**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**Lê Thế Thao**

**PHÂN TÍCH CẢM XÚC KHÁCH HÀNG QUA CÁC BÌNH LUẬN TRÊN TRANG WEB BÀN HÀNG**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành: Truyền thông và mạng máy tính**

**HÀ NỘI – 2017**

**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

**Lê Thế Thao**

**PHÂN TÍCH CẢM XÚC KHÁCH HÀNG QUA CÁC BÌNH LUẬN TRÊN TRANG WEB BÀN HÀNG**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**Ngành: Truyền thông và mạng máy tính**

**Cán bộ hướng dẫn: TS. Nguyễn Văn Nam**

*(ký tên)*

**HÀ NỘI – 2017**

TÓM TẮT

**Tóm tắt:** Việc triển khai cơ chế một cửa trong giải quyết thủ tục hành chính công đang dần trở nên phổ biến ở nhiều địa phương, tỉnh, thành trên cả nước giúp việc giải quyết thủ tục hành chính cho công dân trở nên đơn giản, hiệu quả hơn trước. Hơn nữa, việc đầu tư triển khai các ứng dụng công nghệ thông tin phục vụ công tác quản lý điều hành làm cho hiệu quả xử lý các công việc phục vụ công dân được nâng cao, tăng cường tính minh bạch, giảm chi phí hành chính, tiết kiệm thời gian, chi phí cho công dân, mang lại lợi nhuận tối đa cho công dân và các tổ chức. Với những ưu điểm vượt trội mà hệ thống mang lại cho công dân, cũng như cơ quan có quyền cấp phát thủ tục hành chính như vậy, công ty THHH Nghiên cứu và phát triển Công nghệ phần mềm - SELAB đang có dự án triển khai hệ thống hỗ trợ giải quyết thủ tục hành chính công một cửa cho UBND TP Thái Nguyên. Khóa luận này là một phần trong dự án của công ty triển khai cho UBND TP Thái Nguyên. Nội dung của khóa luận sẽ tập trung trình bày quy trình nghiệp vụ, yêu cầu kĩ thuật, và tiến hành phân tích thiết kế hệ thống hỗ trợ giải quyết thủ tục hành chính công một cửa.

***Từ khóa:*** một cửa, dich vụ công

LỜI CẢM ƠN

Tôi xin bày tỏ lòng biết ơn sâu sắc tới TS. Nguyễn Văn Nam, thầy đã truyền đạt kiến thức, kĩ năng trong học tập và nghiên cứu, hướng dẫn tận tâm giúp tôi hoàn thành khóa luận này.

Và tôi cũng xin chân thành cảm ơn các thầy, cô, các cán bộ làm việc tại trường Đại học Công nghệ, Đại Học Quốc Gia Hà Nội và đặc biệt là các thầy, cô trong khoa Công nghệ thông tin đã trang bị nền tảng kiến thức giúp tôi trong quá trình học tập và làm khóa luận.

Cuối cùng, tôi xin gửi lời cảm ơn tới gia đình, người thân, bạn bè đã quan tâm, động viên, tạo điều kiện cho tôi trong quá trình học tập.

|  |  |
| --- | --- |
|  | Hà Nội, ngày tháng năm  Sinh viên  Lê Thế Thao |

LỜI CAM ĐOAN

Tôi xin cam đoan đã thực hiện quá trình làm khóa luận một cách khoa học, trung thực, đúng đắn dưới sự hướng dẫn của TS.Nguyễn Văn Nam.

Các tài liệu, bài báo khoa học, giáo trình giảng dạy nếu được sử dụng trong khóa luận đều được trích dẫn đầy đủ. Nếu có điều gì sai trái tôi xin chịu trách nhiệm trước hội đồng về kết quả khóa luận của mình.

|  |  |
| --- | --- |
|  | Hà Nội, ngày tháng năm  Sinh viên  Lê Thế Thao |

MỤC LỤC

[TÓM TẮT i](#_Toc499322962)

[LỜI CẢM ƠN ii](#_Toc499322963)

[LỜI CAM ĐOAN iii](#_Toc499322964)

[MỤC LỤC iv](#_Toc499322965)

[BẢNG KÍ HIỆU VIẾT TẮT vi](#_Toc499322966)

[MỤC LỤC BẢNG BIỂU vii](#_Toc499322967)

[MỤC LỤC HÌNH VẼ viii](#_Toc499322968)

[MỞ ĐẦU 1](#_Toc499322969)

[CHƯƠNG 1. TỔNG QUAN PHÂN LỚP QUAN ĐIỂM VÀ BÀI TOÁN PHÂN TÍCH CẢM XÚC KHÁCH HÀNG 3](#_Toc499322970)

[1.1 Giới thiệu chung về phân loại cảm xúc 3](#_Toc499322971)

[1.2 Các hướng tiếp cận và giải quyết bài toán 5](#_Toc499322972)

[1.2.1 Tổng quan 5](#_Toc499322973)

[1.2.2 Mô hình xử lý dữ liệu cho bài toán phân loại cảm xúc 6](#_Toc499322974)

[1.3 Mô hình phân lớp Naïve Bayes 7](#_Toc499322975)

[1.4 Mô hình phân lớp SVM 8](#_Toc499322976)

[1.4.1 Giới thiệu về SVM 8](#_Toc499322977)

[1.4.2 Bài toán phân lớp nhị phân với SVM 9](#_Toc499322978)

[1.4.3 Bài toán phân lớp đa lớp với SVM 11](#_Toc499322979)

[1.4.4 Đánh giá bộ phân lớp SVM 11](#_Toc499322980)

[1.5 Mô hình Neural Network 11](#_Toc499322981)

[1.5.1 Kiến trúc mạng nơ-ron nhân tạo 11](#_Toc499322982)

[1.5.2 Hoạt động của mạng nơ-ron nhân tạo 14](#_Toc499322983)

[1.5.3 Mô hình Convolutional Neural Networks 16](#_Toc499322984)

[1.5.3.1 Tích chập - Convolution 16](#_Toc499322985)

[1.5.3.2 Mạng Convolutional Neural 18](#_Toc499322986)

[1.5.3.3 Các lớp trong mô hình 20](#_Toc499322987)

[1.5.3.4 Hoạt động của mô hình 21](#_Toc499322988)

[1.6 Đánh giá và kết luận 22](#_Toc499322989)

[CHƯƠNG 2: XÂY DỰNG HỆ THỐNG PHÂN TÍCH CẢM XÚC DỰA TRÊN NEURAL NETWORK 23](#_Toc499322990)

[2.1 Mô hình hệ thống 23](#_Toc499322991)

[2.1.1 Thu thập dữ liệu 23](#_Toc499322992)

[2.1.2 Tiền xử lý dữ liệu 25](#_Toc499322993)

[2.1.3 Trích xuất đặc trưng 25](#_Toc499322994)

[2.1.4 Phân loại dữ liệu 26](#_Toc499322995)

[2.1.5 Huấn luyện mô hình 26](#_Toc499322996)

[2.2 Kết quả phân lớp và đánh giá 29](#_Toc499322997)

[CHƯƠNG 3: CÀI ĐẶT HỆ THỐNG 31](#_Toc499322998)

[3.1 Mô hình cài đặt 31](#_Toc499322999)

[3.1.1 Scrapy 31](#_Toc499323000)

[3.1.2 SQLite 31](#_Toc499323001)

[3.1.3 Khung làm việc Tensorflow 31](#_Toc499323002)

[3.1.4 Django 32](#_Toc499323003)

[3.2 Cài đặt hệ thống 32](#_Toc499323004)

[3.2.1 Môi trường cài đặt 32](#_Toc499323005)

[3.2.2 Cài đặt cụ thể 33](#_Toc499323006)

[3.3 Kết quả và đánh giá 33](#_Toc499323007)

[CHƯƠNG 4: ĐÁNH GIÁ 34](#_Toc499323008)

[4.1 Hoạt động của hệ thống 34](#_Toc499323009)

[4.2 Đánh giá hiệu quả 36](#_Toc499323010)

[4.3 Kết luận 36](#_Toc499323011)

[KẾT LUẬN 36](#_Toc499323012)

[TÀI LIỆU THAM KHẢO 36](#_Toc499323013)

BẢNG KÍ HIỆU VIẾT TẮT

|  |  |
| --- | --- |
| SVM | Support Vector Machine |
| CNNs | Convolutional Neural Networks |
| OM | Opinion Mining |
| SA | Sentiment Analysis |
| ANN | Artificial Neural Network |
| NLP | Natural Language Processing |
| RELU | Rectified Linear Unit |

MỤC LỤC BẢNG BIỂU

MỤC LỤC HÌNH VẼ

MỞ ĐẦU

Ngày nay, với sự gia tăng về số lượng và nội dung các trang thương mại điện tử như lazada.vn, tiki.vn, adayroi.vn và các mạng xã hội như Facebook, Twitter,… Internet không chỉ còn là nơi cung cấp thông tin ở dạng sự kiện mà còn là nơi người dùng bày tỏ cảm xúc, trao đổi cảm nhận, kinh nghiệm về các vấn đề đời sống hoặc về các vấn đề mà chính người dùng đó đang gặp phải. Có rất nhiều các nghiên cứu trong việc khai phá nội dung văn bản để tìm ra những điều mới mẻ phục vụ kinh doanh. Chẳng hạn như nhu cầu tổng hợp những phản hồi hay bình luận của người dùng trên các trang bán hàng thương mại điện tử để đưa ra kết luận một sản phẩm có tốt hay không? Phản ứng của khách hàng đối với một sản phẩm nào đó như thế nào…Mặc dù đã có hệ thống rating đơn giản nhưng như thế vẫn chưa đủ và đáp ứng được nhu cầu của người dùng cũng như người bán hàng. Vì lẽ đó, việc kiểm tra thị hiếu của khách hàng về các nhà cung cấp, giám sát thương hiệu thông qua thu thập, phân tích những phản hồi ý kiến, đóng góp của khách hàng trên những kênh bán hàng là vô cùng quan trọng và hữu ích với các công ty, doanh nghiệp, tổ chức và các cửa hàng kinh doanh nói chung. Việc thu thập và xử lý kịp thời các thông tin này sẽ hỗ trợ tích cực cho các công ty, doanh nghiệp, tổ chức và cửa hàng thực hiện được: (i ) nắm bắt được mức độ phổ biến, lan tỏa và tầm ảnh hưởng của thương hiệu; (ii) nắm bắt được tâm tư, nguyện vọng và cả những phản hồi, góp ý trực tiếp từ cộng đồng, những người sử dụng dịch vụ, sản phẩm để từ đó đưa ra những điều chỉnh phù hợp; (iii) nắm bắt và hiểu được những phản hồi và bình luận trên diện rộng đối với các vấn đề, sự kiện quan trọng của tổ chức; (iv) kịp thời bảo vệ thương hiệu sản phẩm của đơn vị trước những thông tin dư luận thiếu chính xác và sai lệch.

Đối với lĩnh vực khai phá dữ liệu truyền thống, thay vì tập trung vào lịch sử người dùng như lịch sử mua bán, thời gian truy cập…, lĩnh vực khai phá cảm xúc người dùng lại tập trung vào việc phân tích ý nghĩa của các bình luận trên các trang bán hàng, thông tin hay mạng xã hội. Do đó, có thể hiểu bài toán phân loại cảm xúc người dùng là sự kết hợp của lĩnh vực Khai phá dữ liệu và Xử lý ngôn ngữ tự nhiên. Tính đến hiện nay, việc xử lý bài toán phân loại cảm xúc người dùng có nhiều hướng tiếp cận khác nhau và nhiều mức xử lý khác nhau, tuy nhiên các tài liệu liên quan đến việc xử lý đa phần là áp dụng cho tiếng Anh. Việc phân loại cảm xúc người dùng cho Tiếng việt còn nhiều hạn chế. Chính vì thế, luận văn “Phân tích cảm xúc khách hàng qua các bình luận trên trang web bán hàng” tập trung vào xây dựng và thực nghiệm mô hình phân lớp quan điểm bình luận của khách hàng sau đó phân tích đánh giá kết quả đó.

**Một số giải pháp hiện tại và giải pháp đề xuất**: trên thế giới đã có rất nhiều công trình nghiên cứu cũng như thực nghiệm đối bài toán phân lớp quan điểm đạt kết quả và hiệu quả tốt và được sử dụng trong rất nhiều lĩnh vực khác nhau. Trong số đó nổi bật lên là phương pháp sử dụng học máy đạt hiệu quả và cho kết quả tốt đặc biệt với những bài toán có lượng dữ liệu lớn, khổng lồ mà đòi hỏi máy tính mới đủ khả năng thực hiện được. Các thuật toán học máy có giám sát phổ biến được sử dụng trong giải quyết bài toán phân lớp quan điểm là: Naïve Bayes, Support Vector Machine (SVM), Neural Network [9]. Các thuật toán này được đánh giá cao về tính chính xác và hiệu quả trong giải quyết bài toán phân lớp quan điểm người dùng. Và trong vài năm gân đây, ứng dụng thành công của deep learning vào trong bài toán phân tích quan điểm đạt được kết quả cao.

