt tích cực và tiêu cực của sản phẩm hoặc dịch vụ mà họ cung cấp. Điều này có thể hỗ trợ quyết định kinh doanh và cải thiện chất lượng sản phẩm và dịch vụ theo hướng phù hợp với nhu cầu và mong muốn của khách hàng.

## 3. CÁCH TIẾP CẬN VÀ PHƯƠNG PHÁP NGHIÊN CỨU

## *• Đối tượng nghiên cứu*

Trong nghiên cứu này, chúng tôi tập trung vào việc phân tích các bình luận và đánh giá từ khách hàng. Mục tiêu là xây dựng một mô hình phân tích quan điểm dựa trên các khía cạnh của các bình luận này.

## *• Phạm vi nghiên cứu*

Phạm vi nghiên cứu bao gồm ứng dụng mô hình PhoBert dựa trên thuật toán Transformer cùng với các thuật toán CNN, LSTM và phương pháp nhúng từ (word embedding) để xây dựng mô hình phân tích quan điểm.

## *• Phương pháp nghiên cứu*

Trong khuôn khổ khóa luận tốt nghiệp, để tiếp cận đề tài và phương pháp nghiên cứu, chúng tôi sẽ thực hiện các giai đoạn sau:

Thu thập tài liệu và phân tích chọn lọc thông tin: Tìm hiểu và nghiên cứu các tài liệu liên quan về xử lý ngôn ngữ tự nhiên, phân tích quan điểm và các phương pháp sử dụng mô hình PhoBert, CNN, LSTM và phương pháp nhúng từ.

Nghiên cứu về Deep Learning, một lĩnh vực của Machine Learning: Tập trung vào việc xây dựng mô hình mạng neural sâu để giải quyết các vấn đề phức tạp trong xử lý ngôn ngữ tự nhiên. Trong nghiên cứu này, chúng ta sẽ tìm hiểu về hai thuật toán quan trọng trong Deep Learning là Convolutional Neural Network (CNN) và Long Short-Term Memory (LSTM), cùng với phương pháp nhúng từ (word embedding).

Chuẩn bị dữ liệu: Chúng tôi sẽ thu thập và tiền xử lý dữ liệu từ tập dữ liệu VLSP 2018 (Vietnamese Language Understanding and Evaluation). Tập dữ liệu này đã được tổ chức và phát triển bởi Viện Công nghệ thông tin - Viện Hàn lâm Khoa học và Công nghệ Việt Nam (IOIT - VAST) và các đối tác liên quan. Quá trình tiền xử lý dữ liệu có thể bao gồm loại bỏ các ký tự đặc biệt, chuyển đổi văn bản thành dạng số và chia tập dữ liệu thành tập huấn luyện và tập kiểm tra. Bằng việc sử dụng tập dữ liệu này, chúng tôi hy vọng sẽ đạt được kết quả tốt hơn trong việc huấn luyện mô hình cho nghiên cứu của mình.

Xây dựng mô hình: Sử dụng mô hình PhoBert dựa trên thuật toán Transformer và kết hợp với các thuật toán CNN, LSTM và phương pháp nhúng từ, chúng tôi sẽ xây dựng một mô hình phân tích quan điểm. Mô hình này sẽ nhận đầu vào là các bình luận và đưa ra dự đoán về quan điểm của chúng dựa trên các khía cạnh được xác định trước. Quá trình xây dựng mô hình bao gồm định nghĩa kiến trúc mạng neural, lựa chọn các hàm kích hoạt, tối ưu hóa tham số và huấn luyện mô hình trên tập dữ liệu huấn luyện.

Đánh giá mô hình: Sử dụng tập dữ liệu kiểm tra để đánh giá hiệu suất của mô hình. Các độ đo phổ biến như độ chính xác (accuracy), độ phủ (recall) và độ chính xác trung bình (F1-score) có thể được sử dụng để đo lường độ chính xác của mô hình trong việc phân loại quan điểm.

Giải thích ứng dụng: Mô hình phân tích quan điểm dựa trên khía cạnh có thể được áp dụng trong nhiều lĩnh vực và ứng dụng khác nhau. Ví dụ, nó có thể được sử dụng để phân tích quan điểm của khách hàng về sản phẩm hoặc dịch vụ trong lĩnh vực thương mại điện tử. Các công ty có thể sử dụng mô hình này để hiểu và đáp ứng nhu cầu của khách hàng, cải thiện chất lượng sản phẩm và dịch vụ, và tăng cường tương tác và hài lòng của khách hàng.

Như vậy, chúng tôi tập trung vào nghiên cứu và ứng dụng các phương pháp Deep Learning như thuật toán CNN, LSTM và cách nhúng từ (embedding) vào việc xây dựng mô hình phân tích quan điểm dựa trên các khía cạnh trong bình luận và đánh giá từ khách hàng.

## 4. KẾT QUẢ DỰ KIẾN ĐẠT ĐƯỢC

Chúng tôi đã tiếp thu kiến thức lý thuyết về Deep Learning và hiểu rõ cách hoạt động của thuật toán CNN, LSTM cũng như phương pháp nhúng từ (embedding). Đồng thời, chúng tôi đã xây dựng và huấn luyện thành công các mô hình mạng CNN, LSTM và nhúng từ embedding để phân tích cảm xúc và quan điểm.

Mô hình chúng tôi đã xây dựng cho phép phân tích quan điểm theo ba mức độ positive, neutral và negative. Qua quá trình huấn luyện, chúng tôi đã đạt được sự chính xác và độ tin cậy cao trong việc phân loại các quan điểm khác nhau.

Với việc xây dựng mô hình này, chúng tôi mong muốn đóng góp vào việc phân tích quan điểm của người dùng và hiểu rõ hơn về cảm xúc và ý kiến của họ. Mô hình của chúng tôi có thể được ứng dụng trong nhiều lĩnh vực, từ đánh giá sản phẩm, dịch vụ cho đến phân tích ý kiến trên mạng xã hội và các nền tảng trực tuyến khác.

Việc áp dụng mô hình phân tích quan điểm này sẽ giúp doanh nghiệp và tổ chức hiểu rõ hơn về ý kiến của khách hàng, từ đó đưa ra các chiến lược và quyết định kinh doanh phù hợp. Ngoài ra, người dùng cũng có thể tận dụng mô hình này để tra cứu, đánh giá và so sánh các sản phẩm và dịch vụ trước khi đưa ra quyết định mua hàng.

# Phần 2: NỘI DUNG

# Chương 1: CƠ SỞ LÝ THUYẾT

## 1.1. TỔNG QUAN VỀ PHÂN TÍCH CẢM XÚC VÀ QUAN ĐIỂM (SENTIMENT ANALYSIS)

### 1.1.1. Khái niệm phân tích quan điểm

Phân tích cảm xúc và quan điểm, hay Sentiment Analysis, là một lĩnh vực trong xử lý ngôn ngữ tự nhiên và trí tuệ nhân tạo, nghiên cứu và áp dụng các phương pháp để nhận biết, phân loại và đánh giá các trạng thái cảm xúc và ý kiến trong văn bản. Nó sử dụng các kỹ thuật từ ngôn ngữ học máy tính, trích xuất thông tin, và phân tích dữ liệu để hiểu và xác định những thông điệp chủ quan và trạng thái cảm xúc trong văn bản một cách có hệ thống.

Phân tích cảm xúc được áp dụng rộng rãi trong nhiều lĩnh vực. Ví dụ, trong lĩnh vực tiếp thị, phân tích cảm xúc giúp các doanh nghiệp hiểu rõ ý kiến và đánh giá của khách hàng về sản phẩm hoặc dịch vụ của họ. Điều này có thể giúp họ điều chỉnh chiến lược kinh doanh, tăng cường tương tác với khách hàng và cải thiện trải nghiệm mua hàng.

### 1.1.2. Lợi ích của phân tích cảm xúc và quan điểm

* *Cung cấp thông tin chuyên sâu khách quan*

Bằng cách sử dụng các công cụ phân tích cảm xúc dựa trên trí tuệ nhân tạo (AI), các doanh nghiệp có thể tránh được thành kiến ​​cá nhân thường liên quan đến người đánh giá là con người. Do đó, các công ty nhận được kết quả nhất quán và khách quan khi phân tích ý kiến ​​của khách hàng. Ví dụ, xem xét các câu sau: *“Tôi ngạc nhiên trước tốc độ của bộ xử lý, nhưng thất vọng vì nó nóng lên quá nhanh.”* Các nhà tiếp thị có thể bỏ qua những phần tiêu cực của các đánh giá để ủng hộ hiệu suất của bộ xử lý. Tuy nhiên, các công cụ phân tích tình cảm chính xác sẽ sắp xếp và phân loại văn bản để chọn ra tình cảm một cách khách quan.

* *Xây dựng sản phẩm và dịch vụ tốt hơn*

Hệ thống phân tích tình cảm giúp các công ty cải thiện sản phẩm và dịch vụ của họ dựa trên phản hồi thực tế và cụ thể của khách hàng. Công nghệ AI xác định các đối tượng hoặc tình huống trong thế giới thực (được gọi là thực thể) mà khách hàng liên tưởng đến những cảm xúc tiêu cực. Từ ví dụ trên, kỹ sư sản phẩm tập trung vào việc cải thiện khả năng quản lý nhiệt của bộ xử lý vì phần mềm phân tích văn bản liên kết từ thất vọng (tiêu cực) với bộ xử lý (thực thể) và sưởi ấm (thực thể).

* *Kết quả theo thời gian thực*

Ngày nay, các doanh nghiệp phải phản ứng nhanh chóng với các cuộc khủng hoảng tiềm ẩn hoặc xu hướng thị trường. Các nhà tiếp thị dựa vào phần mềm phân tích tình cảm để hiểu theo thời gian thực khách hàng cảm thấy thế nào về thương hiệu, sản phẩm và dịch vụ của công ty họ và hành động ngay lập tức dựa trên những phát hiện đó. Họ có thể định cấu hình phần mềm để gửi cảnh báo khi phát hiện thấy tình cảm tiêu cực bằng cách sử dụng các từ khóa cụ thể.

### 1.1.3. Các cấp độ phân tích cảm xúc và quan điểm

Cấp độ phân tích cảm xúc và quan điểm được chia thành nhiều cấp độ khác nhau, từ cấp độ từ vựng cho đến cấp độ khía cạnh. Mỗi cấp độ mang lại thông tin cảm xúc và quan điểm chi tiết, tạo nên một bức tranh toàn diện về ý kiến và cảm nhận của người dùng. Dưới đây là một mô tả chi tiết về các cấp độ phân tích cảm xúc và quan điểm:

***Cấp độ từ vựng (Lexical level):*** Cấp độ này tập trung vào phân tích cảm xúc của từng từ hoặc cụm từ trong văn bản. Mục tiêu là xác định cảm xúc được liên kết với từng đơn vị ngôn ngữ. Để đạt được điều này, có thể sử dụng các từ điển cảm xúc hoặc phân tích ngữ cảnh để xác định cảm xúc của từ. Ví dụ, từ "tuyệt vời" có thể được xem là tích cực trong khi từ "tệ" có thể được xem là tiêu cực.

***Cấp độ câu (Sentence level):*** Ở cấp độ này, chúng ta tập trung vào phân tích cảm xúc của từng câu trong văn bản. Mục tiêu là xác định cảm xúc tổng thể của mỗi câu, ví dụ như xác định liệu câu đó mang tính chất tích cực, trung lập hay tiêu cực. Để đạt được điều này, có thể sử dụng các phương pháp như phân loại máy học, mạng neural hồi quy (RNN), hoặc mô hình học sâu (deep learning) khác. Một ví dụ có thể là: "Sản phẩm này rất tốt" có tính chất tích cực, trong khi "Tôi không hài lòng với dịch vụ khách hàng" có tính chất tiêu cực.

***Cấp độ văn bản (Document level):*** Tại cấp độ này, chúng ta tập trung vào phân tích cảm xúc tổng thể của văn bản. Mục tiêu là xác định cảm xúc chung của văn bản, ví dụ như xác định liệu văn bản mang tính chất tích cực, trung lập hay tiêu cực. Để đạt được điều này, có thể sử dụng các mô hình phân loại như mạng neural, SVM (Support Vector Machines), hoặc các phương pháp học máy khác. Ví dụ, một bài đánh giá sản phẩm có thể được xác định là tích cực nếu cả văn bản thể hiện sự hài lòng với sản phẩm.

***Cấp độ khía cạnh (Aspect level):*** Ở cấp độ này, chúng ta tập trung vào phân tích cảm xúc đối với từng khía cạnh cụ thể trong văn bản. Mục tiêu là xác định cảm xúc liên quan đến các khía cạnh như chất lượng sản phẩm, dịch vụ khách hàng, giá cả, v.v. Để đạt được điều này, chúng ta cần phân đoạn câu, xác định các khía cạnh và áp dụng các phương pháp phân tích cảm xúc phù hợp. Ví dụ, trong một bài đánh giá về một nhà hàng, chúng ta có thể phân tích cảm xúc liên quan đến chất lượng thức ăn, không gian và phục vụ riêng biệt cho mỗi khía cạnh.

Các cấp độ phân tích cảm xúc và quan điểm này cho phép chúng ta hiểu rõ hơn về ý kiến và cảm nhận của người dùng, giúp tăng cường khả năng phân tích và tổng hợp thông tin từ các nguồn dữ liệu văn bản khác nhau. Từ đó, chúng ta có thể áp dụng những thông tin này vào các lĩnh vực như đánh giá sản phẩm, nghiên cứu thị trường, phản hồi khách hàng và định hướng chính sách để đáp ứng nhu cầu và mong muốn của người dùng một cách tốt nhất.