Tuy nhiên hiện nay, các nghiên cứu cũng như ứng dụng phân lớp quan điểm đa số sử dụng và áp dụng cho tiếng Anh mà tiếng Việt thì ít, chưa nhiều. Hơn nữa, nhu cầu xây dựng các hệ thống liên quan đến phân lớp quan điểm, cảm xúc khách hàng ở Việt Nam lại nhiều và lớn. Chính vì thế, sau quá trình tìm hiểu và thực nghiệm tôi quyết định sử dụng thuật toán học máy có giám sát Neural Network, cụ thể là mô hình thuật toán Convolutional Neural Networks (CNNs) vào giải quyết bài toán này.

**Những đóng góp và kết quả đạt được:** Sau quá trình xây dựng và thực hiện chạy mô hình phân lớp quan điểm sử dụng mạng CNNs tôi đã thu được những kết quả rất khả quan , để từ đó áp dụng vào giải quyết vấn đề luận văn đã đặt ra đạt kết quả tốt và có tính tin cậy.

**Nội dung của luận văn:**

Ngoài phần giới thiệu, kết luận và tài liệu tham khảo, nội dung của luận văn bao gồm 4 chương:

Chương 1: Luận văn giới thiệu khái quát về phân loại cảm xúc cùng các khó khăn thách thức đối với bài toán. Đồng thời trình bày một số nghiên cứu liên quan và các hướng tiếp cận giải quyết bài toán cũng như giải pháp hiện thời của luận văn cho bài toán phân tích cảm xúc khách hàng qua các bình luận trên trang web bán hàng.

Chương 2: Luận văn đề xuất mô hình phân lớp quan điểm đồng thời trình bày chi tiết các pha cũng như các bước trong mô hình, huấn luyện và chạy mô hình.

Chương 3: Luận văn trình bày mô hình cài đặt hệ thống, chi tiết cài đặt các công cụ phần mềm cần thiết để chạy hệ thống.

Chương 4: Luận văn trình bày quá trình chạy thử và hoạt động của hệ thống sau đó đánh giá hiệu quả và đưa ra kết luận.

CHƯƠNG 1. TỔNG QUAN PHÂN LỚP QUAN ĐIỂM VÀ BÀI TOÁN PHÂN TÍCH CẢM XÚC KHÁCH HÀNG

1.1 Giới thiệu chung về phân loại cảm xúc

Phân tích tâm lý và khai phá quan điểm người dùng (sentiment analysis hay opinion mining) là một trong những vấn đề được nghiên cứu sôi động nhất trong lĩnh vực xử lý ngôn ngữ tự nhiên, nghiên cứu về các ý kiến, quan điểm, đánh giá, thái độ và cảm xúc của mọi người về một đối tượng.

Cảm xúc được định nghĩa là phản ứng của con người đối với các sự kiện, hiện tượng (kể cả bên trong hoặc bên ngoài cơ thể) cái mà có một ý nghĩa nào đó đối với con người. Cảm xúc, tình cảm là vấn đề được nhiều nhà khoa học quan tâm, nghiên cứu. Vì thế có nhiều quan điểm khác nhau về số lượng các loại cảm xúc. Căn cứ vào tính chất của cảm xúc có thể chia cảm xúc thành 2 loại: cảm xúc tích cực và cảm xúc tiêu cực. Căn cứ vào biểu hiện và nội dung, chúng ta có thể chia cảm xúc thành 6 loại cơ bản: vui, buồn, giận dữ, ngạc nhiên, ghét, sợ hãi. Theo nghiên cứu của W. Gerrod Parrot [9]. Từ những cảm xúc cơ bản nhưng dưới sự tác động của các kích thích khác nhau trong những điều kiện, hoàn cảnh khác nhau mà cảm xúc của con người cũng có lúc đan xen, pha lẫn nhiều cảm xúc khác loại nhưng cùng tồn tại trong một thời điểm. Và chính điều này đã tạo ra hàng loạt các cảm xúc khác.

Các đối tượng ở đây có thể là các cá nhân, các sự việc, sự vật, các dịch vụ, sản phẩm, các công ty, tổ chức, hoặc một chủ đề bất kỳ. Hai thuật ngữ Opinion Mining (OM) và Sentiment Analysis (SA) có thể được sử dụng thay thế cho nhau trong các ngữ cảnh sử dụng. Tuy nhiên, một số nhà nghiên cứu cho rằng OM và SA có một điểm khác nhau nhỏ [10]. Trong khi OM trích xuất và phân tích các ý kiến của về một đối tượng thì SA cần phải xác định các ý kiến từ một văn bản trước khi tiến hành trích xuất và phân tích chúng.

Phân tích quan điểm là một lĩnh vực thu hút được sự quan tâm lớn của cộng đồng nghiên cứu nói chung và cộng đồng xử lý ngôn ngữ nói riêng bởi ba yếu tố chính sau:

Thứ nhất, đó là sự đa dạng trong ứng dụng của nó vào nhiều lĩnh vực. Ví dụ như trong kinh doanh, việc phân tích và nắm được các ý kiến, quan điểm của khách hàng có thể giúp các công ty, tập đoàn xây dựng được những sản phẩm chất lượng cao, đáp ứng được nhu cầu của người dùng, có thể đưa ra những giải pháp kịp thời cho các vấn đề liên quan đến sản phẩm, chăm sóc khách hàng,.. trước khi mọi chuyện diễn biến xấu đi vượt tầm kiểm soát hoặc thậm chí giúp dự đoán sản lượng bán hàng trong thời gian tới.

Thứ hai, đó là sự bùng nổ của thông tin và mạng xã hội. Trong lịch sử loài người, đây là thời điểm mà lượng thông tin, lượng quan điểm trên mạng internet đang ngày càng trở nên khổng lồ, cung cấp những dữ liệu phong phú, thời gian thực, đa dạng mà không có nó, việc nghiên cứu, phân tích quan điểm người dùng là vô nghĩa. Cộng đồng người dùng Internet ngày càng phát triển và hoạt động tích cực trên các kênh mạng xã hội như Facebook, Twitter, forums, các trang báo,.. với rất nhiều các ý kiến, quan điểm riêng về mọi vấn đề trong xã hội. Tuy nhiên, vấn đề đặt ra là: mặc dù kho dữ liệu khổng lồ này (big data) chứa rất nhiều thông tin, bên cạnh các thông tin hữu ích, được các cá nhân, công ty, tổ chức quan tâm là các thông tin rác không cần thiết. Bài toán đặt ra là làm sao có thể lọc được các thông tin hữu ích này từ kho dữ liệu khổng lồ đó.

Thứ ba, đó là sự thách thức của bài toán. Phân tích quan điểm người dùng có thể được chia ra làm nhiều bài toán nhỏ hơn và cũng đầy thách thức với các nhà nghiên cứu như các bài toán phân lớp chủ quan và khách quan (subjectivity classification), phân lớp ý kiến trái chiều (sentiment polarity classification), phát hiện ý kiến rác (spam opinion detection) [1], tóm tắt và tổng hợp quan điểm (opinion summarization), phân tích tính đa diện của của một ý kiến (dual sentiment analysis)[2].

Quan điểm được chia làm hai loại: tích cực (positive) và tiêu cực (negative).

Ngoài hai trạng thái này, một câu hoặc văn bản được xếp vào dạng trung lập (neutral).

Bài toán phân tích quan điểm người dùng thường được tiếp cận và giải quyết ở ba mức độ [5]:

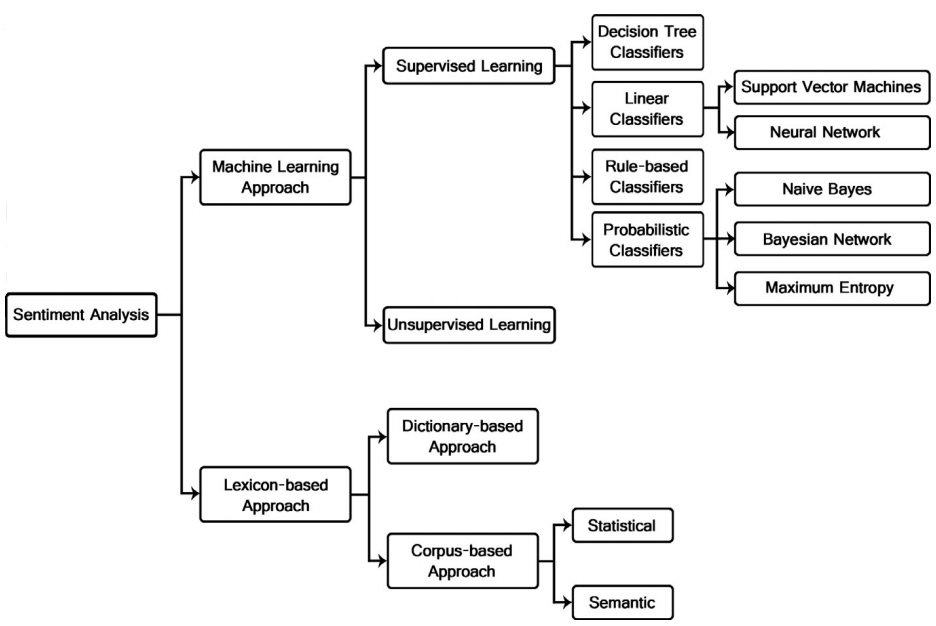
* **Mức độ văn bản, tài liệu** (Document level): ở mức độ này, bài toán cần phân loại xem một văn bản hay tài liệu thể hiện ý kiến tiêu cực hay tích cực. Ví dụ như bài viết đánh giá về một sản phẩm điện thoại thể hiện ý kiến, nhận định chủ yếu là tốt hay không tốt, tích cực hay tiêu cực. Mức độ này được thực hiện với giả sử rằng tài liệu chỉ đưa ra các quan điểm, ý kiến về một thực thể duy nhất chứ không có sự so sánh giữa các thực thể khác nhau.
* **Mức độ câu** (Sentence level) [3]: các phương pháp được áp dụng cho mức độ tài liệu cũng có thể được áp dụng ở mức độ câu. Trong trường hợp đơn giản, các câu chỉ chứa một ý kiến, quan điểm về một thực thể. Trong các trường hợp phức tạp hơn, một câu có thể có nhiều quan điểm, đánh giá về các khía cạnh khác nhau của một đối tượng hoặc thậm chí có thể có sự thay đổi về quan điểm trong cùng một câu (polarity shifting) [16]. Mức độ phân tích quan điểm cho câu rất gần với bài toán phân lớp chủ quan và khách quan, trong đó chúng ta cần phân loại xem một câu đã cho là chủ quan (có quan điểm, ý kiến riêng) hay khách quan (câu chỉ đưa ra thông tin). Tuy nhiên, các câu khách quan cũng có thể từ đó suy ra quan điểm. Ví dụ như câu: ”Mình nhận là hàng fake, không phải chính hãng Samsung, thông tin in mờ”. Trong câu nói này, các mệnh đề đều là nhìn nhận khách quan trong thực tế nhưng từ đó có thể suy luận ra ý kiến chê bai chất lượng sản phẩm và cũng như cách bán hàng của cửa hàng đó.
* **Mức độ khía cạnh** (Aspect level): nếu với hai mức độ nêu trên, vấn đề được tiếp cận theo hướng kiến trúc của văn bản, ngôn ngữ (câu, đoạn, tài liệu, cú pháp), thì ở mức độ khía cạnh, bài toán tập trung vào chính quan điểm, ý kiến được đưa ra, phân tích ở mức độ sâu hơn, đó là phân tích xem ý kiến tiêu cực hay tích cực của là về chủ đề, đối tượng nào [4].

1.2 Các hướng tiếp cận và giải quyết bài toán

1.2.1 Tổng quan

Trong những năm gần đây, có rất nhiều bài báo và các công trình nghiên cứu cải tiến các thuật toán phân tích quan điểm người dùng. Các kỹ thuật này có thể được phân loại như trong Hình 1.1 [7]. Trong đó ta thấy, có hai hướng tiếp cận chính trong các kỹ thuật ứng dụng trong giải quyết bài toán phân lớp quan điểm người dùng, đó là: sử dụng các thuật toán học máy hoặc tiếp cận theo hướng sử dụng các kiến thức về từ vựng và ngữ nghĩa. Trong các thuật toán học máy lại có thể được chia ra thành các thuật toán học có giám sát hay học không giám sát. Ngoài ra, trong một, hai năm trở lại đây bắt đầu xuất hiện các ứng dụng thành công của deep learning vào trong bài toán phân tích quan điểm [12,13] đạt kết quả cao.

Các thuật toán học máy có giám sát phổ biến được sử dụng trong giải quyết bài toán phân lớp quan điểm là: Naïve Bayes, Support Vector Machine (SVM), Neural Network [9]. Các thuật toán này được đánh giá cao về tính chính xác và hiệu quả trong giải quyết bài toán phân lớp quan điểm người dùng. Trong phần sau, chúng tôi sẽ giới thiệu tổng quan về các giải thuật học có giám sát này.



Hình 1.1 Các kỹ thuật sử dụng trong giải quyết bài toán phân lớp quan điểm

1.2.2 Mô hình xử lý dữ liệu cho bài toán phân loại cảm xúc

Mặc dù có những hướng tiếp cận để giải quyết bài toán khác nhau, tuy nhiên nhóm nghiên cứu của Pravesh đã đưa ra mô hình chung cho bài toán này trong hội nghị quốc tế về công nghệ tính toán mềm năm 2014 (International Journal on Soft Computing-IJSC) [17].

Hình 1.2 Mô hình xử lý dữ liệu cho bài toán phân loại cảm xúc

#### 1.2.2.1. Thu thập dữ liệu

Bước này sẽ thu thập các trang web chứa bình luận người dùng. Để thực hiện bước này có thể sử dụng cơ sở dữ liệu của trang web hoặc sử dụng Crawler hoặc Agent để thực hiện tự động. Kết quả bước này sẽ thu thập được toàn bộ các định dạng dữ liệu của website đối tượng ở dạng HTML hay TXT, dữ liệu này sẽ được đi qua xử lý ở các bước sau.

#### 1.2.2.2. Chuẩn hóa dữ liệu

Về mặt cấu trúc, dữ liệu thu được ở bước thứ nhất ở định dạng HTML hoặc PHP, do đó bước Chuẩn hóa dữ liệu sẽ lọc lấy những thông tin cần thiết như thời gian, tiêu đề bài viết cùng với bình luận người dùng. Ngoài ra, những bình luận người dùng thu thập được có thể không thỏa mãn những yêu cầu về ngữ pháp hoặc ngữ nghĩa. Bước này cũng sẽ loại bỏ những mẫu bình luận không phù hợp hoặc sửa đổi (như thêm dấu với Tiếng Việt) để đảm bảo dữ liệu thu thập được phù hợp với việc Gán nhãn hay Trích chọn đặc trưng ở bước sau.

#### 1.2.2.3. Gán nhãn dữ liệu

Với cách tiếp cận bằng bộ từ vựng, bước này sẽ sử dụng dữ liệu từ bộ từ vựng đó để so khớp hay gán nhãn những từ trong văn bản tương ứng trong từ điển. Với phương pháp tiếp cận bằng thuật toán học máy, dữ liệu sẽ được gán nhãn dựa vào đặc trưng ngôn ngữ của văn bản. Cụ thể như gán nhãn câu, gán nhãn từ và từ loại hay các biểu tượng cảm xúc trong bình luận.

#### 1.2.2.4. Trích chọn đặc trưng

Dựa trên dữ liệu được gán nhãn, đặc trưng để phân lớp là trọng số hướng ngữ nghĩa của bình luận với cách sử dụng bộ từ vựng. Các thuật toán học máy sẽ sử dụng đặc trưng ngôn ngữ như n-grams sau khi tách từ hay gán nhãn từ loại để huấn luyện và kiểm thử với bộ dữ liệu chuẩn được chọn ra từ các bước trước đó.

#### 1.2.2.5. Phân lớp dữ liệu

Dữ liệu đầu vào đã qua các bước tiền xử lý sẽ qua bộ phân lớp sử dụng các thuật toán học máy. Với cách tiếp cận bằng bộ từ vựng, bước này sẽ áp dụng các luật ngữ pháp hay các quy tắc thay đổi ngữ nghĩa để tính ra trọng số cuối cùng sau đó quyết định hướng ngữ nghĩa của bình luận.

1.3 Mô hình phân lớp Naïve Bayes

Bộ phân lớp quan điểm Naïve Bayes được xây dựng dựa trên lý thuyết Bayes về xác suất có điều kiện và sử dụng mô hình “bag of words” để phân loại văn bản:

Mục tiêu là tìm được phân lớp c\* sao cho là lớn nhất hay xác suất của tài liệu d thuộc lớp c\* là lớn nhất.

Từ công thức trên ta có thể nhận thấy P(d) không đóng vai trò gì trong việc quyết định phân lớp c -> P(c|d) lớn nhất ⟺ P(c).P(d|c) lớn nhất.

Để có thể xấp xỉ giá trị của P(d|c), thuật toán Naïve Bayes giả sử rằng: các vector đặc trưng fi của một tài liệu khi đã biết phân lớp là độc lập với nhau. Từ đó ta có công thức:

Trong đó f là các vector đặc trưng cho tài liệu d.

Khi tiến hành huấn luyện, thuật toán sử dụng phương pháp xấp xỉ hợp lý cực đại MLE (Maximum Likelihood Estimation) để xấp xỉ P(c) và cùng thuật toán làm mịn add-one (add-one smoothing). Ta có:

Trong đó Nc là số văn bản được phân loại vào lớp c; N là tổng số văn bản trong tập huấn luyện.

Trong đó là số lần xuất hiện của vector đặc trưng i trong tài liệu thuộc phân lớp c. Đánh giá bộ phân lớp sử dụng thuật toán học máy Naive Bayes, ta nhận thấy phương pháp này các ưu điểm như: đơn giản, dễ cài đặt, bộ phân lớp chạy nhanh và cần ít bộ nhớ lưu trữ. Bộ phân lớp cũng không cần nhiều dữ liệu huấn luyện để xấp xỉ được bộ tham số. Tuy nhiên, bộ phân lớp này có nhược điểm là thiếu chính xác do giả thiết độc lập của các vector đặc trưng khi đã biết phân lớp là không có thực trong thực tế.

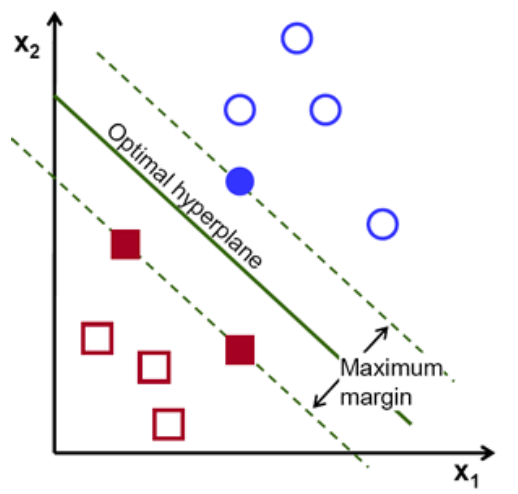
1.4 Mô hình phân lớp SVM

### 1.4.1 Giới thiệu về SVM

Máy vector hỗ trợ (SVM) là một phương pháp học máy nổi tiếng được sử dụng để giải quyết bài toán phân lớp, thuật toán được Vladimir N. Vapnik tìm ra và thuật toán SVM tiêu chuẩ hiện nay sử dụng được tìm ra bởi Vapnik và Corinna Cortes vào năm 1995. SVM xuất phát từ lý thuyết học thống kê, dựa trên nguyên tắc tối thiểu rủi ro cấu trúc (Structural Risk Minimisation) và cố gắng phân loại dữ liệu sao cho lỗi xảy ra trên tập kiểm tra là nhỏ nhất (Test Error Minimisation). Nhiều bài toán trong đời sống thực được SVM giải quyết khá thành công như nhận dạng văn bản, hình ảnh, chữ viết tay, phân loại thư rác điện tử, virus…

Thuật toán SVM ban đầu chỉ được thiết kế để giải quyết bài toán phân lớp nhị phân, tức là số lớp hạn chế là hai lớp, với ý tưởng chính như sau:

Cho trước một tập huấn luyện, được biểu diễn trong không gian vector với mỗi điểm là biểu diễn của một dữ liệu, SVM sẽ tìm ra một siêu phẳng f quyết định tốt nhất có thể chia các điểm trên không gian này thành hai lớp riêng biệt, tương ứng là lớp “+” và lớp “-”. Chất lượng của siêu phẳng được đánh giá bởi khoảng cách lề (margin) giữa hai lớp: khoảng cách càng lớn thì siêu phẳng quyết định càng tốt và chất lượng phân lớp càng cao.



Hình 1.3 Ví dụ về siêu phẳng trong SVM

Trong ví dụ như Hình 1.2, đường thẳng liền nét tô màu xanh lá chính là siêu phẳng tốt nhất để phân tách dữ liệu thành hai lớp khác nhau. Hai bên của siêu phẳng là hai lề, chứa các vector hỗ trợ (support vectors) – tức là các điểm dữ liệu gần siêu phẳng nhất.

1.4.2 Bài toán phân lớp nhị phân với SVM

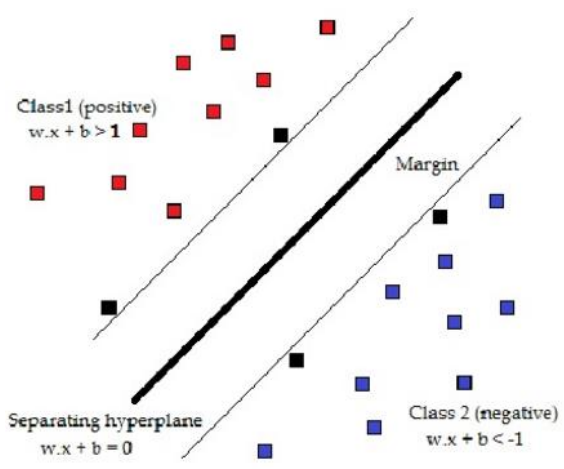
* **Phát biểu bài toán:**

Cho tập mẫu {(x1, y 1), (x2, y2), … (xD, yD)} trong đó xi ∈ RD và yi ∈ {-1, +1}. Giả sử dữ liệu là phân tách tuyến tính, tức là ta có thể phân tách dữ liệu thành hai lớp bằng cách vẽ một đường phẳng trên đồ thị của x1, x2 (với D = 2) hoặc một siêu phẳng trên đồ thị của x1, x2,… xD (với D > 2). Mục đích của thuật toán phân lớp SVM là xây dựng siêu phẳng sao cho khoảng cách lề giữa hai lớp đạt cực đại bằng cách xác định phương trình mô tả siêu phẳng đó trên đồ thị.

* **Phương pháp giải bài toán:**

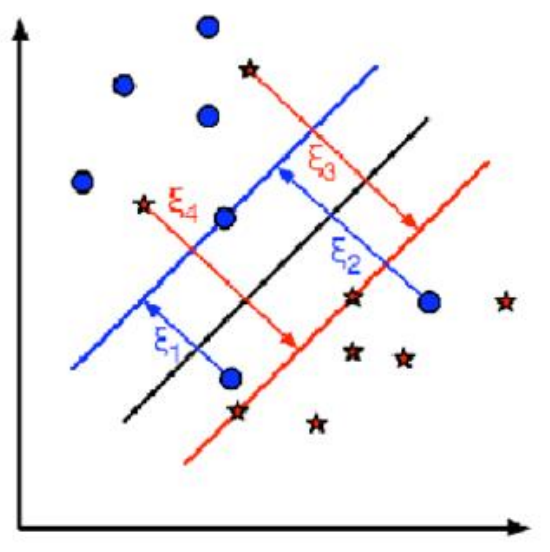
Bài toán này xảy ra ba trường hợp, mỗi trường hợp có một bài toán tối ưu và giải được bài toán này ta sẽ xây dựng được siêu phẳng cần tìm.

* Trường hợp 1: Tập dữ liệu có thể phân chia tuyến tính được mà không có nhiễu, tức là mọi điểm có nhãn “+1” nằm về phía dương trong khi mọi điểm có nhãn “-1” đều nằm về phía âm của mặt phẳng.



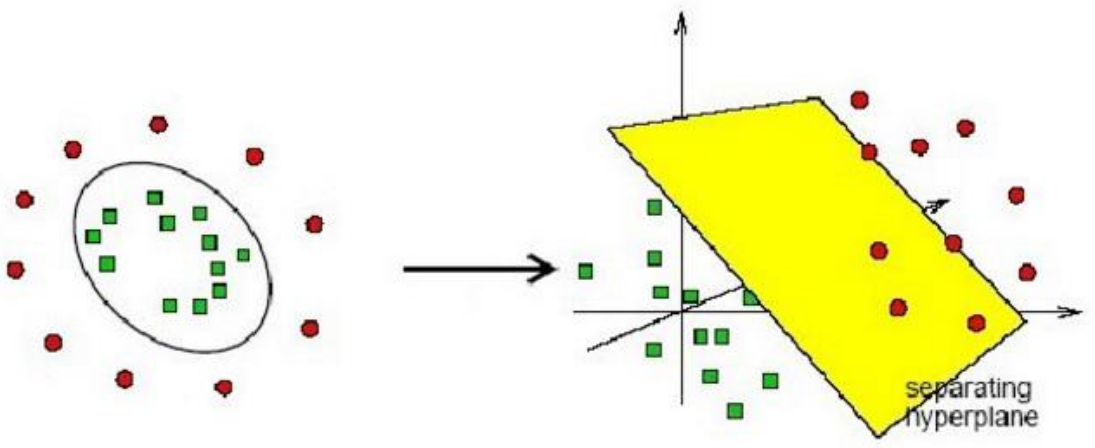
Hình 1.4 Trường hợp phân chia tuyến tính nhị phân sử dụng SVM không có nhiễu

* Trường hợp 2: Tập dữ liệu có thể phân chia tuyến tính được nhưng có nhiễu, tức là hầu hết các điểm được phân chia đúng bởi siêu phẳng nhưng có tồn tại một vài điểm nhiễu (điểm có nhãn “+1” lại nằm về phía âm của siêu phẳng, hoặc ngược lại).