### 1.1.4. Phân tích quan điểm trên cấp độ khía cạnh

Phân tích quan điểm dựa trên khía cạnh (Aspect-based Sentiment Analysis) là một phương pháp trong lĩnh vực xử lý ngôn ngữ tự nhiên và phân tích cảm xúc, tập trung vào việc xác định cảm xúc và quan điểm của người dùng đối với từng khía cạnh cụ thể trong một văn bản. Khía cạnh là một thành phần, đặc điểm hoặc khía cạnh cụ thể của đối tượng hoặc sự kiện mà người dùng đang diễn đạt ý kiến về. Ví dụ, trong một bài đánh giá về một sản phẩm điện thoại di động, các khía cạnh có thể bao gồm hiệu năng, màn hình, pin, camera, hệ điều hành, v.v.

Phân tích quan điểm dựa trên khía cạnh có nhiều ứng dụng quan trọng trong lĩnh vực đánh giá sản phẩm, nghiên cứu thị trường, phản hồi khách hàng và quản lý chất lượng. Nó giúp các doanh nghiệp và tổ chức hiểu được ý kiến, phản hồi và nhu cầu của khách hàng từng khía cạnh riêng biệt, từ đó cải thiện sản phẩm, dịch vụ và chiến lược kinh doanh một cách tốt hơn. Ngoài ra, phân tích quan điểm dựa trên khía cạnh còn hỗ trợ trong việc theo dõi trực tuyến, phân tích mạng xã hội và phản ứng nhanh với ý kiến của người dùng trong thời gian thực.

Phân tích quan điểm dựa trên khía cạnh có các bước chính sau:

***Phân đoạn câu:*** Đầu tiên, văn bản đầu vào được phân đoạn thành các câu để tách riêng các ý kiến và cảm nhận riêng biệt về từng khía cạnh.

*Ví dụ: Văn bản đầu vào "Sản phẩm này có màn hình rộng và sắc nét, nhưng pin yếu và camera không đáng khen ngợi."*

*Các câu sau được phân đoạn:*

*"Sản phẩm này có màn hình rộng và sắc nét."*

*"Pin yếu."*

*"Camera không đáng khen ngợi."*

***Xác định khía cạnh:*** Tiếp theo, chúng ta xác định các khía cạnh được đề cập trong câu. Điều này có thể được thực hiện bằng cách sử dụng từ điển khía cạnh hoặc phương pháp học máy để phân loại từ hoặc cụm từ vào các khía cạnh tương ứng.

*Tiếp tục ví dụ trên các khía cạnh được xác định:*

*Câu 1: Màn hình*

*Câu 2: Pin*

*Câu 3: Camera*

***Phân tích cảm xúc:*** Sau khi xác định khía cạnh, chúng ta phân tích cảm xúc hoặc quan điểm liên quan đến từng khía cạnh đó. Điều này có thể đạt được bằng cách sử dụng phân loại cảm xúc, hồi quy hoặc phương pháp học máy khác để xác định tính tích cực, tiêu cực hoặc trung lập của quan điểm.

*Sau khi xác định khía cạnh, chúng ta phân tích cảm xúc hoặc quan điểm liên quan đến từng khía cạnh.*

*Màn hình: tích cực (positive)*

*Pin: tiêu cực (negative)*

*Camera: tiêu cực (negative)*

***Tổng hợp kết quả:*** Cuối cùng, chúng ta tổng hợp các kết quả phân tích cảm xúc từ các khía cạnh khác nhau để đưa ra tổng thể cảm nhận và quan điểm của người dùng đối với đối tượng hoặc sự kiện được xem xét.

*Tổng hợp kết quả của ví dụ trên: Sản phẩm này được đánh giá tích cực về màn hình, nhưng tiêu cực về pin và camera.*

### 1.1.5. Những trường hợp sử dụng phân tích cảm xúc và quan điểm

Phân tích quan điểm dựa trên khía cạnh là một công cụ mạnh mẽ để hiểu và đánh giá ý kiến và cảm nhận của con người trong nhiều lĩnh vực khác nhau. Dưới đây là một số trường hợp thực tế mà phân tích quan điểm dựa trên khía cạnh có thể được áp dụng:

***Đánh giá sản phẩm và dịch vụ:*** Trong thế giới kinh doanh, việc hiểu quan điểm và cảm nhận của khách hàng về sản phẩm và dịch vụ là vô cùng quan trọng. Phân tích quan điểm dựa trên khía cạnh giúp xác định các khía cạnh cụ thể như chất lượng, tính năng, giá trị và dịch vụ khách hàng. Điều này giúp các doanh nghiệp điều chỉnh chiến lược kinh doanh và cải thiện chất lượng để đáp ứng tốt hơn nhu cầu và mong đợi của khách hàng.

*Ví dụ: Một công ty sản xuất điện thoại di động muốn biết ý kiến của người dùng về camera của sản phẩm mới. Phân tích quan điểm dựa trên khía cạnh sẽ giúp xác định các khía cạnh như độ phân giải, chất lượng hình ảnh và tính năng chụp ảnh để đánh giá ý kiến của người dùng về camera của điện thoại.*

***Đo lường hiệu quả chiến dịch tiếp thị:*** Trong lĩnh vực tiếp thị, việc đánh giá hiệu quả của chiến dịch quảng cáo và tiếp thị là quan trọng để định hướng và tối ưu hóa chiến lược tiếp thị. Phân tích quan điểm dựa trên khía cạnh giúp phân tích ý kiến và cảm nhận của khách hàng về các yếu tố quan trọng như thông điệp quảng cáo, hình ảnh, màu sắc, văn bản quảng cáo, để đo lường hiệu quả của chiến dịch và điều chỉnh nội dung và hình thức tiếp thị để tăng cường tương tác và độ hấp dẫn.

*Ví dụ: Một công ty muốn đánh giá hiệu quả chiến dịch quảng cáo trên mạng xã hội. Phân tích quan điểm dựa trên khía cạnh sẽ giúp xác định ý kiến của người dùng về thông điệp, hình ảnh và hiệu quả của quảng cáo để đánh giá tương tác và hiệu quả của chiến dịch.*

***Đo lường hài lòng khách hàng:*** Việc hiểu mức độ hài lòng của khách hàng là yếu tố quan trọng trong việc đánh giá chất lượng dịch vụ và xác định các điểm mạnh và yếu. Phân tích quan điểm dựa trên khía cạnh giúp xác định các khía cạnh quan trọng như sự phục vụ, chất lượng sản phẩm, thời gian phục vụ, và đánh giá ý kiến và cảm nhận của khách hàng về mỗi khía cạnh đó.

*Ví dụ: Một nhà hàng muốn đo lường mức độ hài lòng của khách hàng về dịch vụ. Phân tích quan điểm dựa trên khía cạnh sẽ giúp xác định ý kiến của khách hàng về chất lượng thức ăn, phục vụ, không gian và giá cả để đánh giá mức độ hài lòng của khách hàng.*

### 1.1.6. Thách thức của bài toán phân tích cảm xúc và quan điểm

Mặc dù đã có sự tiến bộ đáng kể trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP), việc làm cho máy móc hiểu ngôn ngữ của con người vẫn là một thách thức lớn. Máy móc có thể hiểu sai các sắc thái ẩn đằng sau giao tiếp của con người, chẳng hạn như những điều được nêu dưới đây.

Đa nghĩa: Ngôn ngữ tự nhiên có tính đa nghĩa, và một từ hoặc cụm từ có thể có nhiều ý nghĩa khác nhau tùy thuộc vào ngữ cảnh. Điều này làm tăng khó khăn trong việc xác định ý kiến và cảm xúc chính xác liên quan đến một khía cạnh cụ thể.

*Ví dụ: "Camera của sản phẩm này không tốt" - câu này có thể đề cập đến chất lượng hình ảnh của camera hoặc tính năng chụp ảnh của camera. Việc xác định ý kiến chính xác liên quan đến khía cạnh cụ thể trong trường hợp này có thể khó khăn.*

Ngôn ngữ không chuẩn: Trong các nguồn dữ liệu thực tế, người dùng thường viết theo cách không chuẩn, có thể chứa lỗi chính tả, viết tắt, ngôn ngữ lõi, hay ngôn ngữ hình thức. Điều này làm tăng thách thức trong việc xử lý và hiểu đúng ý kiến và cảm nhận của người dùng.

*Ví dụ: "Sản phẩm này rất kk, tôi rất hài lòng" - việc giải mã từ viết tắt "kk" thành "khá khá" để hiểu rằng người dùng hài lòng với sản phẩm có thể là một thách thức.*

Châm biếm: Châm biếm là một hình thức diễn đạt ý kiến trái ngược với ý nghĩa đen của từng từ hoặc câu. Máy tính thường gặp khó khăn trong việc nhận diện châm biếm và hiểu ý nghĩa thực sự của câu.

*Ví dụ: Câu sau "Dịch vụ của công ty này thật tuyệt vời!" được nói với ý châm biếm, nhưng máy tính có thể hiểu nhầm là một ý kiến tích cực.*

Phủ định: Sử dụng các từ phủ định để biểu thị ý nghĩa trái ngược là một thách thức khó khăn trong phân tích cảm xúc. Máy tính cần phải phân biệt giữa sự phủ định và khẳng định trong câu để hiểu ý nghĩa chính xác.

*Ví dụ, câu sau: "Dù không quá tốt, nhưng sản phẩm vẫn đáng giá để thử." chứa một phủ định và có ý nghĩa tích cực, nhưng máy tính có thể hiểu sai và xem đó là một ý kiến tiêu cực.*

Sự đa nghĩa của từ: Một từ có thể có nhiều ý nghĩa khác nhau tùy thuộc vào ngữ cảnh. Điều này tạo ra khó khăn trong việc xác định cảm xúc hoặc quan điểm chính xác.

*Ví dụ, từ "tốt" có thể được sử dụng để diễn tả một ý kiến tích cực ("sản phẩm tốt") hoặc một ý kiến trung lập ("tốt nhưng không đặc biệt").*

Tổng quan, việc phân tích quan điểm dựa trên khía cạnh đòi hỏi máy tính có khả năng hiểu được ngữ cảnh, nhận biết châm biếm, phủ định và sự đa nghĩa của từ. Để vượt qua những thách thức này, phải sử dụng các phương pháp và thuật toán phân tích ngôn ngữ tự nhiên phức tạp và có khả năng xử lý các trường hợp đặc biệt để đưa ra kết quả chính xác và đáng tin cậy.

## 1.2. TỔNG QUAN VỀ XỬ LÍ NGÔN NGỮ TỰ NHIÊN (NLP)

### 1.2.1. Xử lí ngôn ngữ tự nhiên

Xử lí ngôn ngữ tự nhiên (Natural Language Processing - NLP) là một lĩnh vực của trí tuệ nhân tạo (Artificial Intelligence - AI) và khoa học máy tính, tập trung vào việc nghiên cứu và phát triển các phương pháp và công nghệ để máy tính có khả năng hiểu, xử lí và tương tác với ngôn ngữ con người tự nhiên.

Mục tiêu chính của NLP là giúp máy tính hiểu và làm việc với ngôn ngữ con người theo cách tương tự như con người. Điều này bao gồm khả năng hiểu ý nghĩa và cấu trúc của văn bản, phân tích và trích xuất thông tin quan trọng, dịch thuật tự động, tạo ra văn bản tự nhiên và phân tích cảm xúc trong văn bản.

### 1.2.2. Tầm quan trọng của xử lí ngôn ngữ tự nhiên

Xử lý ngôn ngữ tự nhiên giúp máy tính nói chuyện với mọi người bằng ngôn ngữ riêng của họ và mở rộng quy mô các nhiệm vụ khác liên quan đến ngôn ngữ.

Các doanh nghiệp sử dụng một lượng lớn dữ liệu không có cấu trúc, nhiều văn bản và cần một cách để xử lý nó một cách hiệu quả. Rất nhiều thông tin được tạo trực tuyến và được lưu trữ trong cơ sở dữ liệu là ngôn ngữ tự nhiên của con người, và cho đến gần đây, các doanh nghiệp không thể phân tích hiệu quả dữ liệu này. Đây là nơi xử lý ngôn ngữ tự nhiên rất hữu ích.

Lợi thế của xử lý ngôn ngữ tự nhiên có thể được nhìn thấy khi xem xét hai tuyên bố sau: “Cloud computing insurance should be part of every service-level agreement,” and, “A good SLA ensures an easier night's sleep -- even in the cloud.” Nếu người dùng dựa vào xử lý ngôn ngữ tự nhiên để tìm kiếm, chương trình sẽ nhận ra rằng cloud computing là một thực thể, cloud đó là một dạng viết tắt của cloud computing và SLA là từ viết tắt của ngành cho service-level agreement.

Đây là những loại yếu tố mơ hồ thường xuyên xuất hiện trong ngôn ngữ của con người và các thuật toán học máy trước đây rất tệ trong việc diễn giải. Giờ đây, với những cải tiến trong phương pháp học sâu và học máy, các thuật toán có thể diễn giải chúng một cách hiệu quả. Những cải tiến này mở rộng chiều rộng và chiều sâu của dữ liệu có thể được phân tích.

### 1.2.3. Một vài ứng dụng của xử lí ngôn ngữ tự nhiên

Một số chức năng chính mà các thuật toán xử lý ngôn ngữ tự nhiên thực hiện là:

- Phân loại văn bản: NLP có thể được sử dụng để phân loại văn bản vào các danh mục khác nhau. Điều này hữu ích trong việc phân tích cảm xúc, giúp xác định tính tích cực hoặc tiêu cực của một văn bản. Ví dụ, thuật toán có thể xác định tỷ lệ đề cập tích cực và tiêu cực đối với một thương hiệu trong một số lượng văn bản nhất định. Nó cũng có thể giúp dự đoán ý định, đoán những hành động mà người nói hoặc người viết có thể thực hiện dựa trên nội dung văn bản.

- Chatbot và trợ lý ảo: NLP được sử dụng để phát triển chatbot và trợ lý ảo thông minh, giúp tương tác tự nhiên với người dùng. Chúng có khả năng hiểu và trả lời câu hỏi, cung cấp thông tin, hỗ trợ giao dịch và giải đáp thắc mắc. Ứng dụng này rất hữu ích trong lĩnh vực dịch vụ khách hàng, hỗ trợ kỹ thuật và bán hàng trực tuyến.

- Dịch máy và đa ngôn ngữ: NLP đã đóng vai trò quan trọng trong việc phát triển công nghệ dịch máy. Nhờ NLP, người dùng có thể dễ dàng dịch các văn bản, tài liệu từ một ngôn ngữ sang ngôn ngữ khác. Công nghệ dịch máy ngày càng hoàn thiện, giúp kết nối và giao tiếp giữa các quốc gia và ngôn ngữ trở nên dễ dàng hơn.

- Phân tích cảm xúc và quan điểm: NLP đóng vai trò quan trọng trong việc phân tích và đánh giá cảm xúc và quan điểm từ văn bản. Các công nghệ phân tích cảm xúc cho phép nhận biết tính tích cực, tiêu cực hoặc trung lập của một đoạn văn, đánh giá sản phẩm hoặc xác định quan điểm của người dùng trong các bình luận, đánh giá trực tuyến và phản hồi khách hàng.

- Tóm tắt và trích xuất thông tin: NLP được sử dụng để tự động tóm tắt và trích xuất thông tin từ các nguồn văn bản dài. Điều này giúp tiết kiệm thời gian và công sức trong việc nắm bắt thông tin quan trọng từ các bài báo, tài liệu, email, v.v. Bằng cách tổng hợp các điểm chính và loại bỏ những thông tin không quan trọng, NLP giúp người dùng tiếp cận thông tin một cách nhanh chóng và hiệu quả.

- Xử lí ngôn ngữ tự nhiên trong y học: NLP cũng có ứng dụng quan trọng trong lĩnh vực y học. Nó có thể giúp phân tích và trích xuất thông tin từ các bài báo y học, hồ sơ bệnh án, thông tin về thuốc, v.v. NLP cung cấp các công cụ giúp xử lí thông tin y học một cách tự động và hiệu quả, từ việc đưa ra chẩn đoán đến theo dõi tình trạng bệnh nhân.

## 1.3. WORD EMBEDDING

Word embedding là một phương pháp trong xử lý ngôn ngữ tự nhiên (NLP) để biểu diễn từ và câu dưới dạng các vectơ số học trong không gian đa chiều. Mục tiêu của word embedding là ánh xạ từng từ trong ngôn ngữ tự nhiên thành các vectơ có tính chất ngữ nghĩa, nơi các từ có ý nghĩa tương đồng được biểu diễn gần nhau.

Trong word embedding, mỗi từ được biểu diễn bằng một vectơ số học. Các vectơ này có thể có kích thước cố định và được xác định dựa trên các thuật toán học máy, chẳng hạn như Word2Vec, GloVe hoặc FastText. Các thuật toán này học cách biểu diễn từng từ dựa trên ngữ cảnh xuất hiện của chúng trong văn bản.

Giả sử chúng ta có một tập dữ liệu gồm các câu văn đơn giản:

"Tôi thích ăn pizza."

Sử dụng thuật toán Word2Vec, chúng ta có thể biểu diễn mỗi từ trong câu dưới dạng các vectơ số học. Ví dụ, sau khi huấn luyện, chúng ta có thể có các vectơ biểu diễn từ như sau:

"Tôi": [0.2, 0.5, 0.8]

"thích": [0.7, 0.3, 0.1]

"ăn": [0.4, 0.6, 0.9]

"pizza": [0.9, 0.2, 0.6]

Các vectơ này có thể biểu diễn ý nghĩa và mối quan hệ giữa các từ. Ví dụ, từ "pizza" có vectơ gần với từ "ăn", và cũng có một phần tương đồng với từ "thích". Tương tự, từ "chơi" và "bóng đá" có mối quan hệ gần nhau hơn so với các từ khác.

Việc sử dụng word embedding cho phân loại văn bản có thể giúp máy tính hiểu được ý nghĩa của từng từ và mối quan hệ giữa chúng. Ví dụ, nếu chúng ta có một câu mới là "Tôi thích ăn sushi", máy tính có thể dễ dàng nhận ra rằng từ "sushi" tương đồng với "pizza" và "ăn", và do đó có thể kết luận rằng câu này mang tính tích cực.

Có một số phương pháp phổ biến được sử dụng trong việc nhúng từ hoặc câu trong xử lý ngôn ngữ tự nhiên, bao gồm TF-IDF, Word2Vec, GloVe, FastText và BERT.

### 1.3.1. Term Frequency – Inverse Document Frequency (TF-IDF)

TF-IDF (Term Frequency – Inverse Document Frequency) là một kĩ thuật sử dụng trong khai phá dữ liệu văn bản. Trọng số này được sử dụng để đánh giá tầm quan trọng của một từ trong một văn bản. Giá trị cao thể hiện độ quan trọng cao và nó phụ thuộc vào số lần từ xuất hiện trong văn bản nhưng bù lại bởi tần suất của từ đó trong tập dữ liệu. Một vài biến thể của tf-idf thường được sử dụng trong các hệ thống tìm kiếm như một công cụ chính để đánh giá và sắp xếp văn bản dựa vào truy vấn của người dùng. Tf-idf cũng được sử dụng để lọc những từ stopwords trong các bài toán như tóm tắt văn bản và phân loại văn bản.

TF: Term Frequency (Tần suất xuất hiện của từ) là số lần từ xuất hiện trong văn bản. Vì các văn bản có thể có độ dài ngắn khác nhau nên một số từ có thể xuất hiện nhiều lần trong một văn bản dài hơn là một văn bản ngắn. Như vậy, term frequency thường được chia cho độ dài văn bản (tổng số từ trong một văn bản).

=

Trong đó:

: Tần suất xuất hiện của từ t trong văn bản d

: Số lần xuất hiện của từ t trong văn bản d

: Số lần xuất hiện của từ có số lần xuất hiện nhiều nhất trong văn bản d

IDF: Inverse Document Frequency (Nghịch đảo tần suất của văn bản), giúp đánh giá tầm quan trọng của một từ. Khi tính toán TF, tất cả các từ được coi như có độ quan trọng bằng nhau. Nhưng một số từ như “is”, “of” và “that” thường xuất hiện rất nhiều lần nhưng độ quan trọng là không cao. Như thế chúng ta cần giảm độ quan trọng của những từ này xuống.

*= =*

*Trong đó:*

: giá trị idf của từ t trong tập văn bản

: Tổng số văn bản trong tập D

: thể hiện số văn bản trong tập D có chứa từ t.

Cơ số logarit trong công thức này không thay đổi giá trị idf của từ mà chỉ thu hẹp khoảng giá trị của từ đó. Vì thay đổi cơ số sẽ dẫn đến việc giá trị của các từ thay đổi bởi một số nhất định và tỷ lệ giữa các trọng lượng với nhau sẽ không thay đổi. (nói cách khác, thay đổi cơ số sẽ không ảnh hưởng đến tỷ lệ giữa các giá trị IDF). Việc sử dụng logarit nhằm giúp giá trị tf-idf của một từ nhỏ hơn, do chúng ta có công thức tính tf-idf của một từ trong một văn bản là tích của tf và idf của từ đó.

Cụ thể, chúng ta có công thức tính tf-idf hoàn chỉnh như sau: **tf-idf (t, d, D) = tf (t, d) x idf (t, D)**

Khi đó: Những từ có giá trị TF-IDF cao là những từ xuất hiện nhiều trong văn bản này, và xuất hiện ít trong các văn bản khác. Việc này giúp lọc ra những từ phổ biến và giữ lại những từ có giá trị cao (từ khoá của văn bản đó)

### 1.3.2. Word2Vec

Word2vec là một kỹ thuật xử lý ngôn ngữ tự nhiên. Thuật toán word2vec sử dụng một mô hình mạng thần kinh để học các liên kết từ (sự liên quan của từ) từ một kho ngữ liệu văn bản có dung lượng lớn. Sau khi được huấn luyện, mô hình có thể phát hiện các từ đồng nghĩa hoặc gợi ý các từ bổ sung cho một phần của câu. Với cái tên nói lên tất cả, word2vec thể hiện cho mỗi từ riêng biệt với một danh sách cụ thể của các số được gọi là vectơ. Các vectơ được lựa chọn cẩn thận sao cho một hàm toán học đơn giản sẽ (độ tương tự cosin giữa các vectơ) cho biết mức độ của độ tương tự ngữ nghĩa giữa các từ được biểu diễn bằng các vectơ đó.

Kiến trúc tổng quát của word2vec:

Mô hình word2vec sử dụng một kiến trúc neural network đơn giản với một hidden layer duy nhất. Mục tiêu của mô hình là để tìm hiểu ma trận trọng số cho các từ. Các trọng số này chính là các vector từ.

Mô hình này gồm 3 lớp chính:

+ Input layer: vector từng từ đầu vào được biểu diễn dạng one-hot

+ Hidden layer: lớp nhúng

+ Output layer: là một vector phân bố xác suất

Có 2 phương pháp phổ biến: Continuous Bag-of-Words (CBOW) và Skip-gram

* ***Skip-gram:***

Giả định rằng một từ có thể được sử dụng để tạo ra các từ xung quanh nó trong một chuỗi văn bản. Lấy chuỗi văn bản “the”, “man”, “loves”, “his”, “son” làm ví dụ. Hãy chọn “loves” làm từ trung tâm và đặt kích thước cửa sổ ngữ cảnh thành 2. Với từ trung tâm là “loves”, mô hình bỏ qua gam xem xét xác suất có điều kiện để tạo ra các từ ngữ cảnh:

“the”, “man”, “his”, và “son”, cách từ trung tâm không quá 2 từ:

P (“the”, “man”, “his”, “son” | “loves”)

Giả sử rằng các từ ngữ cảnh được tạo độc lập với từ trung tâm (nghĩa là độc lập có điều kiện). Trong trường hợp này, xác suất có điều kiện ở trên có thể được viết lại là:

P (“the” | “loves”). P (“man” | “loves”). P (“his” | “loves”). P (“son” | “loves”)

Trong mô hình skip-gram, mỗi từ được biểu diễn bằng vector d chiều để tính xác

suất có điều kiện. Giả sử chỉ số của một từ trong từ điển là i, vector của từ được

biểu diễn là khi từ này là từ đích trung tâm và là khi từ này là một

từ ngữ cảnh. Gọi c và o lần lượt là chỉ số của từ đích trung tâm và từ ngữ cảnh

trong từ điển. Có thể thu được xác suất có điều kiện sinh ra từ ngữ cảnh cho

một từ đích trung tâm cho trước bằng phép toán softmax trên tích vô hướng của

vector:

*() =*

Trong đó, tập chỉ số trong bộ từ vựng là  *=* . Giả sử trong một

chuỗi văn bản có độ dài T, từ tại bước thời gian t được ký hiệu là . Giả sử rằng

các từ ngữ cảnh được sinh độc lập với từ trung tâm cho trước. Khi kích thước cửa

sổ ngữ cảnh là m, hàm hợp lý (likelihood) của mô hình skip-gram là xác suất kết

hợp sinh ra tất cả các từ ngữ cảnh với bất kỳ từ trung tâm cho trước nào.

trong đó bất kỳ bước thời gian nào nhỏ hơn hoặc lớn hơn có thể được bỏ qua.

Traning:

Các tham số của mô hình là vectơ từ trung tâm và vectơ từ ngữ cảnh cho mỗi từ trong từ vựng. Trong đào tạo, chúng tôi tìm hiểu các tham số mô hình bằng cách tối đa hóa hàm khả năng (nghĩa là ước tính khả năng tối đa). Điều này tương đương với việc cực tiểu hóa hàm mất mát sau:

Khi sử dụng stochastic gradient descent để giảm thiểu tổn thất, trong mỗi lần lặp lại, chúng ta có thể lấy mẫu ngẫu nhiên một chuỗi con ngắn hơn để tính toán độ dốc (ngẫu nhiên) cho chuỗi con này để cập nhật các tham số mô hình. Để tính toán độ dốc (ngẫu nhiên) này, chúng ta cần thu được độ dốc của xác suất có điều kiện nhật ký đối với vectơ từ trung tâm và vectơ từ ngữ cảnh. Nói chung, log xác suất có điều kiện liên quan đến bất kỳ cặp từ trung tâm nào và từ ngữ cảnh là:

*() = –*

Thông qua sự khác biệt, chúng ta có thể thu được độ dốc của nó đối với vectơ từ trung tâm như:

*=-*

*=-*

*=*

* ***Continuous Bag-of-Words (CBOW)***

Sự khác biệt chính so với mô hình Skip-gram là mô hình CBOW giả định rằng một từ trung tâm được tạo ra dựa trên các từ ngữ cảnh xung quanh của nó trong chuỗi văn bản. Ví dụ: trong cùng một chuỗi văn bản “the”, “man”, “loves”, “his” và “son”, với “loves” là từ trung tâm và kích thước cửa sổ ngữ cảnh là 2, mô hình CBOW xem xét xác suất có điều kiện để tạo ra từ trung tâm “loves” dựa trên các từ ngữ cảnh “the”, “man”, “his” và “son”

P (“loves” | “the”, “man”, “his”, “son”)

Vì có quá nhiều từ ngữ cảnh trong mô hình CBOW, ta sẽ lấy trung bình các vector từ của chúng và sau đó sử dụng phương pháp tương tự như trong mô hình skipgram để tính xác suất có điều kiện. Giả sử  và  là vector từ ngữ cảnh và vector từ đích trung tâm của từ có chỉ số i trong từ điển (lưu ý rằng các ký hiệu này ngược với các ký hiệu trong mô hình skip-gram). Gọi c là chỉ số của từ đích trung tâm wc, và O1 ,... ,O2m là chỉ số các từ ngữ cảnh ,. . . , trong từ điển.