Hình 1.5 Trường hợp phân chia tuyến tính nhị phân sử dụng SVM có nhiễu

* Trường hợp 3: Tập dữ liệu không thể phân chia tuyến tính được. Ta sẽ thực hiện phép ánh xạ các vector dữ liệu x vào một không gian khác có nhiều chiều hơn chiều không gian hiện tại sao cho trong không gian này, tập dữ liệu có thể phân chia tuyến tính được.



Hình 1.6 Trường hợp không thể phân chia tuyến tính nhị phân sử dụng SVM

1.4.3 Bài toán phân lớp đa lớp với SVM

Đối với bài toán phân lớp với số lớp nhiều hơn hai lớp, ta sử dụng kỹ thuật phân đa lớp dạng Multiple Binary Classification với hai chiến lược chính là One-vsOne và One-vs-Rest:

* Chiến thuật One-vs-One: Nếu k là số lớp cần phân tách, chiến lược này sẽ thực hiện k.(k-1)/2 lần phân lớp nhị phân SVM. Cụ thể: ta sẽ bắt cặp từng hai lớp một và sử dụng phương pháp chọn đa số để kết hợp các bộ phân lớp lại thành kết quả phân lớp cuối cùng.
* Chiến thuật One-vs-Rest: Chiến lược này sử dụng (k-1) bộ phân lớp đối với k lớp, tức là chuyển bài toán phân lớp k lớp thành k bài toán phân lớp nhị phân. Trong đó bộ phân lớp nhị phân thứ i được xây dựng trên lớp thứ i và các lớp còn lại.

1.4.4 Đánh giá bộ phân lớp SVM

Bộ phân lớp SVM có các ưu điểm như:

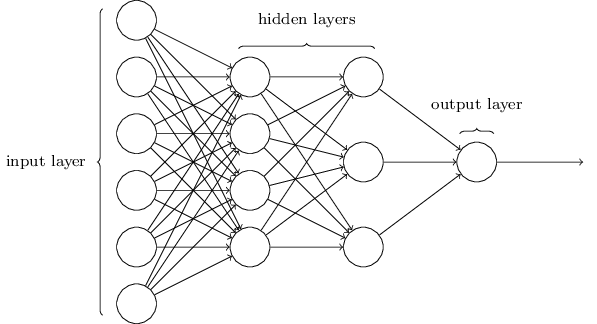
* Độ chính xác phân lớp cao, yêu cầu kích thước bộ dữ liệu huấn luyện nhỏ, dễ áp dụng cho nhiều bài toán.
* Hiệu quả với các bài toán phân lớp dữ liệu có số chiều lớn.
* Hiệu quả với các trường hợp số chiều dữ liệu lớn hơn số lượng mẫu.

Tuy nhiên, bộ phân lớp SVM còn có một số nhược điểm:

* Thời gian huấn luyện lâu, không gian bộ nhớ sử dụng lớn, được thiết kế cho phân lớp nhị phân (trong khi thực tế chủ yếu là phân loại đa lớp).
* Có thể bị overfit trên dữ liệu huấn luyện, nhạy cảm với nhiễu.
  1. Mô hình Neural Network
     1. Kiến trúc mạng nơ-ron nhân tạo

Mạng nơ ron nhân tạo (Artificial Neural Network – ANN) là một mô hình xử lý thông tin được mô phỏng dựa trên hoạt động của hệ thống thần kinh của sinh vật, bao gồm số lượng lớn các Nơ-ron được gắn kết để xử lý thông tin. ANN hoạt động giống như bộ não của con người, được học bởi kinh nghiệm (thông qua việc huấn luyện), có khả năng lưu giữ các tri thức và sử dụng các tri thức đó trong việc dự đoán các dữ liệu chưa biết (unseen data).

Một mạng nơ-ron là một nhóm các nút nối với nhau, mô phỏng mạng nơ ron thần kinh của não người. Mạng nơ ron nhân tạo được thể hiện thông qua ba thành phần cơ bản: mô hình của nơ ron, cấu trúc và sự liên kết giữa các nơ ron.Trong nhiều trường hợp, mạng nơ ron nhân tạo là một hệ thống thích ứng, tự thay đổi cấu trúc của mình dựa trên các thông tin bên ngoài hay bên trong chạy qua mạng trong quá trình học.



Hình 1.7 Kiến trúc mạng nơ-ron nhân tạo

Kiến trúc chung của một ANN gồm 3 thành phần đó là **Input Layer**, **Hidden Layer** và **Output Layer** (Hình 2.1)

Trong đó, lớp ẩn (**Hidden Layer**) gồm các nơ-ron, nhận dữ liệu input từ các Nơ-ron ở lớp (Layer) trước đó và chuyển đổi các input này cho các lớp xử lý tiếp theo. Trong một mạng ANN có thể có nhiều Hidden Layer.

ANN được huấn luyện theo ba kỹ thuật cơ bản đó là: học có giám sát (Supervised learning) và học không giám sát (Unsupervised learning) và học tăng cường (Reinforcement Learning )

* Học có giám sát (Supervised learning): một thành phần quan trọng trong phương pháp này đó là người thầy với kiến thức vô cùng phong phú thể hiện qua tập dữ liệu training. ANN sẽ phải tìm cách thay đổi các tham số (parameter) và các ngưỡng (threshold) để ánh xạ các đầu vào thành các đầu ra mong muốn. ví dụ: trong nhận dạng khuôn mặt. Người thầy sẽ cung cấp cho hệ thống tập hợp các loại khuôn mặt và tên gắn cho từng khuôn mặt. Các mạng neural sẽ phải dự đoán sau đó so sánh với kết quả đúng mà người thầy cung cấp. Từ đó, hệ thống neural sẽ phải điều chỉnh các sai sót của mình để những lần sau đó có thể đưa ra các câu trả lời chính xác hơn.
* Học không giám sát (Unsupervised learning): khác biệt với học có giám sát ở chỗ là đầu ra đúng tương ứng cho mỗi đầu vào là không biết trước. Trong học không có giám sát, một tập dữ liệu đầu vào được thu thập. Nó thường đối xử với các đối tượng đầu vào như là một tập các biến ngẫu nhiên. Sau đó, một mô hình mật độ kết hợp sẽ được xây dựng cho tập dữ liệu đó. Hãy tưởng tượng tìm kiếm một mô hình ẩn trong một tập dữ liệu.Một ứng dụng của việc này là phân cụm (clustering), nghĩa là chia một tập các phần tử thành các nhóm theo một số mô hình không rõ.
* Học tăng cường (Reinforcement Learning): Chiến lược gây dựng của phương pháp này thông qua những quan sát. Bạn có thể tưởng tượng một con chuột nhỏ chạy trong một mê cung. Nếu nó rẽ trai, nó sẽ nhận được một miếng pho mát, nhưng nếu nó rẽ phải, ở đó có một chiếc bẫy chuột (yaoming). Theo thời gian, con chuột sẽ học được rằng nó cần phải rẽ trái. Mạng nerural của nó sẽ phải quan sát môi trường xung quanh, thu thập thông tin và đưa ra các quyết định rẽ trái học rễ phải. Nếu những quan sát của nó không tốt, mạng neural sẽ phải điều chỉnh các trọng số của nó để đưa ra các quyết định khác trong thời gian tới. Phương pháp học tăng cường thường phổ biến trong các robot.

Mạng nơron nhân tạo đã được sử dụng để giải quyết nhiều bài toán thuộc nhiều lĩnh vực của các ngành khác nhau. Các nhóm ứng dụng mà mạng nơ-ron nhân tạo đã được áp dụng rất có hiệu quả là:

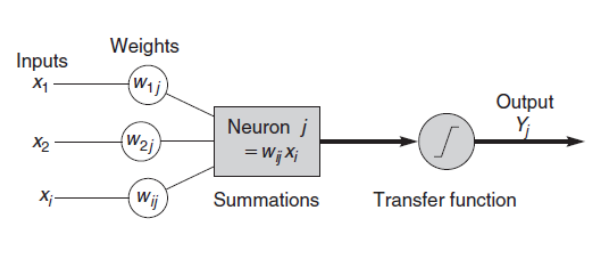
* Bài toán phân lớp (classification): Phân loại các đối tượng quan thành thành các nhóm. Ví dụ: phân loại chữ viết, văn bản, nhận diện hình ảnh ...
* Bài toán dự đoán (preditive): Mạng nơ-ron nhân tạo đã được ứng dụng thành công trong việc xây dựng các mô hình dự báo sử dụng tập dữ liệu trong quá khứ để dự đoán số liệu trong tương lai. Ví dụ: dữ báo thiên tai, dự báo chứng khoán ...
* Bài toán điều khiển và tối ưu hoá: Nhờ khả năng học và xấp xỉ hàm mà mạng nơ-ron nhân tạo đã được sử dụng trong nhiều hệ thống điều khiển tự động cũng như góp phần giải quyết những bài toán tối ưu trong thực tế.

Lợi thế lớn nhất của các mạng ANN là khả năng được sử dụng như một cơ chế xấp xỉ hàm tùy ý mà “học” được từ các dữ liệu quan sát. Tuy nhiên, sử dụng chúng không đơn giản như vậy, một số các đặc tính và kinh nghiệm khi thiết kế một mạng nơ-ron ANN.

* **Chọn mô hình**: Điều này phụ thuộc vào cách trình bày dữ liệu và các ứng dụng. Mô hình quá phức tạp có xu hướng dẫn đến những thách thức trong quá trình học.
* **Cấu trúc và sự liên kết giữa các nơ-ron**
* **Thuật toán học:** Có hai vấn đề cần học đối với mỗi mạng ANN, đó là học tham số của mô hình (parameter learning) và học cấu trúc (structure learning). Học tham số là thay đổi trọng số của các liên kết giữa các nơ-ron trong một mạng, còn học cấu trúc là việc điều chỉnh cấu trúc mạng bằng việc thay đổi số lớp ẩn, số nơ-ron mỗi lớp và cách liên kết giữa chúng. Hai vấn đề này có thể được thực hiện đồng thời hoặc tách biệt.

Nếu các mô hình, hàm chi phí và thuật toán học được lựa chọn một cách thích hợp, thì mạng ANN sẽ cho kết quả có thể vô cùng mạnh mẽ và hiệu quả.

* + 1. Hoạt động của mạng nơ-ron nhân tạo



Hình 1.8 Quá trình xử lý thông tin của một mạng nơ-ron nhân tạo

**Inputs**: Mỗi Input tương ứng với 1 đặc trưng của dữ liệu. Ví dụ như trong ứng dụng của ngân hàng xem xét có chấp nhận cho khách hàng vay tiền hay không thì mỗi input là một thuộc tính của khách hàng như thu nhập, nghề nghiệp, tuổi, số con,…

**Output**: Kết quả của một ANN là một giải pháp cho một vấn đề, ví dụ như với bài toán xem xét chấp nhận cho khách hàng vay tiền hay không thì output là yes hoặc no.

**Connection Weights** (Trọng số liên kết) : Đây là thành phần rất quan trọng của một ANN, nó thể hiện mức độ quan trọng, độ mạnh của dữ liệu đầu vào đối với quá trình xử lý thông tin chuyển đổi dữ liệu từ Layer này sang layer khác. Quá trình học của ANN thực ra là quá trình điều chỉnh các trọng số Weight của các dữ liệu đầu vào để có được kết quả mong muốn.

**Summation Function** (Hàm tổng): Tính tổng trọng số của tất cả các input được đưa vào mỗi Nơ-ron. Hàm tổng của một Nơ-ron đối với n input được tính theo công thức sau:

**Transfer Function** (Hàm chuyển đổi): Hàm tổng của một nơ-ron cho biết khả năng kích hoạt của nơ-ron đó còn gọi là kích hoạt bên trong. Các nơ-ron này có thể sinh ra một output hoặc không trong ANN, nói cách khác rằng có thể output của 1 Nơ-ron có thể được chuyển đến layer tiếp theo trong mạng Nơ-ron hoặc không. Mối quan hệ giữa hàm tổng và kết quả output được thể hiện bằng hàm chuyển đổi.

Việc lựa chọn **Transfer Function** có tác động lớn đến kết quả của ANN. Hàm chuyển đổi phi tuyến được sử dụng phổ biến trong ANN là sigmoid (logical activation) function.

)

Kết quả của Sigmoid Function thuộc khoảng [0,1] nên còn gọi là hàm chuẩn hóa (Normalized Function).

Kết quả xử lý tại các nơ-ron (Output) đôi khi rất lớn, vì vậy transfer function được sử dụng để xử lý output này trước khi chuyển đến layer tiếp theo. Đôi khi thay vì sử dụng Transfer Function người ta sử dụng giá trị ngưỡng (Threshold value) để kiểm soát các output của các neuron tại một layer nào đó trước khi chuyển các output này đến các Layer tiếp theo. Nếu output của một neuron nào đó nhỏ hơn Threshold thì nó sẻ không được chuyển đến Layer tiếp theo.

Mạng nơ-ron của chúng ta dự đoán dựa trên forward propagation là các phép nhân ma trận cùng với activation function để thu được kết quả đầu ra. Nếu input x là vector 2 chiều thì ta có thể tính kết quả dự đoán bằng công thức sau:

Trong đó, zi là input của layer thứ i, ai là output của layer thứ sau khi áp dụng activation function. W1, b1, W2, b2 là các thông số (parameters) cần tìm của mô hình mạng nơ-ron.

Huấn luyện để tìm các thông số cho mô hình tương đương với việc tìm các thông số W1, b1, W2, b2, sao cho độ lỗi của mô hình đạt được là thấp nhất. Ta gọi hàm độ lỗi của mô hình là loss function. Đối với softmax function, ta dùng cross-entropy loss (còn gọi là negative log likelihood).

Nếu ta có N dòng dữ liệu huấn luyện, và C nhóm phân lớp (trường hợp này là hai lớp nam, nữ), khi đó loss function giữa giá trị dự đoán và được tính như sau:

Ý nghĩa công thức trên nghĩa là: lấy tổng trên toàn bộ tập huấn luyện và cộng dồn vào hàm loss nếu kết quả phân lớp sai. Độ dị biệt giữa hai giá trị và càng lớn thì độ lỗi càng cao. Mục tiêu của chúng ta là tối thiểu hóa hàm lỗi này. Ta có thể sử dụng phương pháp gradient descent để tối tiểu hóa hàm lỗi. Có hai loại gradient descent, một loại với fixed learning rate được gọi là batch gradient descent, loại còn lại có learning rate thay đổi theo quá trình huấn luyện được gọi là SGD (stochastic gradient descent) hay minibatch gradient descent.

Gradient descent cần các gradient là các vector có được bằng cách lấy đạo hàm của loss function theo từng thông số . Để tính các gradient này, ta sử dụng thuật toán backpropagation (lan truyền ngược). Đây là cách hiệu quả để tính gradient khởi điểm từ output layer.

Áp dụng backpropagation ta có các đại lượng:

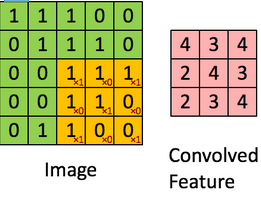
* + 1. Mô hình Convolutional Neural Networks

CNNs là một trong những thuật toán Deep Learning cho kết quả tốt nhất hiện nay trong hầu hết các bài toán về thị giác máy như phân lớp, nhận dạng, …

Hiện nay, CNNs đã được áp dụng vào các vấn đề trong xử lý ngôn ngữ tự nhiên (NLP - Natural Language Processing) và nhận được một số kết quả tốt. Trong phần này tôi sẽ tóm tắt CNNs là gì và cách chúng được sử dụng trong NLP.

* + - 1. Tích chập - Convolution

Convolution là một cửa sổ trượt (Sliding Windows) trên một ma trận như mô tả hình dưới:

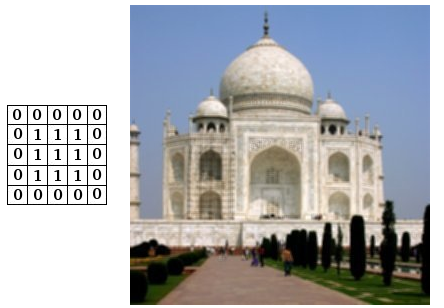


Hình 1.9 Cửa sổ tích chập

Ma trận bên trái là một hình ảnh trắng đen được số hóa. Ma trận có kích thước 5x5 và mỗi điểm ảnh có giá trị 1 hoặc 0 là giao điểm của dòng và cột. Sliding Window hay còn gọi là kernel, filter hoặc feature detect là một ma trận có kích thước nhỏ như trong ví dụ trên là 3x3. Convolution hay tích chập là nhân từng phần tử bên trong ma trận 3x3  với ma trận bên trái. Kết quả được một ma trận gọi là Convoled feature được sinh ra từ việc nhận ma trận Filter với ma trận ảnh 5x5 bên trái.

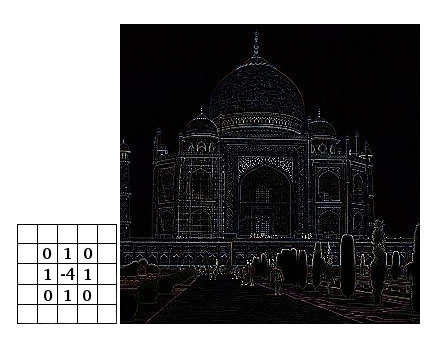
Công dụng của tích chập:

* Nếu lấy giá trị trung bình xung quanh cho vị trí điểm ảnh trung tâm sẽ làm mờ bức ảnh ban đầu.



Hình 1.10 Làm mờ bức ảnh bằng tích chập

* Nếu tính tích phân sẽ phát hiện được các biên cạnh giữa các điểm ảnh lân cận bằng cách thêm vào hoặc hủy các giá trị bằng 0 trong ma trận. Nếu có một đường sắt nét nghĩa là giá trị điểm ảnh tại vị trí đó cao thì quá trình biến đổi sẽ tạo ra một sự tách biệt giữa các giá trị màu trắng và đen khi đó sẽ phát hiện ra các đường biên rõ.



Hình 1.11 Phát hiện các đường biên của bức ảnh bằng tích chập

* + - 1. Mạng Convolutional Neural

CNNs về cơ bản chỉ là một vài lớp của tích chập và sử dụng các hàm kích hoạt phi tuyển (nonlinear activation) như ReLU và tanh để áp dụng vào kết quả. Trong mô hình mạng truyền ngược (feedforward neural network) thì mỗi neural đầu vào cho mỗi neural đầu ra trong các lớp tiếp theo. Mô hình này gọi là mạng kết nối đầy đủ (fully connected layer) hay mạng toàn vẹn (affine layer). Còn trong mô hình CNNs thì không làm vậy mà sử dụng convolutions trên lớp đầu vào để tính đầu ra. Layer tiếp theo sử dụng kết quả convolution từ layer trước đó, điều này dẫn đến kết nối cục bộ.

Mỗi lớp đều áp dụng các bộ lọc khác nhau, thường là hàng trăm hoặc hàng ngàn và kết hợp các kết quả của chúng lại. Ngoài ra có một số layer khác như pooling (subsampling) layer dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu). Trong quá trình huấn luyện mạng (traning)  CNNs tự động học các giá trị qua các lớp bộ lọc (filter) dựa vào cách thức mà bạn thực hiện. Ví dụ: trong phân loại hình ảnh CNNs có thể tìm hiểu để phát hiện các cạnh từ các pixel thô trong lớp đầu tiên, sau đó sử dụng các cạnh để phát hiện các hình dạng đơn giản trong lớp thứ hai và sau đó sử dụng các hình dạng này để ngăn các đặc trưng mức cao hơn, chẳng hạn như hình dạng khuôn mặt ở các lớp cao hơn. Lớp cuối cùng là phân loại sử dụng những đặc trưng này.

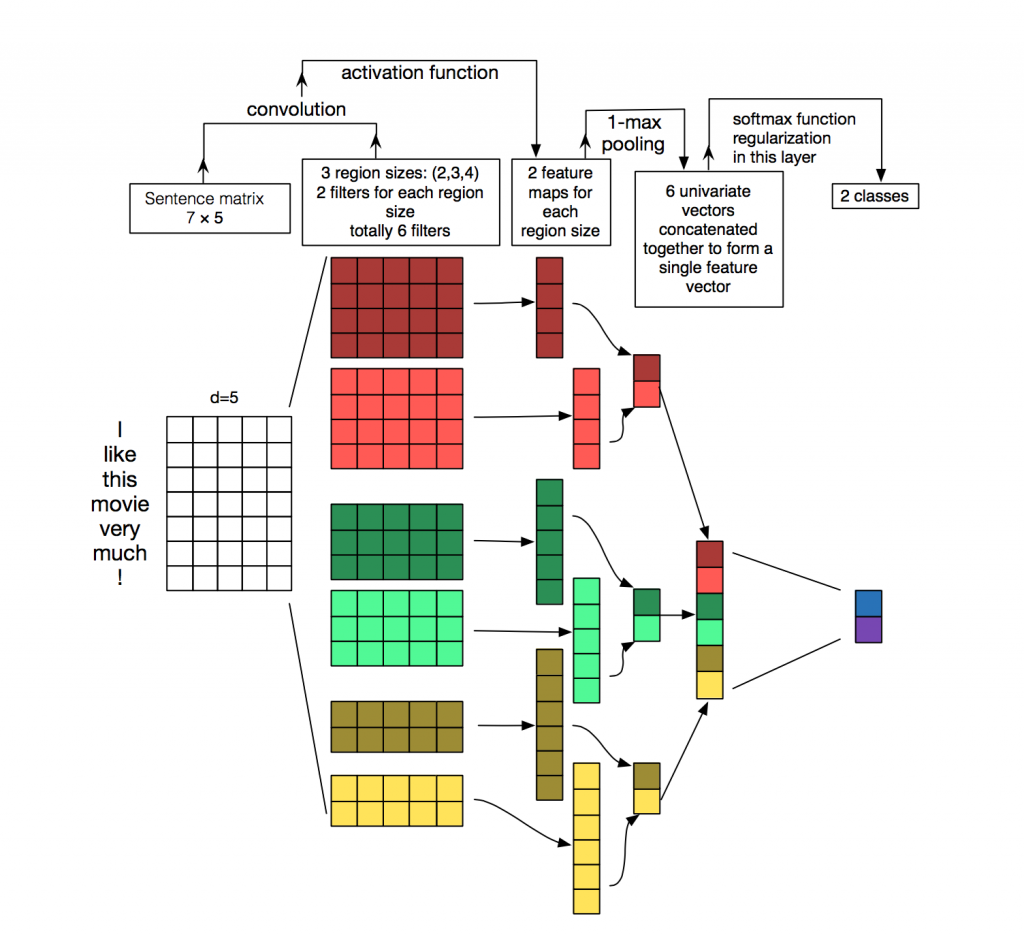


Hình 1.12 Các lớp trong mô hình CNNs

Trong mô hình CNNs có 2 khía cạnh cần quan tâm là Location Invariance (tính  bất biến) and Compositionality (tính kết hợp). Với cùng một đối tượng, nếu đối tượng này được chiếu theo các gốc độ khác nhau (translation, rotation, scaling) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể. Pooling layer sẽ cho bạn tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling). Tính kết hợp cục bộ cho ta các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn thông qua convolution từ các bộ lọc filter. Đó là lý do tại sao CNNs cho ra mô hình với độ chính xác rất cao. cũng giống như cách con người nhận biết các vật thể trong tự nhiên.

Thay vì các điểm ảnh, đầu vào cho hầu hết các tác vụ NLP là các câu hoặc tài liệu được biểu diễn dưới dạng ma trận. Mỗi hàng của ma trận tương ứng với một từ hoặc nó có thể là một ký tự. Mỗi hàng là vector đại diện cho một từ. Thông thường, những vector này là words embeddings (hay word vector) như word2vec, nhưng chúng cũng có thể là các vector với chỉ mục từ trong một bộ từ vựng (vocabulary). Đối với một câu 10 từ sử dụng 100 chiều sẽ có một ma trận 10×100 là đầu vào như một hình ảnh.

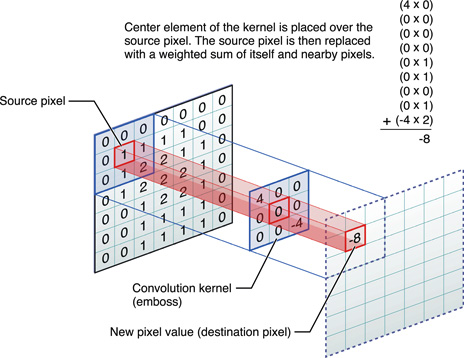
Trong thị giác máy tính, các bộ lọc trượt qua các vùng cục bộ của một hình ảnh, nhưng trong NLP thường sử dụng bộ lọc trượt qua hàng đầy đủ của ma trận từ. Do đó, "chiều rộng" của bộ lọc của thường bằng với chiều rộng của ma trận đầu vào. Chiều cao, hoặc kích thước vùng, có thể thay đổi, nhưng cửa sổ trượt trên 2-5 từ một lần là điển hình. Kết hợp tất cả những điều trên với nhau, CNNs cho NLP có thể trông như sau:



Hình 1.13 Mô hình CNNs cho phân loại văn bản

* + - 1. Các lớp trong mô hình
* **Lớp Convolution**

Lớp này chính là nơi thể hiện tư tưởng ban đầu của mạng nơ-ron tích chập. Thay vì kết nối toàn bộ điểm ảnh, lớp này sẽ sử dụng một bộ các bộ lọc filters có kích thước nhỏ so với ảnh (thường từ 2 đến 5) áp vào một vùng trong ảnh và tiến hành tính tích chập giữa bộ lọc filter và giá trị điểm ảnh trong vùng cục bộ đó. Bộ lọc filter sẽ lần lượt được dịch chuyển theo một giá trị bước trượt (stride) chạy dọc theo ảnh và quét toàn bộ ảnh.



Hình 1.14 Tính tích chập với bộ lọc filter [14]

Như vậy nếu với một bức ảnh 32×32 và một filter 3×3, ta sẽ có kết quả là một tấm ảnh mới có kích thước 32×32 (với điều kiện đã thêm padding vào ảnh gốc để tính tích chập cho các trường hợp filter quét ra các biên cạnh) là kết quả tích chập của filter và ảnh. Với bao nhiêu filter trong lớp này thì ta sẽ có bấy nhiêu ảnh tương ứng mà lớp này trả ra và được truyền vào lớp tiếp theo. Các trọng số của filter ban đầu sẽ được khởi tạo ngẫu nhiên và sẽ được học dần trong quá trình huấn luyện mô hình.