Do đó, xác suất có điều kiện sinh ra từ đích trung tâm dựa vào các từ ngữ cảnh cho

trước là

*P() =*

Để cho ngắn gọn, hãy để *W0 = và 0 =*. Sau đó đơn giản hóa phương trình thành

*P() =*

Cho một chuỗi văn bản có độ dài T, trong đó từ tại bước thời gian t được ký hiệu là . Đối với kích thước cửa sổ ngữ cảnh m, hàm khả năng của mô hình CBOW là xác suất tạo ra tất cả các từ trung tâm cho các từ ngữ cảnh của chúng:

*Traning:*

Đào tạo mô hình CBOW gần giống như mô hình skip gram. Uớc lượng hợp lý cực đại của mô hình CBOW tương đương với việc cực tiểu hóa hàm mất mát:

-

*Lưu ý rằng:*

*log P() = – log ()*

Thông qua sự khác biệt, chúng ta có thể thu được gradient của nó đối với bất kỳ vectơ từ ngữ cảnh nào v0i với *i=*như:

*= ( = (*

### 1.3.3. Glove

GloVe (Global Vectors for Word Representation) là một phương pháp nhúng từ được giới thiệu bởi Jeffrey Pennington và đồng nghiệp tại Stanford vào năm 2014. GloVe sử dụng một phương pháp dựa trên xác suất thống kê để tạo ra các vectơ nhúng từ.

GloVe xây dựng trực tiếp trên ma trận tần số từ/cặp từ trong ngữ liệu huấn luyện. Ma trận này cho biết tần số xuất hiện của các từ đơn lẻ và cặp từ trong ngữ liệu. Từ ma trận này, GloVe tạo ra các vectơ nhúng từ bằng cách xác định các đặc trưng ngữ nghĩa của từ dựa trên mối quan hệ tương quan giữa các từ và cặp từ trong ngữ liệu.

Phương pháp này đạt được điểm mạnh bằng cách xem xét tần số xuất hiện của các từ và cặp từ, và từ đó xác định mối quan hệ ngữ nghĩa giữa chúng. Kết quả là chúng ta có các vectơ nhúng từ biểu diễn các từ dựa trên ngữ nghĩa và mối quan hệ giữa chúng.

Giả sử chúng ta có một tập dữ liệu huấn luyện gồm các câu sau đây:

"I love to eat pizza"

"Pizza is my favorite food"

"I enjoy eating pizza with friends"

Để sử dụng phương pháp GloVe, chúng ta cần tiền xử lý dữ liệu và tính toán ma trận tần số từ/cặp từ. Đầu tiên, chúng ta chia các câu thành các từ đơn lẻ (token) bằng cách tách chúng dựa trên khoảng trắng, ví dụ:

["I", "love", "to", "eat", "pizza"]

["Pizza", "is", "my", "favorite", "food"]

["I", "enjoy", "eating", "pizza", "with", "friends"]

Tiếp theo, chúng ta tính toán tần số xuất hiện của từ đơn lẻ và cặp từ trong tập huấn luyện. Bằng cách duyệt qua các câu và đếm tần số xuất hiện, chúng ta có thể tính được rằng từ "pizza" xuất hiện 3 lần và cặp từ "pizza" và "eat" xuất hiện 1 lần.

Sau khi tính toán ma trận tần số từ/cặp từ, chúng ta sử dụng phương pháp GloVe để huấn luyện mô hình nhúng từ. Trong quá trình huấn luyện, GloVe tìm cách tối ưu các vectơ nhúng từ sao cho việc tích vô hướng giữa các vectơ này phản ánh mối quan hệ ngữ nghĩa giữa các từ và cặp từ.

Để đạt được mục tiêu này, GloVe sử dụng hàm mục tiêu đặc biệt và kỹ thuật tối ưu hóa để điều chỉnh các tham số của mô hình và tối thiểu hóa hàm mục tiêu. Quá trình huấn luyện cố gắng điều chỉnh các vectơ nhúng từ sao cho việc tích vô hướng của chúng phản ánh mối quan hệ ngữ nghĩa giữa các từ và cặp từ dựa trên ma trận tần số từ/cặp từ.

Khi quá trình huấn luyện hoàn tất, chúng ta sẽ có các vectơ nhúng từ cho mỗi từ trong tập huấn luyện. Ví dụ, chúng ta có thể có vectơ nhúng từ cho từ "pizza" là [0.2, -0.1, 0.4, ..., 0.8].

Các vectơ nhúng từ này biểu diễn các từ dựa trên ngữ nghĩa và mối quan hệ giữa chúng. Với các vectơ nhúng từ này, chúng ta có thể tính toán độ tương đồng giữa các từ dựa trên khoảng cách cosine. Ví dụ, nếu tính toán khoảng cách cosine giữa vectơ nhúng từ của "pizza" và "eat", chúng ta có thể thấy rằng hai từ này có ý nghĩa gần nhau.

### 1.3.4. FastText

Phương pháp FastText là một phương pháp nhúng từ (word embedding) được phát triển bởi Tomas Mikolov và đồng nghiệp tại Facebook AI Research. Nó mở rộng phương pháp Word2Vec bằng cách sử dụng các n-gram của từ để biểu diễn từ và xử lý các từ không xuất hiện trong tập huấn luyện.

Giả sử chúng ta có tập dữ liệu huấn luyện gồm các câu sau đây:

"I love to eat pizza"

"Pizza is my favorite food"

"I enjoy eating pizza with friends"

Để sử dụng phương pháp FastText, chúng ta cần tiền xử lý dữ liệu và tạo ra các n-gram của từ. Một n-gram là một đơn vị con liên tiếp của từ. Ví dụ, với từ "apple", ta có thể tạo ra các n-gram là "<ap", "app", "ppl", "ple", "le>" (các dấu chevron chỉ đại diện cho đầu hoặc cuối từ).

Tiếp theo, chúng ta huấn luyện mô hình FastText sử dụng các n-gram và tần số xuất hiện của chúng. Trong quá trình huấn luyện, FastText tạo ra các vectơ đặc trưng cho các n-gram và tổ hợp chúng để tạo ra vectơ nhúng từ.

Ví dụ, trong câu "I love to eat pizza", chúng ta có các n-gram như "<I", "I ", " lo", "lov", "ove", "ve ", "e t", " to", "to ", "o e", " ea", "eat", "at ", "t p", " pi", "piz", "izz", "zza", "za>". FastText tạo ra các vectơ đặc trưng cho các n-gram này và tổ hợp chúng để tạo ra vectơ nhúng từ cho từng từ trong câu.

Sau quá trình huấn luyện, chúng ta có các vectơ nhúng từ cho từng từ trong tập huấn luyện. Ví dụ, chúng ta có thể có vectơ nhúng từ cho từ "pizza" là [0.2, -0.1, 0.4, ..., 0.8].

Phương pháp FastText có ưu điểm là có khả năng xử lý các từ không xuất hiện trong tập huấn luyện bằng cách sử dụng các n-gram của chúng. Ví dụ, nếu chúng ta có từ "apple" trong tập kiểm tra nhưng không xuất hiện trong tập huấn luyện, FastText vẫn có thể tạo ra vectơ nhúng từ cho từ "apple" bằng cách sử dụng các n-gram "<ap", "app", "ppl", "ple", "le>".

### 1.3.5. BERT

Phương pháp nhúng từ BERT (Bidirectional Encoder Representations from Transformers) là một trong những phương pháp nhúng từ tiên tiến nhất và hiệu quả nhất trong xử lý ngôn ngữ tự nhiên (NLP). Được giới thiệu bởi Jacob Devlin và đồng nghiệp tại Google vào năm 2018, BERT sử dụng mô hình Transformer và quá trình huấn luyện đồng thời hai tác vụ để tạo ra các vectơ nhúng từ biểu diễn ngữ nghĩa phong phú và hiểu ngữ cảnh từ hai chiều.

Cách thức hoạt động của phương pháp nhúng từ với BERT như sau:

Tiền xử lý dữ liệu: Đầu tiên, dữ liệu văn bản được tiền xử lý để chuẩn bị đầu vào phù hợp cho BERT. Các từ được chia thành các đơn vị nhỏ hơn gọi là "subword". BERT sử dụng một bộ từ vựng subword (ví dụ: WordPiece) để mã hóa từng từ thành các mã số. Các mã số này sau đó được chuyển thành vectơ one-hot để tạo ra đầu vào cho mô hình BERT.

Kiến trúc mạng Transformer: BERT sử dụng kiến trúc mạng Transformer, gồm nhiều lớp Encoder Transformer. Mỗi lớp Encoder Transformer bao gồm một lớp Self-Attention và mạng Perceptron nhiều lớp (Feed-Forward Neural Network). Kiến trúc này cho phép BERT hiểu mối quan hệ ngữ cảnh giữa các từ trong câu.

Quá trình huấn luyện đồng thời hai tác vụ:

Dự đoán từ bị che giấu (Masked Language Modeling - MLM): BERT sẽ che giấu ngẫu nhiên một số từ trong câu huấn luyện và cố gắng dự đoán chúng dựa trên ngữ cảnh xung quanh. Ví dụ, trong câu "I love to eat [MASK]", BERT sẽ cố gắng dự đoán từ bị che giấu "pizza" dựa trên các từ "I", "love", "to", "eat".

Dự đoán quan hệ giữa các cặp câu (Next Sentence Prediction - NSP): BERT sẽ đưa vào đầu vào hai câu liên tiếp và dự đoán xem chúng có liên quan logic hay không. Ví dụ, cho hai câu "I love to eat pizza" và "Pizza is my favorite food", BERT sẽ cố gắng dự đoán rằng chúng có liên quan với nhau.

Quá trình huấn luyện và điều chỉnh tham số: BERT được huấn luyện thông qua quá trình tối ưu hóa và điều chỉnh các tham số của mạng Transformer. Mục tiêu là tối thiểu hóa hàm mất mát (loss function) trong quá trình dự đoán từ bị che giấu và quan hệ giữa các cặp câu. Quá trình huấn luyện này sẽ tạo ra các vectơ nhúng từ biểu diễn ngữ nghĩa phong phú và hiểu ngữ cảnh từ hai chiều.

Sau quá trình huấn luyện, BERT cung cấp các vectơ nhúng từ phong phú biểu diễn từng từ trong câu. Ví dụ, chúng ta có thể có vectơ nhúng từ cho từ "pizza" là [0.2, -0.1, 0.4, ..., 0.8]. Các vectơ nhúng từ này bao gồm thông tin về ngữ nghĩa của từ, vị trí của từ trong câu và mối quan hệ giữa từ và ngữ cảnh xung quanh.

## 1.4. TỔNG QUAN VỀ DEEP LEARNING

### 1.4.1. Deep Learning

Deep Learning (Học sâu) là một phương pháp học máy (Machine Learning) trong lĩnh vực trí tuệ nhân tạo (Artificial Intelligence) dựa trên các mô hình mạng neural nhân tạo (Artificial Neural Networks) sâu và có nhiều lớp ẩn. Deep Learning nhằm mô phỏng cách mà hệ thống thần kinh của con người hoạt động, nơi thông tin được truyền qua các lớp neuron và các liên kết giữa chúng.

Với khả năng học từ dữ liệu lớn và đa dạng, Deep Learning đã mang lại những thành tựu đáng kể trong nhiều lĩnh vực như thị giác máy tính (Computer Vision), xử lý ngôn ngữ tự nhiên (Natural Language Processing), nhận dạng giọng nói (Speech Recognition) và nhiều ứng dụng khác. Các mô hình Deep Learning phổ biến bao gồm mạng neural tích chập (Convolutional Neural Networks - CNN) trong thị giác máy tính và bộ nhớ ngắn hạn kéo dài (Long Short-Term Memory - LSTM) trong xử lý dữ liệu chuỗi thời gian.

Deep Learning đã đạt được nhiều kết quả ấn tượng, từ việc nhận dạng hình ảnh, dịch máy tự động, gợi ý nội dung, cho đến tự lái xe. Nhờ khả năng học tập và biểu diễn mô hình phức tạp, Deep Learning đóng vai trò quan trọng trong việc thúc đẩy tiến bộ của trí tuệ nhân tạo và mang lại nhiều tiềm năng ứng dụng trong thực tế.

### 1.4.2. Một vài ứng dụng của Deep Learning

- Ứng dụng trong lái xe tự động: Tự động hóa là từ khóa quan trọng trong cả cuộc Cách mạng Công nghiệp 4.0 và xe tự lái. Ngày nay, người ta đã ứng dụng thành công công nghệ Deep Learning để xây dựng hệ thống tự lái. Do đó, mạng neural này sẽ thực hiện các phép tính: phát hiện vật thể xung quanh xe, xác định khoảng cách an toàn, hiểu rõ luật giao thông của từng quốc gia, … đưa ra quyết định tối ưu và nhanh chóng.

- Phân tích cảm xúc người dùng: Nhiều doanh nghiệp có thể sử dụng học sâu để phân tích ngôn ngữ tự nhiên và cách diễn đạt mà người dùng để lại trên mạng xã hội khi tương tác, bình luận và thích hình ảnh. Từ đó, dựa vào nội dung, họ biết được cảm xúc và thái độ của người dùng để xây dựng cách thức phù hợp tiếp cận khách hàng.

- Mô phỏng, hình ảnh và nhận dạng Khuôn mặt: Học sâu cũng được sử dụng để tạo ra công nghệ nhận dạng hình ảnh và khuôn mặt. Ví dụ như tính năng tự động gắn thẻ người dùng trên Facebook, hay tính năng gắn thẻ ảnh giúp dễ dàng quản lý trong Google Photos.

- Ứng dụng phát triển trợ lý ảo: Các ứng dụng deep learning phổ biến nhất hiện nay là trợ lý ảo từ Alexa cho đến Siri, Google Assistant. Trợ lý ảo sử dụng học sâu để tìm hiểu thêm về đối tượng của người dùng để có thể hiểu các lệnh bằng cách đánh giá ngôn ngữ tự nhiên của con người được sử dụng để thực thi chúng.