* **Lớp RELU – Rectified Linear Unit**

Lớp này thường được cài đặt ngay sau lớp Convolutional. Lớp này sử dụng hàm kích hoạt . Nói một cách đơn giản, lớp này có nhiệm vụ chuyển toàn bộ giá trị âm trong kết quả lấy từ lớp Convolutional thành giá trị 0. Ý nghĩa của cách cài đặt này chính là tạo nên tính phi tuyến cho mô hình. Tương tự như trong mạng truyền thẳng, việc xây dựng dựa trên các phép biến đổi tuyến tính sẽ khiến việc xây dựng đa tầng đa lớp trở nên vô nghĩa. Có rất nhiều cách để khiến mô hình trở nên phi tuyến như sử dụng các hàm kích hoạt sigmoid, tanh, … nhưng hàm dễ cài đặt, tính toán nhanh mà vẫn hiệu quả.

* **Lớp Pooling**

Lớp này sử dụng một cửa sổ trượt quét qua toàn bộ ảnh dữ liệu, mỗi lần trượt theo một bước trượt (stride) cho trước. Khác với lớp Convolutional, lớp Pooling không tính tích chập mà tiến hành lấy mẫu (subsampling). Khi cửa sổ trượt trên ảnh, chỉ có một giá trị được xem là giá trị đại diện cho thông tin ảnh tại vùng đó (giá trị mẫu) được giữ lại. Các phương thức lấy phổ biến trong lớp Pooling là MaxPooling ( lấy giá trị lớn nhất), MinPooling (lấy giá trị nhỏ nhất) và AveragePooling (lấy giá trị trung bình).

Lớp Pooling có vai trò giảm kích thước dữ liệu. Với một bức ảnh kích thước lớn qua nhiều lớp Pooling sẽ được thu nhỏ lại tuy nhiên vẫn giữ được những đặc trưng cần cho việc nhận dạng (thông qua cách lấy mẫu). Việc giảm kích thước dữ liệu sẽ làm giảm lượng tham số, tăng hiệu quả tính toán và góp phần kiểm soát hiện tượng quá khớp (overfitting).

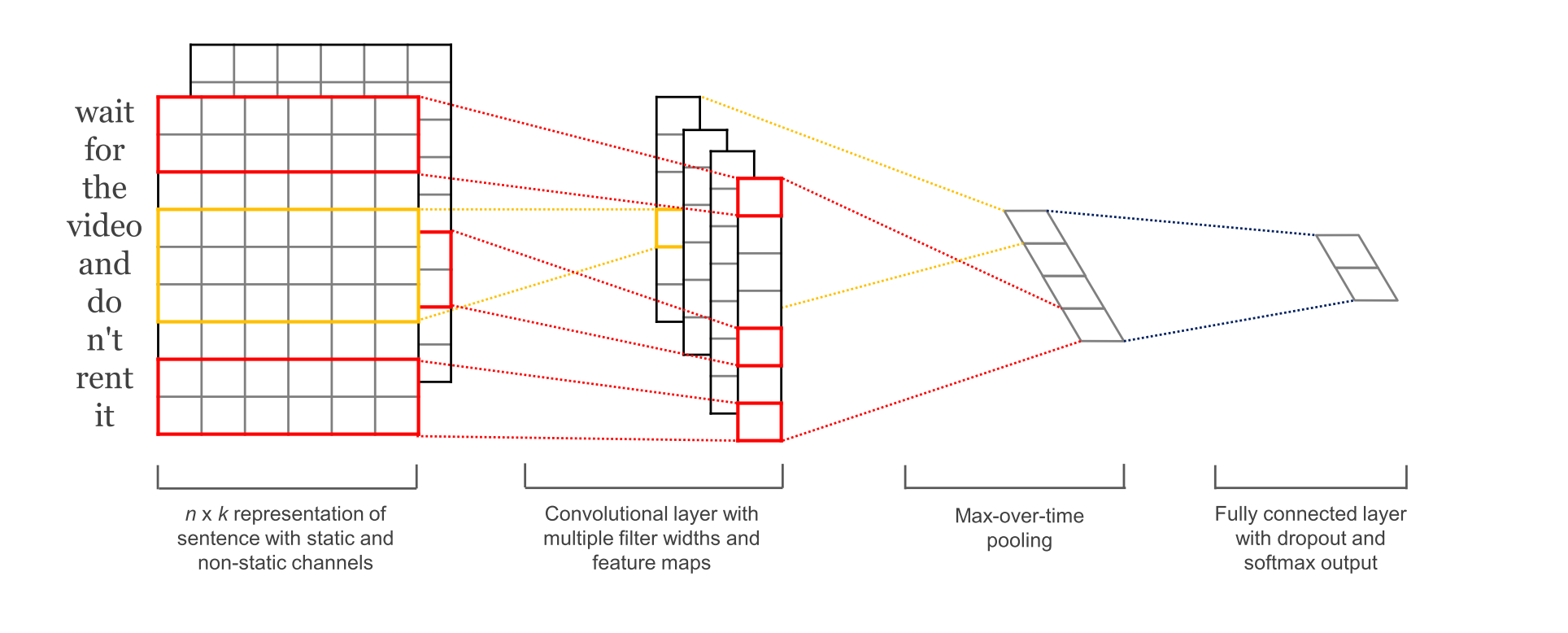
* **Lớp FC – fully connected**

Sau khi ảnh được xử lý và rút trích đặc trưng từ các lớp trước đó, dữ liệu ảnh sẽ không còn quá lớn so với mô hình truyền thẳng nên ta có thể sử dụng mô hình truyền thẳng để tiến hành nhận dạng. Tóm lại, lớp fully-connected đóng vai trò như một mô hình phân lớp và tiến hành dựa trên dữ liệu đã được xử lý ở các lớp trước đó.

* + - 1. Hoạt động của mô hình

Một mạng nơ-ron tích chập được hình thành bằng cách ghép các lớp nêu trên lại với nhau. Mô hình bắt đầu với lớp Convolutional. Lớp RELU thường luôn được cài đặc ngay sau lớp Convolutional hoặc thậm chí kết hợp cả hai lớp này thành một lớp. Các lớp tiếp theo có thể là Convolutional hay Pooling tùy theo kiến trúc mà ta muốn xây dựng. Cuối cùng sẽ là lớp fully-connected để tiến hành phân lớp.Để xem mô hình này hoạt động như thế nào ta có thể xét một kiến trúc sau đây:

Conv1 (với RELU) – Pooling – FC



Hình 1.15 Hoạt động của mô hình CNNs

Có thể coi đầu vào ở đây là một hình ảnh với kích thước AxB sẽ được đưa vào lớp Conv1 (Convolutional kết hợp RELU) gồm các cửa sổ có kích thước thường từ 2 đến 5, mỗi filter sẽ được dùng để tính tích chập với ảnh và cho ra một ảnh kết quả tương ứng. Nếu 4 filter ta sẽ có 4 ảnh kết quả như sau:

Mỗi ảnh trên đều có kích thước tương ứng là 1×B. Sau đó, cả 4 ảnh này đều được cho qua lớp Pooling và kết quả trả ra sẽ là 4 ảnh có kích thước 1x1.  
Với kích thước đủ nhỏ như vậy, lớp Fully-connected tiếp theo sẽ xử lý và đưa ra kết quả phân lớp hay kết quả nhận dạng.

Tương tự như mạng nơ-ron truyền thẳng, mạng nơ-ron tích chập cũng là một mô hình học cho nên khởi tạo ban đầu cho các trọng số trong mạng là ngẫu nhiên và sẽ được điều chỉnh thông qua quá trình học. Thuật toán học cho mạng nơ-ron tích chập cũng tương tự như mạng nơ-ron truyền thẳng là thuật toán lan truyền ngược sử dụng gradient descent; chỉ khác nhau ở chỗ mạng tích chập không liên kết đầy đủ nên độ lỗi ở mỗi lớp chỉ tính dựa vào một phần các node trong lớp tiếp theo chứ không phải toàn bộ.

* 1. Đánh giá và kết luận

CHƯƠNG 2: XÂY DỰNG HỆ THỐNG PHÂN TÍCH CẢM XÚC DỰA TRÊN NEURAL NETWORK

Trong phần này, khóa luận đề xuất một mô hình giải quyết bài toán, trình bày chi tiết hướng giải quyết của từng pha trong mô hình.

* 1. Mô hình hệ thống

Qua quá trình khảo sát dữ liệu, nghiên cứu và tham khảo các công trình khác, luận văn sẽ áp dụng mô hình phân lớp quan điểm theo các bước sau:

**Bước 1:** Thu thập dữ liệu bình luận của sản phẩm từ trang web bán hàng (lazada) sử dụng khung làm việc Scrapy.

**Bước 2:** Tiền xử lý dữ liệu thu thập được: làm sạch và chuẩn hóa dữ liệu.

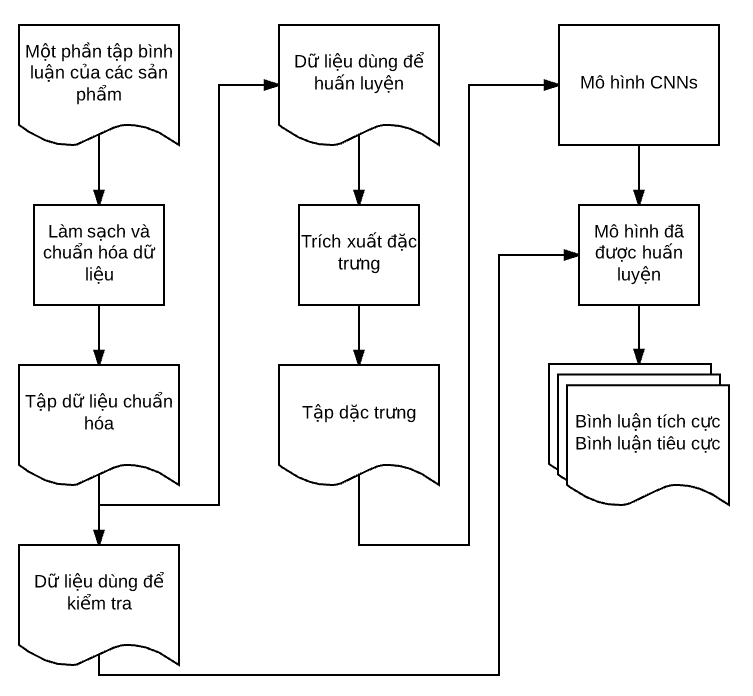
**Bước 3:** Nhận dạng thủ công từng bình luận trong bộ dữ liệu mẫu, phân vào các lớp positive (tích cực), negative (tiêu cực).

**Bước 4:** Lựa chọn, trích xuất đặc trưng và huấn luyện mô hình.

**Bước 5:** Chạy bộ phân lớp, so sánh kết quả phân lớp tự động và phân lớp thủ công.

**Bước 6:** Chạy phân loại cho toàn bộ bình luận thu thập được

Các bước trên được sơ đồ hóa với mô hình phân lớp được thể hiện trong hình:



Hình 2.1 Sơ đồ mô hình hệ thống

* + 1. Thu thập dữ liệu

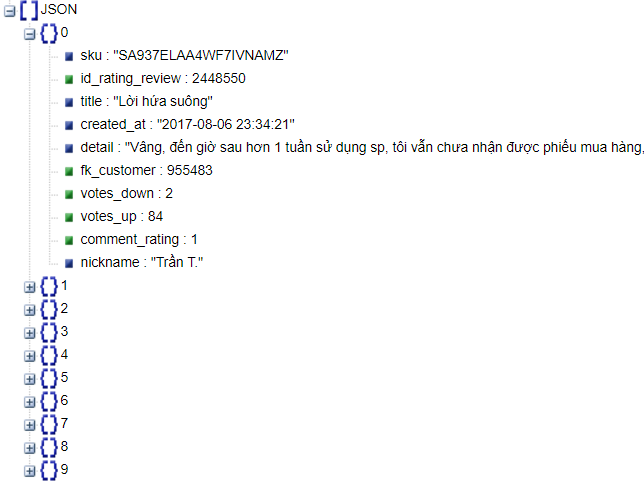
Trong bước này tôi có hai nhiệm vụ chính:

* **Nhiệm vụ 1:** Thu thập thông tin của các sản phẩm trên trang lazada, cụ thể trong nhóm sản phẩm điện tử.
  + Thu thập danh sách đường dẫn của các mục con trong nhóm sản phẩm điện tử như: điện thoại, máy tính bảng, phụ kiện điện thoại, tivi …
  + Sử dụng khung làm việc Scrapy để lấy danh sách sản phẩm trong các mục con đó

Bảng 2.1 Danh sách mục thu thập dữ liệu

|  |  |
| --- | --- |
| Danh mục sản phẩm | Các mục con |
| Điện tử | Điện thoại |
| Máy tính bảng |
| Phụ kiện điện thoại |
| Phụ kiện máy tính bảng |
| Laptop |
| Máy tính để bàn |
| Phụ kiện máy tính |
| Phụ kiện máy ảnh |
| Tivi |
| Phụ kiện tivi |
| Thiết bị âm thanh |
| Game |
| Thiết bị đeo công nghệ |
| Thiết bị đeo |

* **Nhiệm vụ 2:** Lấy bình luận của từng sản phẩm trong danh sách sản phẩm đã lấy được ở bước trên

****

Hình 2.2 Dữ liệu bình luận sản phẩm thu thập được

* + 1. Tiền xử lý dữ liệu

Vì dữ liệu bình luận là nhiễu và sai chính tả nhiều do đó việc làm sạch và chuẩn hóa là rất quan trọng. Dữ liệu chúng tôi cần là nội dung của các bình luận nên chúng tôi cần phải thực hiện cắt bỏ các dữ liệu dư thừa như: thời gian, tên người viết, tiêu đề… Thêm vào đó tập dữ liệu có hiện tượng các bình luận lặp lại làm ảnh hưởng đến độ chính xác khi đánh giá vì vậy thực hiện loại bỏ các dữ liệu dư thừa và các bình luận lặp lại.