- Chức năng dịch văn bản tự động: Đây là một ứng dụng học sâu để nhận dạng ngôn ngữ và chuyển đổi hình ảnh.

- Hệ thống gợi ý trên các nền tảng: các nền tảng lớn hiện nay như Facebook, Amazon, Netflix, ... tất cả đều có một hệ thống khuyến nghị rất mạnh giúp tăng đáng kể sự tương tác của người dùng. Bằng cách dựa trên các dữ liệu người dùng phát sinh ra khi dùng để gợi ý thêm những sản phẩm họ sẽ thích (trên các nền tảng mua sắm), những bộ phim họ sẽ muốn xem (vd. như trên Netflix), gợi ý các bài quảng cáo/được tài trợ (trên Facebook) hay các khóa học người học quan tâm (trên các nền tảng học online).

### 1.4.3. Cách thức hoạt động của Deep Learning

Một mạng neural bao gồm nhiều lớp (layer) khác nhau, số lượng layer càng nhiều thì mạng sẽ càng “sâu”. Trong mỗi layer là các nút mạng (node) và được liên kết với những lớp liền kề khác. Mỗi kết nối giữa các node sẽ có một trọng số tương ứng, trọng số càng cao thì ảnh hưởng của kết nối này đến mạng neural càng lớn.

Mỗi neural sẽ có một hàm kích hoạt, về cơ bản thì có nhiệm vụ “chuẩn hoá” đầu ra từ neural này. Dữ liệu được người dùng đưa vào mạng neural sẽ đi qua tất cả layer và trả về kết quả ở layer cuối cùng, gọi là output layer.

Trong quá trình huấn luyện mô hình mạng neural, các trọng số sẽ được thay đổi và nhiệm vụ của mô hình là tìm ra bộ giá trị của trọng số sao cho phán đoán là tốt nhất.

Các hệ thống Deep Learning yêu cầu phần cứng phải rất mạnh để có thể xử lý được lượng dữ liệu lớn và thực hiện các phép tính phức tạp. Nhiều mô hình Deep Learning có thể mất nhiều tuần, thậm chí nhiều tháng để triển khai trên những phần cứng tiên tiến nhất hiện nay.

### 1.4.4. Convolutional Neural Network (CNNs)

Convolutional Neural Network (CNNs – Mạng neural tích chập) là một kiểu kiến trúc mạng neural nhân tạo được sử dụng rộng rãi trong lĩnh vực xử lý ảnh và thị giác máy tính. CNNs được thiết kế dựa trên cách thức hoạt động của thị giác con người để nhận diện và phân loại hình ảnh.

Mạng CNN là một tập hợp các lớp tích chập chồng lên nhau và sử dụng các hàm nonlinear activation 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.

Quá trình huấn luyện mạng CNNs thường bao gồm việc cung cấp dữ liệu huấn luyện gồm các ảnh đầu vào và nhãn đúng tương ứng. Mạng CNNs sẽ học cách tự động tinh chỉnh các trọng số của nó thông qua quá trình lan truyền ngược và tối ưu hóa, nhằm tối đa hóa độ chính xác của việc phân loại ảnh.

Với cấu trúc và khả năng học tập sâu, CNNs đã đạt được nhiều thành công đáng kể trong nhiều nhiệm vụ xử lý ảnh, bao gồm nhận diện khuôn mặt, phân loại đối tượng, nhận dạng chữ viết tay, và nhiều ứng dụng khác liên quan đến thị giác máy tính.

Mạng CNN gồm những lớp cơ bản sau:

- Convolutional layer: Đây chính là lớp đóng vai trò mấu chốt của CNN, khi layer này đảm nhiệm việc thực hiện mọi tính toán. Stride, padding, filter map, feature map là những yếu tố quan trọng nhất của convolutional layer.

- Relu layer: Còn có tên gọi khác là activation function, đây là một hàm được kích hoạt trong neural network. Nó có tác dụng mô phỏng các neuron có tỷ lệ truyền xung qua axon. Trong activation function chúng còn có hàm nghĩa là: Relu, Tanh, Sigmoid, Maxout, Leaky, … Relu layer được ứng dụng phổ biến trong việc huấn luyện neural do sở hữu nhiều ưu điểm tiên tiến.

- Pooling layer: Khi nhận phải đầu vào quá lớn, các lớp pooling layer sẽ được xếp giữa những lớp Convolutional layer nhằm mục đích giảm parameter. Pooling layer được chia thành 2 loại phổ biến là max pooling và average.

- Fully connected layer: Khi 2 lớp convolutional layer và pooling layer nhận được ảnh truyền, lớp này sẽ có nhiệm vụ xuất kết quả. Khi ta nhận được kết quả là model đọc được thông tin ảnh, ta cần phải tạo sự liên kết để cho ra nhiều output hơn. Đây chính là lúc các lập trình viên sử dụng fully connected layer. Hơn nữa, nếu fully connected layer có dữ liệu về hình ảnh thì chúng sẽ chuyển thành mục chưa được phân chia chất lượng.

Diagram

Description automatically generated

#### Hình 1.1: Kiến trúc CNN phân loại câu [19]

Quan sát Hình 1.1, ta có:

Đầu vào là một câu gồm 7 từ (kể cả ký tự!) “I like this movie very much!”. Dùng phương pháp word embedding để chuyển đổi các từ thành vector có số chiều d = 5. Khi đó, ta có một ma trận đầu vào là 7×5.

Ở layer tiếp theo, sử dụng các filter với region size khác nhau gồm 2, 3, và 4. Gấp đôi số lượng từng dạng filter này lên, ta có tổng cộng 6 filter. Tiếp đến, thực hiện nhân tích chập tương ứng với từng filter và cho qua activate function (softmax hay tanh) để thu được các vector đầu ra. Sau đó, cho các vector này đi qua filter max-pooling (average-pooling hay các phép toán tương tự khác), để chuẩn hóa đầu ra các vector sao cho chúng có số chiều là như nhau.

Ở layer cuối cùng, tùy theo bài toán mà ta sẽ giới hạn số chiều của vector phân lớp này thành 2 hay nhiều chiều tương ứng.

### 1.4.5. Recurrent neural network (RNNs)

Recurrent Neural Network (RNNs – Mạng neural hồi quy) là một loại mạng neural nhân tạo chủ yếu được sử dụng trong nhận dạng giọng nói và xử lý ngôn ngữ tự nhiên (NLP). RNN được sử dụng trong học tập sâu và trong việc phát triển các mô hình bắt chước hoạt động của các tế bào thần kinh trong não người.

A picture containing text, clock

Description automatically generated

#### Hình 1.2: Kiến trúc mạng RNN [20]

Quan sát Hình 1.2, ta có:

Lớp đầu vào ‘x’ nhận đầu vào cho mạng neural và xử lý nó và chuyển nó vào lớp giữa.

Lớp giữa ‘h’ có thể bao gồm nhiều lớp ẩn, mỗi lớp có chức năng kích hoạt và trọng số và thành kiến riêng.

RNN sẽ chuẩn hóa các chức năng kích hoạt và trọng số và độ lệch khác nhau để mỗi lớp ẩn có các tham số giống nhau. Sau đó, thay vì tạo nhiều lớp ẩn, nó sẽ tạo một lớp và lặp lại nhiều lần theo yêu cầu.

Mạng RNN là một kiến ​​trúc đặc biệt có tính đến thông tin thời gian. Trạng thái ẩn của RNN tại thời điểm t lấy thông tin từ cả đầu vào tại thời điểm t và kích hoạt từ các đơn vị ẩn tại thời điểm t-1, để tính toán đầu ra cho thời điểm t. Điều này có thể được nhìn thấy trong hình trên. Điều này mang lại cho bộ nhớ RNN hoặc khả năng ghi nhớ các đầu vào trước đó và đầu ra của chúng.

Điều này cực kỳ quan trọng đối với xử lý ngôn ngữ tự nhiên, vì trong NLP dữ liệu đầu vào không có kích thước cố định và từ tiếp theo phụ thuộc nhiều vào từ trước đó.. Kích thước của mỗi câu khác nhau và đầu ra của mỗi câu cũng khác nhau. Do đó, khả năng lấy kích thước đầu vào thay đổi và tính toán đầu ra có kích thước thay đổi là rất có lợi và RNN có khả năng này. RNN cũng có bộ nhớ và có thể ghi nhớ thông tin theo ngữ cảnh.

### 1.4.6. Long short-term memory (LSTMs)

Long short-term memory (LSTMs) là một phiên bản mở rộng của mạng Recurrent Neural Network (RNN), nó được thiết kế để giải quyết các bài toán về phụ thuộc xa (long-term dependencies). RNN là mạng neural có chứa vòng lặp. Mạng này có khả năng lưu trữ thông tin, thông tin được truyền từ lớp này sang lớp khác. Đầu ra của lớp ẩn phụ thuộc vào thông tin của các lớp tại mọi thời điểm. RNN đã được sử dụng phổ biến trong xử lý ngôn ngữ tự nhiên hay các bài toán có dữ liệu tuần tự. Tuy nhiên, do kiến trúc của RNN khá đơn giản nên khả năng lien kết các lớp có khoảng cách xa là không tốt. Nó cơ bản không có khả năng ghi nhớ thông tin từ các dữ liệu có khoảng cách xa, và do đó, những phần tử đầu tiên trong chuỗi đầu vào thường không có nhiều ảnh hưởng đến kết quả dự đoán phần tử cho chuỗi đầu ra các bước sau. Nguyên nhân của việc này là do RNN chịu ảnh hưởng bởi việc đạo hàm bị thấp dần trong quá trình học – biến mất đạo hàm (vanishing gradient). Mạng LSTM được thiết kế để khắc phục vấn đề này. Cơ chế hoạt động của LSTM là chỉ ghi nhớ những thông tin liên quan, quan trọng cho việc dự đoán, còn các thông tin khác sẽ được bỏ đi.

Ở state thứ t của mô hình LSTM:

* Output: , , ta gọi c là cell state, h là hidden state.
* Input:  , , Trong đó   là input ở state thứ t của model. , là output của layer trước. **h** đóng vai trò khá giống như **s**ở RNN, trong khi **c**là điểm mới của LSTM.

Diagram

Description automatically generated

#### Hình 1.3: Kiến trúc của một node trong mạng LSTM [20]

: forget gate quyết định xem cần lấy bao nhiêu từ cell state trước

: input gate sẽ quyết định lấy bao nhiêu từ input của state và hidden layer của layer trước.

: output gate quyết định xem cần lấy bao nhiêu từ cell state để trở thành output của hidden state

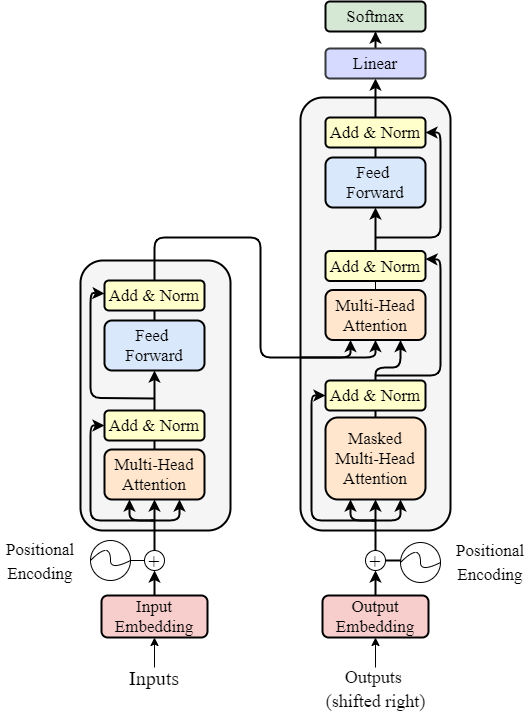
*= tanh*

*=∗​+∗*

*=​∗tanh ()*

### 1.4.7. Transformer

Transformer là một mô hình học sâu được thiết kế để giải quyết nhiều bài toán trong xử lý ngôn ngữ tự nhiên và tiếng nói, bao gồm dịch tự động, sinh ngôn ngữ, phân loại, nhận dạng thực thể, nhận dạng tiếng nói và chuyển văn bản thành tiếng nói. Điều đặc biệt về Transformer so với mô hình RNN là nó không yêu cầu xử lý các phần tử trong chuỗi một cách tuần tự. Trong khi RNNs phải xử lý từng phần tử theo thứ tự từ đầu đến cuối, Transformer có thể xử lý toàn bộ câu ngôn ngữ tự nhiên cùng một lúc.

Điều này cho phép Transformer tận dụng tối đa khả năng tính toán song song của GPU và giảm thời gian xử lý đáng kể. Thay vì phải chờ đợi kết quả từ các phần tử trước đó trong chuỗi, Transformer có thể đồng thời xem xét và ánh xạ mối quan hệ giữa tất cả các từ trong câu. Điều này giúp mô hình hiểu được mối quan hệ toàn diện trong câu và tạo ra các biểu diễn ngữ nghĩa phong phú hơn.

#### Hình 1.4. Kiến trúc mô hình Transformer [21]

Transformer hoạt động dựa trên cơ chế attention, trong đó mỗi từ trong câu được so sánh với tất cả các từ khác để xác định mức độ quan trọng của từ đó đối với các từ khác. Quá trình attention này cho phép mô hình tập trung vào các từ quan trọng và xây dựng biểu diễn phản ánh mối quan hệ giữa chúng. Đồng thời, Transformer sử dụng các lớp mạng neural đầy đủ để thực hiện biến đổi phi tuyến trên biểu diễn từ sau quá trình attention.

Dưới đây là nguyên lý cách thức hoạt động của mạng Transformer trong xử lý ngôn ngữ tự nhiên:

Đầu vào và biểu diễn từ: Đầu tiên, các từ trong câu được biểu diễn dưới dạng vector. Mô hình sử dụng phương pháp nhúng từ (word embedding) để ánh xạ các từ vào không gian vector. Các vector này chứa thông tin về ý nghĩa và ngữ cảnh của từ trong câu.