* + 1. Trích xuất đặc trưng

**Tách từ tập dữ liệu tiếng Việt**

Tách từ là một quá trình xử lý nhằm mục đích xác định ranh giới của các từ trong câu văn, cũng có thể hiểu đơn giản rằng tách từ là quá trình xác định các từ đơn, từ ghép… có trong câu. Đối với xử lý ngôn ngữ, để có thể xác định cấu trúc ngữ pháp của câu, xác định từ loại của một từ trong câu, yêu cầu nhất thiết đặt ra là phải xác định được đâu là từ trong câu. Vấn đề này tưởng chừng đơn giản với con người nhưng đối với máy tính, đây là bài toán rất khó giải quyết.

Chính vì lý do đó tách từ được xem là bước xử lý quan trọng đối với các hệ thống Xử Lý Ngôn Ngữ Tự Nhiên, đặc biệt là đối với các ngôn ngữ thuộc vùng Đông Á theo loại hình ngôn ngữ đơn lập, ví dụ: tiếng Trung Quốc, tiếng Nhật, tiếng Thái, và tiếng Việt. Với các ngôn ngữ thuộc loại hình này, ranh giới từ không chỉ đơn giản là những khoảng trắng như trong các ngôn ngữ thuộc loại hình hòa kết như tiếng Anh…, mà có sự liên hệ chặt chẽ giữa các tiếng với nhau, một từ có thể cấu tạo bởi một hoặc nhiều tiếng. Vì vậy đối với các ngôn ngữ thuộc vùng Đông Á, vấn đề của bài toán tách từ là khử được sự nhập nhằng trong ranh giới từ.

Bởi vì các lý do trên, trước khi đưa vào mô hình huấn luyện và trả lời câu hỏi chúng tôi đã thực hiện tách từ tiếng Việt và sử dụng công cụ vnTokenizer của tác giả Lê Hồng Phương cùng các cộng sự với độ chính xác tới 95%-97% [11].

* + 1. Phân loại dữ liệu

Sau khi thu thập và tiền xử lí chúng tôi thu được tập dữ liệu có 5797 bình luận. Tập này được tách thành tập dữ liệu huấn luyện và tập dữ liệu test với thành phần nhãn của các tập dữ liệu cụ thể như sau:

Bảng 2.2 Tập dữ liệu huấn luyện

|  |  |
| --- | --- |
| Tích cực | Tiêu cực |
| 3237 | 992 |

Bảng 2.3 Tập dữ liệu kiểm tra

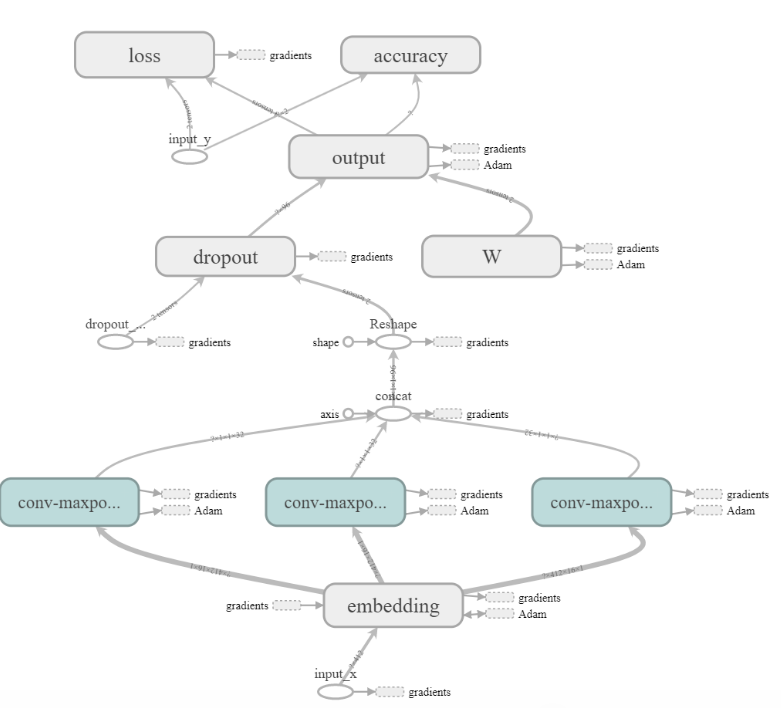
|  |  |
| --- | --- |
| Tích cực | Tiêu cực |
| 809 | 248 |

* + 1. Huấn luyện mô hình

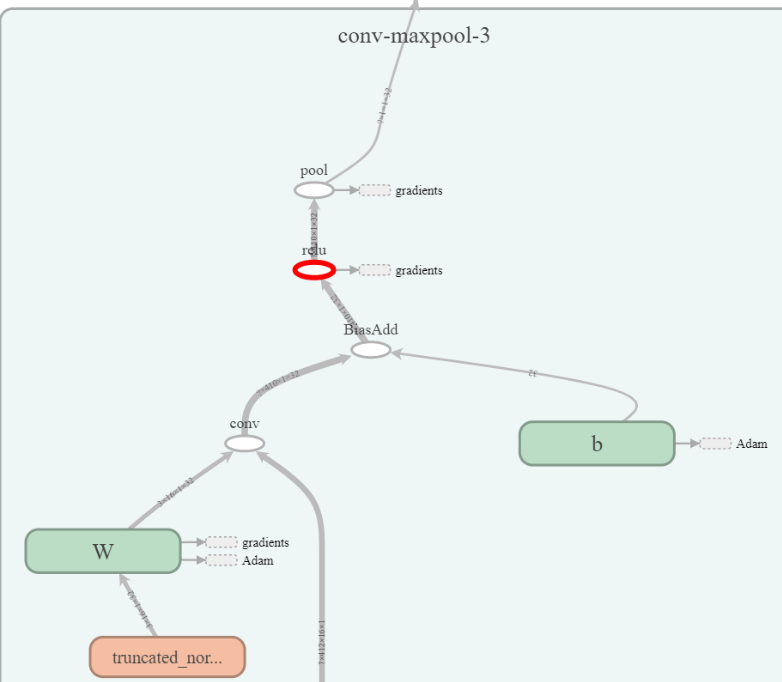
Quá trình huấn luyện sử dụng tập dữ liệu 4229 bình luận. Tập dữ liệu này lại được chia ra thành hai phần là tập dữ liệu để huấn luyện (train) và tập dữ liệu dùng để tinh chỉnh (dev) theo tỉ lệ 9:1 như sau:

Hình 2.4 Phân chia dữ liệu tập huấn luyện và tập phát triển

Mô hình sau khi cài đặt bằng tensorflow gồm các quá trình xử lý như hình sau:

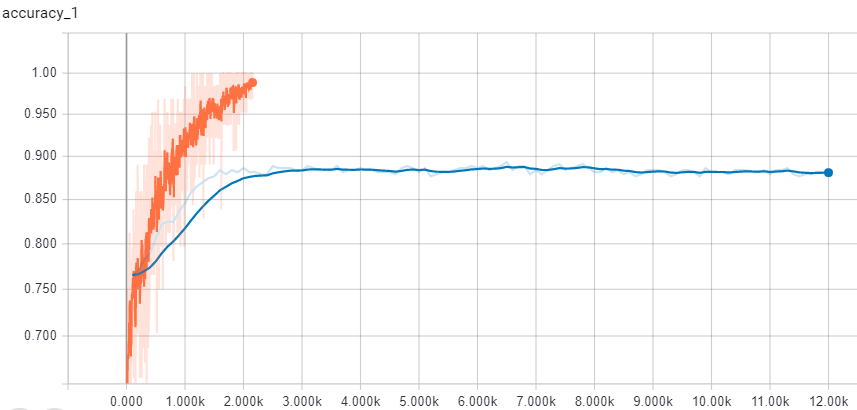


Hình 2.5 Mô hình cài đặt bằng tensorflow



Hình 2.6 Quá trình xử lý của chi tiết của lớp tích chập

Mô hình sau khi chạy huấn luyện đạt kết quả rất khả quan: 88,15%



Hình 2.7 Kết quả chạy mô hình khi huấn luyện

* 1. Kết quả phân lớp và đánh giá

Các chỉ số đo kiểm chất lượng bộ phân lớp

Mô hình được đánh giá dựa trên bộ ba tiêu chí đánh giá: độ chính xác (Precision), độ bao phủ (recall) và độ đo F1 là giá trị trung hòa giữa hai giá trị độ chính xác và độ bao phủ.

* Độ chính xác (precision)

Độ chính xác của bộ phân lớp được định nghĩa như sau:

* Độ bao phủ (recall)

Độ bao phủ của bộ phân lớp được định nghĩa như sau:

* F­1

Độ đo F1 của bộ phân lớp được định nghĩa như sau:

Đồ án tiến hành phân lớp quan điểm đối với tập dữ liệu trên Lazada gồm 5797 bình luận với 4229 bình luận dữ liệu huấn luyện và 1057 bình luận dữ liệu test. Thực nghiệm trên bộ phân lớp CNNs với các đặc trưng cũng như biểu diễn dữ liệu đã được trình bày ở trên.

Nếu chỉ căn cứ vào độ chính xác và độ bao phủ, ta không thể so sánh và đánh giá các mô hình phân lớp với nhau trong trường hợp bộ phân lớp này có độ chính xác cao, độ bao phủ thấp trong khi mô hình phân lớp khác lại có độ chính xác thấp và độ bao phủ cao.

Đồ án thực hiện phân lớp dữ liệu thành hai lớp trong bảng sau:

Bảng 2.3 Kí hiệu các lớp dữ liệu

|  |  |  |
| --- | --- | --- |
| Kí hiệu | Tên lớp tiếng Anh | Tên lớp tiếng Việt |
| 1 | Positive | Tích cực |
| 0 | Negative | Tiêu cực |

Kết quả phân lớp trên bộ dữ liệu kiểm tra với mô hình như sau:

Bảng 2.4 Kết quả thực nghiệm phân lớp sử dụng CNNs

Chính xác/Accurancy: 85,24%

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Lớp | Số bình luận | Độ chính xác(%) | Độ bao phủ(%) | F1(%) |
| Positive | 809 | 91,49 | 89 | 90,23 |
| Negative | 248 | 67,04 | 72,98 | 69,88 |
| All | 1057 | 85,24 | 85,24 | 85,24 |

Từ bảng kết quả trên chúng ta có thể thấy, kết quả của bộ phân lớp tính theo tiêu chí độ chính xác của các nhãn positive, negative lần lượt là 91,49%, 67,04% . Các giá trị này xấp xỉ với kết quả tính theo độ bao phủ, lần lượt là 89%, 72,98%. Có thể thấy độ chính xác của các nhãn negative thấp hơn các nhãn positive là do lượng dữ liệu huấn luyện của negative ít hơn hẳn nên độ chính xác khi kiểm tra cũng thấp hơn. Điều này cho thấy, bộ phân lớp tương đối ổn định khi đánh giá theo hai tiêu chí trên, kết quả là giá trị F1 theo từng nhãn cũng xấp xỉ nhau. Kết quả tính theo tiêu chí F1 đạt 85,24% nếu tính theo tổng toàn bộ nhãn của chương trình.

CHƯƠNG 3: CÀI ĐẶT HỆ THỐNG

* 1. Mô hình cài đặt

Hình 3.1 Sơ đồ mô hình cài đặt hệ thống

* + 1. Scrapy

Scrapy là một framework được viết bằng Python, cung cấp sẵn 1 cấu trúc tương đối hoàn chỉnh để thực hiện việc thu thập (crawl) và trích xuất dữ liệu (data extract) từ website một cách nhanh chóng và dễ dàng. Nếu muốn lấy dữ liệu từ các website nhưng dữ liệu đó quá lớn để sao chép rồi dán vào cơ sở dữ liệu, scrapy hỗ trợ ta làm điều đó. Việc lấy dữ liệu website hoàn toàn tự động nhanh chóng và việc sử dụng scrapy cũng rất đơn giản giúp tiếp kiệm được nhiều thời gian và công sức.