Mã hóa (Encoder): Sau đó, các từ được đưa vào lớp Encoder của mạng Transformer. Lớp Encoder gồm nhiều block Transformer, mỗi block bao gồm hai phần quan trọng là self-attention và mạng neural đầy đủ.

Self-attention: Trong phần self-attention, mỗi từ trong câu được so sánh với tất cả các từ khác trong câu để xác định mức độ quan trọng của từ đó đối với các từ khác. Quá trình này tạo ra một ma trận attention cho mỗi từ, cho biết mức độ quan trọng của từ đó đối với các từ khác trong câu. Ma trận attention được tính toán bằng cách sử dụng các hàm tương tự (similarity functions) và các trọng số học tự động.

Mạng neural đầy đủ: Sau quá trình self-attention, một mạng neural đầy đủ (feed-forward neural network) được áp dụng để biến đổi biểu diễn của từ. Mạng neural đầy đủ thực hiện các biến đổi phi tuyến trên biểu diễn từ, tạo ra một biểu diễn mới có khả năng bao gồm thông tin ngữ nghĩa và ngữ cảnh.

Giải mã (Decoder): Sau quá trình mã hóa, biểu diễn ngữ nghĩa của câu được truyền vào lớp Decoder. Lớp Decoder cũng gồm nhiều block Transformer, nhưng có thêm một phần attention khác, gọi là attention tiếp theo, để tập trung vào các từ trong câu đầu vào và biểu diễn đã được mã hóa.

Tổng quan, sự kết hợp giữa cơ chế attention và tính toán song song của Transformer làm cho nó trở thành một mô hình mạnh mẽ trong xử lý ngôn ngữ tự nhiên. Với khả năng xử lý hiệu quả các câu ngôn ngữ tự nhiên và hiểu được các mối quan hệ phức tạp, Transformer đã đạt được nhiều thành tựu đáng chú ý trong lĩnh vực NLP.

### 1.4.8. Mô hình BERT dựa trên kiến trúc Transformer

Mô hình BERT (Bidirectional Encoder Representations from Transformers) là một mô hình học sâu dựa trên kiến trúc Transformer, được sử dụng rộng rãi trong xử lý ngôn ngữ tự nhiên. Mô hình BERT được huấn luyện trên một lượng lớn văn bản từ các nguồn khác nhau để học các biểu diễn ngữ nghĩa của các từ và câu. Điểm đặc biệt của BERT là khả năng hiểu được ngữ cảnh từ cả hai phía của một từ, bằng cách sử dụng một quá trình huấn luyện "mã hóa hai chiều". Thay vì chỉ sử dụng ngữ cảnh từ trước hoặc sau, BERT sử dụng cả hai hướng để nắm bắt thông tin ngôn ngữ.

*Nguyên tắc hoạt động của mô hình BERT*

Input: Mô hình BERT nhận các câu đầu vào và mã hóa chúng thành một chuỗi các token (từ hoặc ký tự). Mỗi token được nhúng thành một vector trước khi được đưa vào mạng neural.

Encoding: Các vector nhúng đầu vào được truyền qua một kiến trúc mạng Transformer. Trong mạng Transformer, các lớp attention đa đầu (multi-head attention) được áp dụng để nắm bắt mối quan hệ giữa các token trong câu.

Output: Kết quả của mô hình BERT là một chuỗi các vector có cùng kích thước với input. Mỗi vector trong chuỗi đại diện cho một token trong câu đầu vào và mang thông tin về ngữ nghĩa của token đó.

Trong quá trình huấn luyện, mô hình BERT đối mặt với một thách thức là hạn chế trong việc hiểu ngữ cảnh của các từ. Để khắc phục nhược điểm này, BERT sử dụng hai chiến lược huấn luyện như sau:

Masked Language Model (MLM): Trước khi đưa vào mô hình BERT, khoảng 15% số từ trong chuỗi được thay thế bằng token [MASK]. Mô hình sau đó cố gắng dự đoán các từ ban đầu của các vị trí bị che, dựa trên ngữ cảnh được cung cấp bởi các từ khác trong chuỗi. Mô hình BERT giữ lại kiến trúc Encoder trong tác vụ MLM. Các vector embedding đầu ra sau khi thực hiện self-attention và feed-forward được ký hiệu là O1, O2, ..., O5. Để tính toán phân phối xác suất cho từ đầu ra, một lớp Fully Connected Layer được thêm vào sau bộ mã hóa Transformer. Vector đầu ra được nhân với ma trận nhúng để chuyển đổi chúng thành các chiều từ vựng. Hàm softmax được áp dụng để tính toán phân phối xác suất. Cuối cùng, các véc-tơ nhúng của các từ tại vị trí [MASK] sẽ là các véc-tơ giảm chiều của các véc-tơ Oi sau khi đi qua lớp Fully Connected.

Next Sentence Prediction (NSP): Mô hình BERT sử dụng cặp câu làm dữ liệu đầu vào và dự đoán liệu câu thứ hai có phải là câu tiếp theo của câu đầu tiên hay không. Trong quá trình huấn luyện, 50% dữ liệu đầu vào là các cặp câu trong đó câu thứ hai thực sự là câu tiếp theo của câu đầu tiên, và 50% còn lại là câu thứ hai được chọn ngẫu nhiên từ tập dữ liệu. Các bước xử lý trong tác vụ Next Sentence Prediction bao gồm đưa toàn bộ câu đầu vào qua mô hình Transformer, chuyển đổi vector đầu ra của token [CLS] về kích thước 2x1 bằng một lớp phân loại, và tính toán xác suất cho việc là câu tiếp theo bằng hàm softmax.

*Tinh chỉnh mô hình BERT*

Sau quá trình huấn luyện, mô hình BERT có thể được tinh chỉnh (fine-tuning) cho các tác vụ cụ thể. Một lớp đầu ra được thêm vào kiến trúc mô hình để điều chỉnh cho tác vụ huấn luyện cụ thể đó.

Tinh chỉnh là quá trình gồm các bước sau:

Bước 1: Nhúng toàn bộ các từ trong cặp câu sử dụng các véc tơ nhúng từ mô hình đã được huấn luyện trước. Các từ nhúng bao gồm cả [CLS] và [SEP] để đánh dấu vị trí bắt đầu của câu hỏi và vị trí phân tách giữa hai câu. Các từ này được dự đoán ở đầu ra để xác định các vị trí bắt đầu/kết thúc của câu đầu ra.

Bước 2: Các véc tơ nhúng sau đó được đưa qua kiến trúc multi-head attention với nhiều khối mã (thông thường là 6, 12 hoặc 24 khối tùy thuộc vào kiến trúc BERT). Kết quả là một véc tơ đầu ra từ bộ mã hóa.

Bước 3: Để dự đoán phân phối xác suất cho từng vị trí từ trong bộ giải mã, ở mỗi bước thời gian chúng ta truyền véc tơ đầu ra từ bộ mã hóa và véc tơ nhúng đầu vào của bộ giải mã để tính toán sự chú ý giữa bộ mã hóa và bộ giải mã (gọi là encoder-decoder attention). Sau đó, chúng ta chiếu qua một lớp tuyến tính và áp dụng hàm softmax để thu được phân phối xác suất cho đầu ra tương ứng ở thời điểm đó.

Bước 4: Trong kết quả đầu ra của transformer, chúng ta giữ phần câu hỏi cố định để khớp với câu hỏi đầu vào. Các vị trí còn lại đại diện cho phần mở rộng Start/End Span tương ứng với câu trả lời tìm thấy từ câu đầu vào.

*Các biến thể của BERT*

Trong họ mô hình BERT, có nhiều biến thể được phát triển để đáp ứng các yêu cầu và nhu cầu cụ thể của các tác vụ xử lý ngôn ngữ tự nhiên. Dưới đây là một số biến thể quan trọng và phổ biến của mô hình BERT:

BERT-base: BERT-base là biến thể cơ bản và đáng chú ý của mô hình BERT. Nó được huấn luyện trên một lượng lớn dữ liệu văn bản và có kích thước lớn, thường có khoảng 110 triệu tham số. BERT-base mang lại khả năng hiểu và biểu diễn ngữ nghĩa tốt cho các từ và câu.

BERT-large: BERT-large là phiên bản mở rộng của BERT-base, với kích thước lớn hơn và số lượng tham số gấp đôi. BERT-large có khả năng biểu diễn ngữ nghĩa và hiểu ngữ cảnh tốt hơn, nhưng đòi hỏi nhiều tài nguyên tính toán hơn để huấn luyện và sử dụng.

BERT-multilingual: BERT-multilingual là biến thể của BERT được huấn luyện trên nhiều ngôn ngữ khác nhau. Điều này cho phép mô hình BERT hỗ trợ đa ngôn ngữ, với khả năng hiểu và biểu diễn ngữ nghĩa cho các từ và câu trong nhiều ngôn ngữ khác nhau.

DistilBERT: DistilBERT là phiên bản nhỏ gọn của BERT, được thu gọn từ BERT-base thông qua quá trình huấn luyện có giám sát. Mặc dù có số lượng tham số ít hơn, DistilBERT vẫn giữ lại khả năng biểu diễn ngữ nghĩa tương đối tốt và có tốc độ nhanh hơn để huấn luyện và sử dụng.

RoBERTa: RoBERTa là biến thể khác của BERT, được huấn luyện trên lượng dữ liệu lớn hơn và với các phương pháp huấn luyện khác nhau. RoBERTa cải thiện hiệu suất của BERT trên một số tác vụ và thường được sử dụng trong các nhiệm vụ đòi hỏi sự hiểu ngữ cảnh chi tiết.

ALBERT: ALBERT (A Lite BERT) là một phiên bản nhẹ của BERT, với lượng tham số giảm đáng kể. ALBERT sử dụng các kỹ thuật huấn luyện mới để tăng tốc quá trình huấn luyện và sử dụng, trong khi vẫn giữ được khả năng biểu diễn ngữ nghĩa và hiểu ngữ cảnh.

ELECTRA: ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) là một biến thể khác của BERT, sử dụng mô hình adversarial training để đạt được hiệu suất cao hơn và tăng tốc quá trình huấn luyện. ELECTRA tập trung vào việc đánh giá xem một từ đã được thay thế đúng hay không, thay vì dự đoán từ ban đầu như trong BERT.

PhoBERT: PhoBERT (Pretrained hetero-lingual BERT) là một biến thể của mô hình BERT được phát triển đặc biệt cho tiếng Việt. "Pho" trong tên PhoBERT đề cập đến phần tiền xử lý ngôn ngữ tiếng Việt bằng cách sử dụng công cụ PhoBERT tokenizer, một tokenizer được tạo ra để phân tách từ trong tiếng Việt. PhoBERT được huấn luyện trên một lượng lớn dữ liệu tiếng Việt từ các nguồn khác nhau để hiểu và biểu diễn ngữ nghĩa của các từ và câu trong tiếng Việt. Nó sử dụng kiến trúc Transformer và áp dụng cả hai chiến lược huấn luyện của BERT là Masked Language Model (MLM) và Next Sentence Prediction (NSP) như đã được mô tả trước đó.

# 

# Chương 2: XÂY DỰNG MÔ HÌNH PHÂN TÍCH QUAN ĐIỂM

## 2.1. BÀI TOÁN

Bài toán phân tích quan điểm dựa trên khía cạnh (Aspect-based Sentiment Analysis) là một dạng mở rộng của bài toán phân tích quan điểm thông thường. Trong bài toán này, không chỉ xác định quan điểm tổng thể mà còn phân tích và đánh giá quan điểm theo từng khía cạnh cụ thể của một đối tượng hoặc sự kiện. Cụ thể, trong một văn bản, các khía cạnh là những yếu tố, tính chất, hay phần tử cụ thể mà người viết hoặc người đánh giá muốn bày tỏ ý kiến.

Để gải quyết bài toán phân tích quan điểm dựa trên khía cạnh, chúng ta cần xử lý dữ liệu văn bản và hiểu được sự tương quan giữa các từ và câu trong văn bản đó. Để làm điều này, chúng ta có thể sử dụng mô hình CNN-LSTM và mô hình PhoBERT.

Mô hình CNN-LSTM kết hợp hai kiến trúc mạng nơ-ron phổ biến: CNN để trích xuất đặc trưng từ văn bản và LSTM để hiểu phụ thuộc dài hạn. Điều này giúp mô hình nhận diện quan điểm dựa trên khía cạnh trong văn bản.

Mô hình PhoBERT là một mô hình ngôn ngữ tự nhiên dựa trên Transformer, đã được huấn luyện trên dữ liệu tiếng Việt. Nó có khả năng hiểu ngữ cảnh và ý nghĩa của từ, giúp phân tích quan điểm dựa trên khía cạnh.

## 2.2. DỮ LIỆU

Dữ liệu lấy từ bộ dữ liệu Vietnamese Language and Speech Processing (VLSP) 2018 về đánh giá nhà hàng. Bộ dữ liệu này cung cấp một nguồn thông tin quý giá để nghiên cứu và phát triển các mô hình phân tích quan điểm trong ngữ cảnh nhà hàng bằng tiếng Việt. Bộ dữ liệu gồm 3 tập: tập train gồm 2961 đánh giá, tập val gồm 1290 đánh gias, tập test gồm 500 đánh giá

Mỗi tập dữ liệu chứa các đánh giá về các nhà hàng được viết bằng tiếng Việt, mỗi đánh giá được gắn nhãn cho nhiều khía cạnh khác nhau của nhà hàng, như chất lượng dịch vụ, món ăn, không gian, giá cả, và nhiều khía cạnh khác.