* + 1. SQLite

SQLite là phần mềm quản lý cơ sở dữ liệu tương tự như Mysql, PostgreSQL... Đặc điểm của SQLite là gọn, nhẹ, đơn giản. Chương trình gồm 1 file duy nhất vỏn vẹn chưa đến 400kB, không cần cài đặt, không cần cấu hình hay khởi động mà có thể sử dụng ngay. Dữ liệu Database cũng được lưu ở một file duy nhất. Không có khái niệm user, password hay quyền hạn trong SQLite Database.

SQLite không thích hợp với những hệ thống lớn nhưng ở quy mô vừa tầm thì SQLite phát huy uy lực và không hề yếu kém về mặt chức năng hay tốc độ. Với các đặc điểm trên SQLite được sử dụng nhiều trong việc phát triển, thử nghiệm …

* + 1. Khung làm việc Tensorflow

Tensorflow là một thư viện phần mềm nguồn mở cho tính toán số sử dụng biểu đồ luồng dữ liệu. TensorFlow ban đầu được phát triển bởi các nhà nghiên cứu và kỹ sư làm việc trong nhóm Brain Google trong tổ chức nghiên cứu máy tình báo của Google nhằm mục đích tiến hành học máy và sâu nghiên cứu các mạng nơ-ron thần kinh, nhưng hệ thống là đủ nói chung có thể áp dụng trong một loạt các lĩnh vực khác như tốt.

Tensorflow là một hệ thống học máy hoạt động ở quy mô lớn và trong môi trường phức tạp. TensorFlow [46, 47] sử dụng đồ thị luồng dữ liệu Dataflow để đại diện cho sự tính toán, chia sẻ trạng thái, và các hoạt động biến đổi trạng thái đó. Nó ánh xạ các nút của một đồ thị dataflow trên nhiều máy trong một cluster, và bên trong một máy trên nhiều thiết bị tính toán, bao gồm CPU, GPU đa lõi, các chíp ASIC tùy biến được gọi là tenxơ Processing Units (TPUs). Kiến trúc này rất linh hoạt cho phép cho các nhà phát triển ứng dụng: trong khi trước đây "tham số máy chủ" thiết kế quản lý chia sẻ trạng thái (shared state) được xây dựng sẵn trên hệ thống, TensorFlow cho phép các nhà phát triển để thử nghiệm các tối ưu hoá mới và các thuật toán huấn luyện.

TensorFlow hỗ trợ một loạt các ứng dụng, với sự hỗ trợ đặc biệt mạnh mẽ cho việc huấn luyện và suy luận trên các mạng học sâu Deep Learning [47]. Google đã phát hành TensorFlow như là một dự án mã nguồn mở, và nó đã trở thành sử dụng rộng rãi cho các nghiên cứu học máy.

Trong bài luận văn này, chúng tôi sử dụng TensorFlow để huấn luyện và tạo ra các mô hình đối thoại cho tiếng Việt, một kết quả rất khả quan khi sử dụngTensorFlow là chúng tôi đạt được những mô hình có chất lượng tốt.

* + 1. Django

Django là một web framework miễn phí mã nguồn mở được viết bằng Python. Django sử dụng mô hình Model-View-Control (MVC). Django được phát triển bởi Django Software Foundation(DSF) – một tổ chức phi lợi nhuận độc lập.

Django đơn giản hóa việc tạo các website phức tạp có sử dụng cơ sở dữ liệu. Django tập trung vào tính năng “có thể tái sử dụng” và “có thể tự chạy” của các component, tính năng phát triển nhanh, không làm lại những gì đã làm.

* 1. Cài đặt hệ thống
     1. Môi trường cài đặt

Bảng 3.1 Cấu hình phần cứng hệ thống:

|  |  |
| --- | --- |
| **Thành phần** | **Chỉ số** |
| Bộ vi xử lý | Intel Core i5 (2.6GHz) |
| RAM | 4 GB |
| Hệ điều hành | Windows 10 Education 64bit |
| Bộ nhớ | 232 GB |

Bảng 3.2 Danh sách các công cụ phần mềm:

|  |  |  |
| --- | --- | --- |
| STT | Tên công cụ | Phiên bản |
| 1 | Python | 2.7.14 và 3.5.3 |
| 2 | Scrapy | 1.4.0 |
| 3 | Sqlite | 3 |
| 4 | Tensorflow | 1.3.0 |
| 5 | Django | 1.7 |

* + 1. Cài đặt cụ thể

Chúng tôi sử dụng 2 phiên bản python khác nhau là python 2.7.14 cho Scrapy, Django và python 3.5.3 cho Tensorflow. Vì vậy chúng tôi dùng phần mềm Virtualenv để tạo môi trường ảo để tách biệt các môi trường project khác nhau giúp việc quản lý, cài đặt dễ dàng hơn. Chi tiết cài đặt như sau:

Tải 2 phiên bản python về tại đường dẫn :

<https://www.python.org/downloads/>

Sau khi cài đặt xong python tại các vị trí thu mục C:\python27, C:\python35, tiếp tục cài Virtualenv bằng câu lệnh:

pip install virtualenv

Tạo môi trường ảo, cài Scrapy và Django:

virtualenv --python=C:\Python27\python.exe env27

env27\Scripts\activate

pip install scrapy

pip install Django

deactivate

Tạo môi trường ảo vcài Tensorflow:

virtualenv --python=C:\Python27\python.exe env35

env35\Scripts\activate

pip install tensorflow

* 1. Kết quả và đánh giá

CHƯƠNG 4: ĐÁNH GIÁ

* 1. Hoạt động của hệ thống

Sử dụng Django chạy server trên localhost bằng câu lệnh sau:

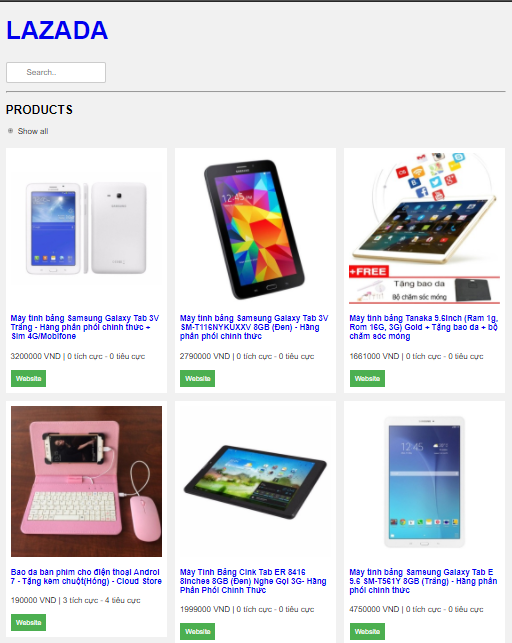
python manage.py runserver

Server sẽ chạy tại địa chỉ: <http://127.0.0.1:8000/>

Truy cập theo đường dẫn sau để truy cập trang web:

<http://127.0.0.1:8000/lazada>

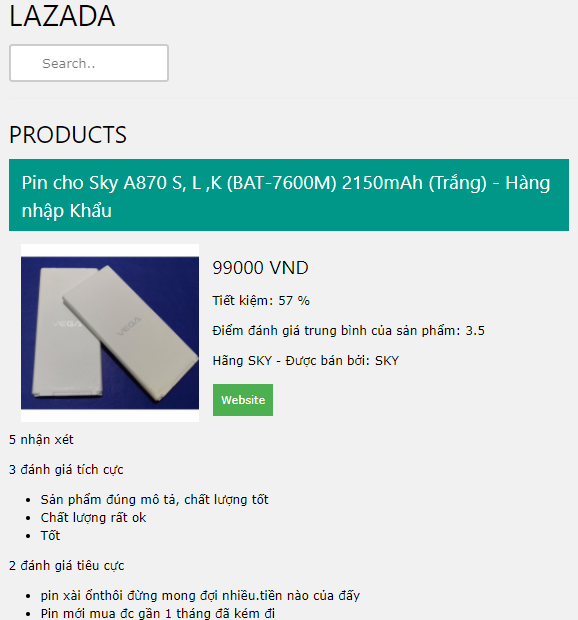
Giao diện trang chủ:



Hình 4.1 Giao diện hiển thị của trang web đánh giá

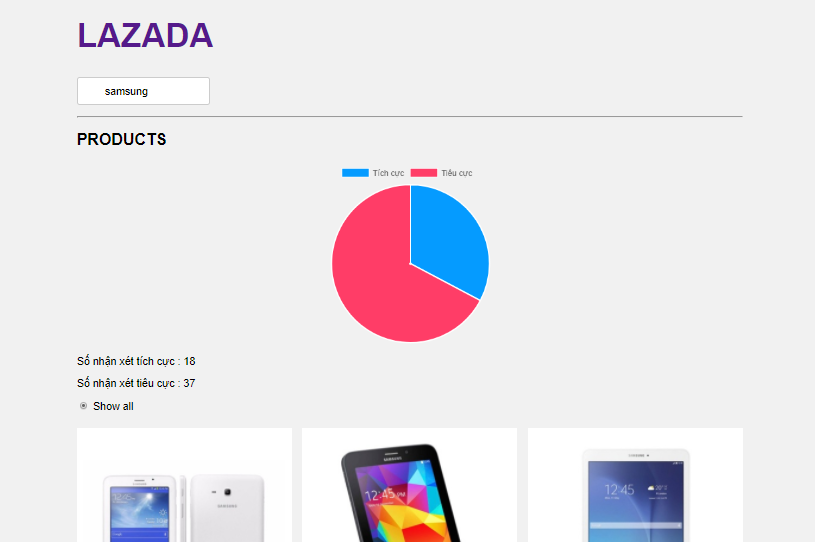
Trang web chúng tôi xây dựng có các tính năng sau:

* Tại trang chủ: hiển thị danh sách sản phẩm với các thông tin: tên sản phẩm, giá sản phẩm khi đã giảm giá, thông tin đánh giá sản phẩm tích cực hay tiêu cực
* Truy cập sản phẩm trên trang web chính thức
* Bấm vào để xem chi tiết sản phẩm với các thông tin như: hãng sản xuất, cửa hàng bán, giá, giảm giá bao nhiêu, có những bình luận tích cực, tiêu cực nào.



Hình 4.2 Thông tin cụ thể của sản phẩm

* Tìm kiếm sản phẩm, cửa hàng, hãng sản xuất để xem các thông tin sản phẩm và đánh giá trực quan.



Hình 4.3 Kết quả tìm kiếm sản phẩm với đánh giá theo tỉ lệ phần trăm

* 1. Đánh giá hiệu quả
  2. Kết luận

KẾT LUẬN

TÀI LIỆU THAM KHẢO

1. Rădulescu, C., Dinsoreanu, M. and Potolea, R. (2014), “Identification of Spam Comments Using Natural Language Processing Techniques”, Intelligent Computer Communication and Processing (ICCP), pp. 29-35.

2. Xia, R., Xu, F., Zong, C., Li, Q., Qi, Y. and Li, T. (2015), “Dual Sentiment Analysis: Considering Two Sides of One Review.”, IEEE transactions on knowledge and data engineering, 27(8), pp.2120-2133.

3. Devi, G.D. and Rasheed, A.A. (2015), “A Survey on Sentiment Analysis and

Opinion Mining.”, International journal for research in emerging science and technology 2(8), pp. 26-31.

4. Hu, M. and Liu, B. (2004), “Mining and Summarizing Customer Reviews.”, Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 168-177.

5. Liu, B. (2012), Sentiment Analysis and Opinion Mining, Morgan & Claypool.

6. Ben Krose and Patrick van der Smagt, An Introduction to Neural Networks, @ 1996 University of Amsterdam.

7. Medhat, W., Hassan, A. and Korashy, H. (2014), “Sentiment Analysis Algorithms and Applications: A Survey.”, Ain Shams Engineering Journal 5(4), pp. 1093-1113.

8. Robert J. Schallkoff – Artificial Neural Networks, the McGraw – Hill

Companies, Inc 1997.

9. W. Gerrod Parrot. “Emotions In Social Psychology”, 2001

10. Tsytsarau, M. and Palpanas, T., “Survey on Mining Subjective Data on the Web, Data Mining and Knowledge Discovery”, 2012, pp.478-514.

11. Le-Hong, P., T M H. Nguyen, A. Roussanaly, and T V. Ho (2008), “A hybrid approach to word segmentation of Vietnamese texts”, pp. 240-249

12. Severyn, A. and Moschitti, A. (2015), “Twitter Sentiment Analysis with Deep Convolutional Neural Networks.”, Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 959-962.

13. Tang, D., Qin, B. and Liu, T. (2015), “Deep Learning for Sentiment

Analysis: Successful Approaches and Future Challenges.”, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 5(6), pp. 292-303.

14. Zhang, Y., & Wallace, B. (2015). A Sensitivity Analysis of (and Practitioners’ Guide to) Convolutional Neural Networks for Sentence Classification.

15. Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification

16. Zhang, X., Li, S., Zhou, G. and Zhao, H. (2011), “Polarity Shifting: Corpus

Construction and Analysis.”, Asian Language Processing (IALP), pp. 272-275.

17. Singh, Pravesh Kumar, and Mohd Shahid Husain. "METHODOLOGICAL STUDY OF OPINION MINING AND SENTIMENT ANALYSIS TECHNIQUES."International Journal on Soft Computing 5.1, 2014.]