* *Nhãn đánh giá, quan điểm, cảm nhận (opinion polarity) bao gồm:*

- positive: tức nhận xét tích cực, khen

- negative: tức nhận xét tiêu cực, chê

- neutral: tức nằm giữa tích cực và tiêu cực (trung lập)

* *Nhãn aspect: gồm bộ entity-attribute của thực thể*

Các loại Entity:

- restaurant (nhà hàng): nói đến nhà hàng nói chung, khi không chỉ cụ thể một entity nào như ở dưới, thì có nghĩa là đề cập chung đến restaurant

- ambience (không gian): liên quan đến không gian của nhà hàng, bao gồm việc bố trí, thiết kế và cảm giác chung. Ví dụ: "Nhà hàng có không gian đẹp, có sân vườn rộng, và chỗ cho trẻ em chơi".

- location (vị trí): liên quan vị trí nhà hàng có thuận tiện không, view nhà hàng có đẹp không, có chỗ gửi xe không, …

- food (đồ ăn): liên quan đến các các món ăn tronh nhà hàng

- service (phục vụ): Liên quan đến nhân viên phục vụ và cách họ đối xử với khách hàng.

- drinks (đồ uống): liên quan đến các loại đồ uống trong nhà hàng

Các loại Attribute:

- general (nói chung): thường được sử dụng với khái niệm "nhà hàng" nói chung.

- quality (chất lượng): thường được sử dụng với đồ ăn, đồ uống để đánh giá chất lượng

- price (giá cả): liên quan đến giá của nhà hàng hoặc các mặt hàng như đồ ăn, đồ uống, dịch vụ.

- style\_option (kiểu, tùy chọn): sử dụng để miêu tả cách trình bày, kiểu phục vụ, sự đa dạng trong thực đơn.

- miscellaneous (thuộc tính khác): các thuộc tính khác không thuộc vào nhóm trên.

* *Ta sẽ có một số aspect như sau:*

restaurant#general: nhà hàng nói chung

restaurant#price: giá cả của nhà hàng

restaurant#miscellaneous: thuộc tinh khác của nhà h

food#quality: chất lượng món ăn

food#stype\_option: trình bày món ăn

food#price: giá cả của món ăn

drinks#quality: chất lượng đồ uống

drinks#stype\_option: trình bày đồ uống

drinks#price: giá cả của đô uống

service#general: phục vụ nói chung

embience#general: không gian nói chung

location#general: vị trí nói chung

A picture containing text, font, screenshot

Description automatically generated

#### Hình 2.1: Minh họa về tập dữ liệu

## 2.3. XÂY DỰNG MÔ HÌNH

### 2.3.1. Tiền xử lý dữ liệu

Loại bỏ các ký tự đặc biệt các ký tự kéo dài, xóa các emoji, tách từ, thay thế từ, chuyển chữ hoa sang chữ thường, chuyển về chữ không dấu có thể được thực hiện bằng các bước sau:

Loại bỏ các ký tự đặc biệt: Các ký tự đặc biệt như dấu câu, ký tự đặc biệt, và ký tự đặc biệt trong URL được thay thế bằng khoảng trắng.

Loại bỏ các ký tự kéo dài: Những ký tự liên tiếp có cùng chữ cái được rút gọn lại thành một chữ cái duy nhất. Ví dụ: "hayyy" được chuyển thành "hay".

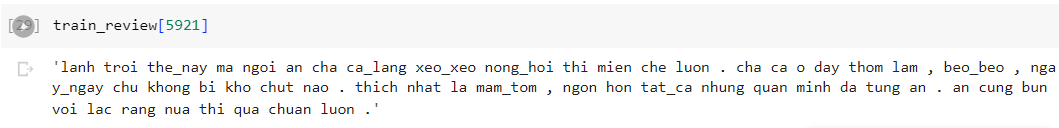
Loại bỏ emoji: Các biểu tượng cảm xúc như hình vui mặt và biểu tượng đặc biệt khác được loại bỏ khỏi văn bản.

Thay thế từ: thay thế các từ trong replace\_list bằng các giá trị tương ứng để chuẩn hóa tiếng Việt.

Tách từ: Văn bản được tách thành các từ riêng lẻ để chuẩn bị cho các bước xử lý tiếp theo.

Chuyển chữ hoa thành chữ thường: Các chữ hoa trong văn bản được chuyển đổi thành chữ thường.

Chuyển về chữ không dấu: Các ký tự có dấu trong văn bản được chuyển thành các ký tự không dấu tương ứng.



#### Hình 3.1: Minh họa các đánh giá sau khi tiền xử lý

### 2.3.2. Vector hóa dữ liệu

*Vector hóa dữ liệu input trong mô hình CNN\_LSTM*

Tokenization: Tiếp theo, câu được chia thành các từ (hoặc token) riêng lẻ. Quá trình này có thể sử dụng các công cụ như tokenizer trong thư viện Natural Language Toolkit (NLTK) hoặc các phương pháp tùy chỉnh.

Chuyển đổi thành mã số: Mỗi từ (hoặc token) trong câu được chuyển đổi thành một mã số tương ứng. Điều này thường được thực hiện bằng cách sử dụng từ điển hoặc bảng ánh xạ, trong đó mỗi từ có một chỉ số số học tương ứng.

Padding và Truncation: Để đảm bảo rằng tất cả các câu có cùng độ dài, ta có thể thực hiện padding (thêm mã số đặc biệt như 0) để các câu có độ dài bằng nhau hoặc truncation (cắt bớt) để các câu có độ dài giới hạn.

Vector hóa: Cuối cùng, các mã số đã được chuyển đổi được biểu diễn dưới dạng các vector nhúng. Sử dụng các phương pháp nhúng từ (word embeddings) như FastText để chuyển đổi các mã số thành các vector nhúng tương ứng.

*Vector hóa dữ liệu input trong mô hình phoBERT*

Tokenization: Dữ liệu văn bản được phân tách thành các từ hoặc ký tự đơn với sự hỗ trợ của tokenizer trong PhoBERT. Tokenizer trong PhoBERT sử dụng quy tắc và kiến thức ngôn ngữ tiếng Việt để đảm bảo sự chính xác trong việc phân tách từ.

Mã hóa thành mã số: Các từ hoặc ký tự được mã hóa thành các mã số tương ứng dựa trên từ điển trong mô hình PhoBERT. Mỗi mã số đại diện cho một từ hoặc ký tự trong từ điển.

Thêm mã số đặc biệt: Trong quá trình mã hóa, các mã số đặc biệt như mã số đại diện cho [CLS] (đánh dấu câu hỏi), [SEP] (phân cách giữa hai câu) và [PAD] (đệm) được thêm vào chuỗi mã số. [CLS] được thêm vào đầu câu, [SEP] được thêm vào giữa hai câu và [PAD] được thêm vào cuối câu để đảm bảo các câu có cùng độ dài.

Chuyển đổi thành vector nhúng: Các mã số sau khi được mã hóa được chuyển đổi thành các vector nhúng tương ứng trong mô hình PhoBERT. Các vector nhúng này đại diện cho mỗi từ hoặc ký tự trong câu.

*Vector hóa dữ liệu output*

Sử dụng phương pháp one-hot, gán các giá trị cảm xúc thành chuỗi vector số như sau:

none = [1, 0, 0, 0]

positive = [0, 1, 0, 0]

negative = [0, 0, 1, 0]

neutral = [0, 0, 0, 1]

### 2.3.3. Xây dựng và huấn luyện mô hình mạng neural xử lý ngôn ngữ tự nhiên

* *Mô hình CNN kết hợp LSTM*

Input Layer: Mô hình nhận đầu vào là một sequence (dãy) các từ, có kích thước tối đa là max\_feature.

Embedding Layer: Đầu vào sequence được biểu diễn dưới dạng các vector embedding. Mô hình sử dụng một lớp Embedding để ánh xạ các từ vào các vector embedding có kích thước embedding\_dim. Trọng số của lớp Embedding được khởi tạo từ ma trận embedding\_matrix và không được huấn luyện lại (trainable=False).

Convolutional Layers: Các lớp Conv1D và MaxPooling1D được sử dụng để trích xuất các đặc trưng từ các vector embedding. Các lớp này giúp mô hình nhận biết các mẫu và cấu trúc trong đầu vào.

Dropout Layers: Các lớp Dropout được sử dụng để ngẫu nhiên loại bỏ một phần các đơn vị trong quá trình huấn luyện, nhằm giảm overfitting.

LSTM Layers: Các lớp Bidirectional LSTM được sử dụng để học các mẫu tuần tự và thông tin phụ thuộc vào ngữ cảnh từ các đặc trưng trích xuất từ Convolutional Layers.

Dense Layers: Các lớp Dense được sử dụng để biểu diễn các đặc trưng học được thành các đặc trưng có kích thước nhỏ hơn. Các lớp Dense này cũng giúp mô hình học các kết nối phức tạp và tạo ra đầu ra dự đoán.

Output Layer: Lớp Dense cuối cùng có kích thước đầu ra là 4, được áp dụng hàm kích hoạt sigmoid nếu aspect\_only=True hoặc tanh nếu aspect\_only=False. Các giá trị đầu ra này có thể được hiểu là xác suất hoặc điểm số cho các lớp đầu ra tương ứng.

*A picture containing text, receipt

Description automatically generated*

#### Hình 3.2. Kiến trúc mô hình CNN\_LSTM

* *Mô hình PhoBERT*

Input Layer: Mô hình nhận ba đầu vào input\_ids, token\_type\_ids, và attention\_mask, đại diện cho các mã số, mã loại và mặt nạ chú ý của câu văn bản. Các đầu vào này được xử lý trong lớp TFAutoModel từ thư viện Transformers.

BERT Model: Sử dụng mô hình BERT đã được xác định trước (pretrained\_bert) từ thư viện Transformers. Đầu vào được truyền vào mô hình BERT và lấy ra các hidden states của mô hình.

Concatenate Hidden States: Các hidden states được lấy từ các layer cuối cùng trong mô hình BERT (từ lớp -4 đến lớp -1) và kết hợp thành một hidden state duy nhất. Quá trình này giúp tổng hợp thông tin từ nhiều lớp trong BERT.

Dropout: Áp dụng lớp Dropout với tỷ lệ 0.2 để ngẫu nhiên loại bỏ một phần các đơn vị trong quá trình huấn luyện, nhằm giảm overfitting.

Output Layer: Sử dụng lớp Dense để dự đoán kết quả phân loại cho từng nhãn. Mô hình sử dụng softmax activation và có số đơn vị đầu ra là 4 (tương ứng với số nhãn trong cột của df\_train).

A picture containing diagram, drawing, sketch, text

Description automatically generated

#### Hình 3.2: Kiến trúc mô hình PhoBert

Đánh giá kết quả trên tập test

A screenshot of a graph

Description automatically generated with low confidence

#### Hình 3.3: Kết quả mô hình PhoBert trên tập test

Kiểm tra kết quả trên tập test

A screenshot of a computer

Description automatically generated with low confidence

#### Hình 3.4: Kiểm tra kết quả trên tập test

## 2.4. XÂY DỰNG ỨNG DỤNG CHO MÔ HÌNH PHÂN TÍCH QUAN ĐIỂM DỰA TRÊN KHÍA CẠNH

- Độ chính xác mô hình CNN kết hợp LSTM: 0.6846

- Độ chính xác của mô hình PhoBert: 0.8043

So sánh độ chính xác giữa các mô hình chúng tôi thấy mô hình PhoBert cho độ chính xác cao nhất. Vì vậy chúng tôi chọn mô hình PhoBert để xây dựng mô hình phân tích quan điểm dựa trên khía cạnh của người dùng sử dụng văn bản tiếng Việt.

### 2.4.1. Thiết kế

Chúng tôi sử dụng ngôn ngữ lập trình Python kết hợp với các ngôn ngữ HTML, CSS và JavaScript để xây dựng ứng dụng Phân tích ý kiến người dùng theo khía cạnh. Quy trình thiết kế ứng dụng được thực hiện như sau:

* *Backend*

Sử dụng ngôn ngữ Python và thư viện Flask để xây dựng phần backend của ứng dụng.

Backend sẽ xử lý các yêu cầu từ phía client và gửi lại các phản hồi tương ứng.

Backend sẽ tương tác với các mô hình đã được lưu trữ để dự đoán và trả về kết quả phân tích ý kiến.

* *Server*

Phần server của ứng dụng được xây dựng bằng Python để hỗ trợ việc truy xuất các mô hình phân tích ý kiến đã được lưu trữ.

Server cung cấp các chức năng như sau:

Hiển thị trang phân tích văn bản bình luận trực tiếp: Cho phép người dùng nhập văn bản bình luận và hiển thị kết quả phân tích ý kiến tương ứng.

Hiển thị kết quả: Nhận đầu vào là một câu phân tích và trả về kết quả phân tích ý kiến dựa trên các mô hình đã được huấn luyện.

Hiển thị trang thống kê dữ liệu phân: Hiển thị thông tin về dữ liệu huấn luyện và kiểm tra sử dụng trong quá trình xây dựng mô hình.

Qua quy trình thiết kế trên, chúng tôi tạo ra một ứng dụng phân tích ý kiến người dùng theo khía cạnh, cho phép người dùng nhập các bình luận và nhận được kết quả phân tích ý kiến tương ứng.

A diagram of a diagram

Description automatically generated

#### Hình 4.1. Phân tích thiết kế giao diện

A diagram with black text and white text

Description automatically generated

#### Hình 4.2. Phân tích thiết kế chức năng

### 2.4.2. Thực hiện

Ở trang Aspect-Based Sentiment Analysis người dùng có thể nhập một câu đánh giá vào ô “Review” sau đó nhấn “Analyze” để xem kết quả phân tích hoặc ở ô “Result” chọn “Reset” để load lại trang

A screenshot of a computer

Description automatically generated

#### **Hình 4.3. Giao diện trang phân tích bình luận**

Người dùng có thể xem kết quả thống kê dữ liệu ở trang Statistic

A graph with blue bars

Description automatically generated

#### Hình 4.4. Giao diện trang thống kê dữ liệu

# Phần 3: KẾT LUẬN

## 1. KẾT QUẢ ĐẠT ĐƯỢC

Đề tài "Phân tích quan điểm dựa trên khía cạnh của người dùng" đã đạt được một số kết quả quan trọng trong việc phân tích ý kiến và đánh giá từ người dùng về các khía cạnh khác nhau. Dưới đây là một số kết quả đáng chú ý:

Phân tích quan điểm chính xác: Hệ thống phân tích quan điểm đã đạt được mức độ chính xác đáng kể trong việc xác định và phân loại quan điểm của người dùng theo các khía cạnh như nhà hàng, không gian, đồ ăn, dịch vụ và đồ uống. Điều này giúp người dùng có cái nhìn tổng quan về ý kiến của người khác về các khía cạnh cụ thể.

Giao diện người dùng thuận tiện: Ứng dụng được thiết kế với giao diện người dùng thân thiện và dễ sử dụng, giúp người dùng dễ dàng nhập liệu và nhận kết quả phân tích một cách nhanh chóng. Các chức năng như hiển thị trang mô hình đánh giá kết quả, phân tích dữ liệu huấn luyện và kiểm tra, và phân tích văn bản bình luận trực tiếp đều mang lại trải nghiệm tốt cho người dùng.

Tích hợp Flask và Python: Sử dụng Flask và ngôn ngữ lập trình Python đã giúp xây dựng phần backend và server một cách hiệu quả. Flask cung cấp các công cụ và thư viện hỗ trợ mạnh mẽ, trong khi Python là một ngôn ngữ linh hoạt và phổ biến trong lĩnh vực xử lý ngôn ngữ tự nhiên.

## 2. ƯU ĐIỂM VÀ NHƯỢC ĐIỂM

* *Ưu điểm*

Đáp ứng nhu cầu phân tích ý kiến: Đề tài giải quyết một vấn đề quan trọng trong việc hiểu và phân tích ý kiến của người dùng về các khía cạnh cụ thể. Điều này có thể hỗ trợ quyết định kinh doanh, nâng cao chất lượng dịch vụ và tạo trải nghiệm tốt hơn cho khách hàng.

Mở rộng và tùy chỉnh được: Hệ thống có khả năng mở rộng và tùy chỉnh để phân tích quan điểm trong nhiều lĩnh vực khác nhau và cho các khía cạnh khác nhau. Điều này tạo ra tiềm năng ứng dụng rộng lớn và linh hoạt cho công nghệ phân tích ý kiến.

* *Nhược điểm:*

Hạn chế đối với ngôn ngữ: Hiện tại, hệ thống chỉ hỗ trợ phân tích quan điểm trong tiếng Việt. Việc mở rộng hệ thống để hỗ trợ nhiều ngôn ngữ khác nhau có thể là một thách thức và yêu cầu công sức và tài nguyên phát triển.

Độ chính xác phụ thuộc vào dữ liệu: Độ chính xác của hệ thống phân tích quan điểm phụ thuộc vào dữ liệu huấn luyện. Nếu dữ liệu huấn luyện không đủ đại diện hoặc không cung cấp đủ thông tin về các khía cạnh, kết quả phân tích có thể không chính xác hoặc không đầy đủ.

Độ phức tạp của đánh giá: Đánh giá một khía cạnh trong quan điểm người dùng có thể phụ thuộc vào ngữ cảnh và có thể khá phức tạp. Hệ thống cần được cải tiến để xử lý những trường hợp phức tạp này và đảm bảo tính toàn vẹn và đúng đắn của kết quả phân tích.

Tuy có nhược điểm nhất định, nhưng đề tài "Phân tích quan điểm dựa trên khía cạnh của người dùng" mang lại nhiều ưu điểm và tiềm năng phát triển trong việc hiểu và phân tích ý kiến người dùng. Việc tiếp tục nghiên cứu và cải thiện hệ thống sẽ đóng góp vào sự phát triển của lĩnh vực phân tích ý kiến và ứng dụng trí tuệ nhân tạo.

## 3. HƯỚNG PHÁT TRIỂN

Mở rộng khả năng phân tích khía cạnh: Hiện tại, đề tài tập trung vào một số khía cạnh cụ thể như nhà hàng, không gian, đồ ăn, dịch vụ và đồ uống. Tuy nhiên, có thể mở rộng hệ thống để phân tích và hiển thị kết quả cho nhiều khía cạnh khác nhau, tùy thuộc vào lĩnh vực và nhu cầu cụ thể của người dùng.

Xử lý ngôn ngữ tự nhiên tiên tiến hơn: Nghiên cứu các phương pháp và mô hình tiên tiến trong lĩnh vực xử lý ngôn ngữ tự nhiên có thể cải thiện độ chính xác và hiệu suất của hệ thống. Các phương pháp như BERT, Transformer và các mô hình ngôn ngữ sâu hơn có thể được khám phá để cải thiện khả năng phân tích quan điểm và hiểu ý kiến.

Mở rộng ứng dụng sang các lĩnh vực khác: Đề tài có thể được mở rộng để áp dụng trong nhiều lĩnh vực khác nhau như sản phẩm công nghệ, dịch vụ du lịch, phim ảnh, và nhiều ngành công nghiệp khác. Điều này đòi hỏi việc thu thập và xử lý dữ liệu phù hợp cho từng lĩnh vực cụ thể.

Tích hợp hệ thống với dữ liệu thời gian thực: Để đáp ứng nhu cầu của người dùng, có thể tích hợp hệ thống với dữ liệu thời gian thực từ các nguồn như mạng xã hội, diễn đàn, trang đánh giá sản phẩm để cung cấp kết quả phân tích quan điểm cập nhật và chính xác hơn.

Tăng cường giao diện người dùng: Cải thiện giao diện người dùng để đảm bảo sự thuận tiện và tương tác tốt cho người dùng. Các tính năng như phân tích văn bản bình luận trực tiếp, gợi ý phân tích khía cạnh, và trực quan hóa dữ liệu có thể được phát triển để nâng cao trải nghiệm người dùng

Tối ưu hóa hiệu suất và mở rộng quy mô: Đối với ứng dụng thực tế, cần nghiên cứu và triển khai các giải pháp tối ưu hóa hiệu suất và mở rộng quy mô của hệ thống. Điều này bao gồm việc tối ưu hóa mã nguồn, xử lý song song, và triển khai hệ thống trên các nền tảng phân tán.

## 4. LỜI KẾT

Trong nghiên cứu và phát triển ứng dụng phân tích quan điểm dựa trên khía cạnh của người dùng, chúng tôi đã tiến hành xây dựng một hệ thống mạnh mẽ và hiệu quả. Đề tài này nhằm mục đích phân tích ý kiến của người dùng theo từng khía cạnh cụ thể như nhà hàng, không gian, đồ ăn, dịch vụ, đồ uống và nhiều khía cạnh khác.

Qua quá trình nghiên cứu, chúng tôi đã tìm hiểu về các phương pháp và kỹ thuật xử lý ngôn ngữ tự nhiên, phân loại cảm xúc và phân tích quan điểm. Chúng tôi đã sử dụng mô hình Machine Learning để xây dựng các mô hình phân loại và phân tích quan điểm dựa trên dữ liệu huấn luyện.

Để triển khai ứng dụng, chúng tôi sử dụng ngôn ngữ lập trình Python và các công nghệ như Flask, HTML, CSS và JavaScript. Backend của ứng dụng được xây dựng bằng Python và sử dụng thư viện Flask để xử lý các yêu cầu từ phía client. Phần server được xây dựng để truy xuất các mô hình đã được lưu trữ và trả về kết quả phân tích ý kiến cho người dùng.

Qua quá trình thiết kế và phát triển, chúng tôi đã xây dựng thành công một ứng dụng phân tích quan điểm dựa trên khía cạnh của người dùng. Ứng dụng này cho phép người dùng nhập các bình luận và nhận được kết quả phân tích ý kiến chi tiết về từng khía cạnh mà họ quan tâm. Điều này giúp người dùng có cái nhìn toàn diện và chính xác hơn về các khía cạnh của sản phẩm, dịch vụ hoặc sự kiện mà họ quan tâm.

Đề tài phân tích quan điểm dựa trên khía cạnh của người dùng đã mang lại những kết quả tích cực và hữu ích trong việc hiểu ý kiến và quan điểm của người dùng. Điều này có thể ứng dụng trong nhiều lĩnh vực, từ công nghiệp du lịch, nhà hàng, sản phẩm công nghệ cho đến lĩnh vực chính trị và xã hội. Chúng tôi hi vọng rằng đề tài này sẽ góp phần nâng cao khả năng hiểu và tương tác với người dùng trong các ứng dụng thực tế.

# TÀI LIỆU THAM KHẢO

1. [Aston Zhang](https://www.amazon.com/s/ref=dp_byline_sr_book_1?ie=UTF8&field-author=Aston+Zhang&text=Aston+Zhang&sort=relevancerank&search-alias=books), [Zachary C. Lipton](https://www.amazon.com/s/ref=dp_byline_sr_book_2?ie=UTF8&field-author=Zachary+C.+Lipton&text=Zachary+C.+Lipton&sort=relevancerank&search-alias=books), [Mu Li](https://www.amazon.com/s/ref=dp_byline_sr_book_3?ie=UTF8&field-author=Mu+Li&text=Mu+Li&sort=relevancerank&search-alias=books), [Alexander J. Smola](https://www.amazon.com/s/ref=dp_byline_sr_book_4?ie=UTF8&field-author=Alexander+J.+Smola&text=Alexander+J.+Smola&sort=relevancerank&search-alias=books), *Dive into Deep Learning,* https://d2l.djl.ai/
2. Wikipedia, *Sentiment analysis*, https://en.wikipedia.org/wiki/Sentiment\_analysis , 2022
3. Wikipedia, *Word2vec*, https://vi.wikipedia.org/wiki/Word2vec, 2022
4. Mayur Wankhade et al. Annavarapu Chandra Sekhara Rao, *A survey on sentiment analysis methods, applications, and challenges*, https://link.springer.com/article/10.1007/s10462-022-10144-1 ,2022
5. Ye Zhang et al. Byron C. Wallace, *A Sensitivity Analysis of (and Practitioners’ Guide to) Convolutional Neural Networks for Sentence Classification*, https://aclanthology.org/I17-1026.pdf, 2022
6. *Recurrent Neural Network (RNN) trong TensorFlow*, https://websitehcm.com/recurrent-neural-network-rnn-trong-tensorflow/, 2022
7. Akash Shastri, *3 neural network architectures you need to know for NLP!*, https://towardsdatascience.com/3-neural-network-architectures-you-need-to-know-for-nlp-5660f11281be, 2022
8. Amazon Vietnam, *Phân tích cảm xúc là gì?*, https://aws.amazon.com/vi/what-is/sentiment-analysis/, 2022
9. FPT Digital, *Xử lý ngôn ngữ tự nhiên: Công nghệ giúp máy tính hiểu và giao tiếp với con người*, https://digital.fpt.com.vn/dxarticles/xu-ly-ngon-ngu-tu-nhien.html, 2022
10. Ben Lutkevich, *natural language processing (NLP)*, https://www.techtarget.com/searchenterpriseai/definition/natural-language-processing-NLP, 2021
11. Bing Liu, *Sentiment Analysis and Opinion Mining, Morgan & Claypool Publishers*, May 2012
12. Nguyễn Hưng, *Deep Learning là gì? Tổng quan về Deep Learning từ A-Z*, 21-04-2022, 30-12-2022, https://vietnix.vn/deep-learning-la-gi/
13. Topdev, *Thuật toán CNN là gì? Cấu trúc mạng Convolutional Neural Network*, 21-01-2022, 29-12-2022, https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/
14. Vietnix, *Thuật toán CNN là gì? Tìm hiểu về Convolutional Neural Network*, 19-07-2022, 29-12-2022, https://vietnix.vn/cnn-la-gi/
15. Tritueviet, *Sử dụng mạng LSTM (Long Short Term Memory) để dự đoán số liệu hướng thời gian*, 21-09-2022, 29-12-2022, http://trituevietvn.com/chi-tiet/-su-dung-mang-lstm-long-short-term-memory-de-du-doan-so-lieu-huong-thoi-gian-123
16. Bùi Quang Mạnh, *Word Embedding - Tìm hiểu khái niệm cơ bản trong NLP*, 14-09-2020, 29-12-2022, https://viblo.asia/p/word-embedding-tim-hieu-khai-niem-co-ban-trong-nlp-1Je5E93G5nL
17. Openbot , *TF-IDF là gì?* , 02-08-2022, 29-12-2022, https://blog.openbot.vn/tf-idf-la-gi/
18. Nguyễn Ngọc Hoài, Lê Đình Khang, *“Phân tích cảm xúc và quan điểm bằng học sâu và ứng dụng”*, Khóa luận tốt nghiệp, Trường Đại học Sư phạm Kỹ thuật Thành phố Hồ Chí Minh, năm 2022.
19. Nguyễn Văn Hiếu, *Thuật toán CNN trong xử lí ngôn ngữ tự nhiên*, 01-07-2023, https://blog.luyencode.net/thuat-toan-cnn-trong-nlp/
20. Kelvin Jose, *RNNs, LSTMs, CNNs, Transformers and BERT*, 01-07-2023, https://medium.com/analytics-vidhya/rnns-lstms-cnns-transformers-and-bert-be003df3492b
21. Phạm Nam, *Giải mã kiến trúc transformer trong paper Attention is all you need*, 01-07-2023, <https://viblo.asia/p/giai-ma-kien-truc-transformer-trong-paper-attention-is-all-you-need-RnB5pJeGZPG>
22. Phạm Đình Khang, *BERT model*, 01-07-2023, https://phamdinhkhanh.github.io/2020/05/23/BERTModel.html#2-gi%E1%BB%9Bi-thi%E1%BB%87u-v%E1%BB%81-bert