TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

ĐỒ ÁN TỐT NGHIỆP

Ứng dụng học chuyển tiếp trong bài toán phân tích khía cạnh cảm xúc tiếng việt

NGUYỄN THỊ HẰNG

hang.nt161384@sis.hust.edu.vn

Ngành Toán Tin

Giảng viên hướng dẫn: **ThS. NGUYỄN DANH TÚ** Chữ kí của GVHD

Bộ môn: **Toán ứng dụng**

Viện: **Toán ứng dụng và Tin học**

HÀ NỘI–2021

ĐỒ ÁN TỐT NGHIỆP

Chuyên ngành: Toán Tin

Chuyên sâu:

Ứng dụng học chuyển tiếp trong bài toán phân tích khía cạnh - cảm xúc tiếng việt

Giảng viên hướng dẫn: **ThS. NGUYỄN DANH TÚ**

Họ và tên sinh viên: **NGUYỄN THỊ HẰNG**

Số hiệu sinh viên: **20161384**

Lớp: **TOÁN TIN K61**

HÀ NỘI–2021

Nhận xét của giảng viên hướng dẫn

1. Mục tiêu và nội dung của đồ án

(a) Mục tiêu: Đề tài nghiên cứu về mô hình học chuyển tiếp trong bài toán phân tích khía cạnh, cảm xúc trong tiếng Việt. Cải tiến xử lý bài toán đa nhãn sang bài toán đa lớp đơn giản hơn.

(b) Nội dung: Giới thiệu về bài toán phân tích khía cạnh, cảm xúc, cơ sở lý thuyết về mô hình học chuyển tiếp. Trình bày kết quả nghiên cứu của ứng dụng mô hình học chuyển tiếp và ý tưởng cải tiến bài toán đa nhãn trong bài toán phân tích khía cạnh, cảm xúc.

2. Kết quả đạt được

(a) Trình bày được cơ sở lý thuyết cốt lõi của mô hình học chuyển tiếp.

(b) Giải quyết được vấn đề chọn ngưỡng ngẫu nhiên trong bài toán đa nhãn.

3. Ý thức làm việc của sinh viên:

(a)

(b)

(c)

Hà Nội, ngày 02 tháng 07 năm 2021

Giảng viên hướng dẫn

ThS. NGUYỄN DANH TÚ

Lời cảm ơn

Tác giả đã có những năm tháng không thể nào quên khi được học tập và rèn luyện dưới mái trường Đại học Bách Khoa Hà Nội với sự chỉ dạy của các thầy cô Viện Toán Ứng dụng và Tin học. Thời gian trôi qua nhanh như một cơn gió, từ khi còn là cô nhóc học lớp 12 với khát khao được bước chân vào cánh cổng Đại Học Bách Khoa vì đơn giản người ta nói học Bách Khoa rất ngầu. Chưa từng có định hướng sẽ theo học ngành gì vì chẳng biết đam mê của mình là gì, rồi một cơ duyên đưa tác giả với viện toán nơi tác giả nhìn thấy những con người nhiệt huyết tài năng và là nơi tác giả nhận được sự giúp đỡ, dìu dắt nhiệt tình của các thầy cô. Mặc dù quãng thời gian này không quá dài nhưng nó sẽ luôn luôn là quãng thời gian tuyệt vời trong cuộc đời tác giả.

Lời đầu tiên tác giả muốn gửi lời cảm ơn sâu sắc nhất tới **Th.s Nguyễn Danh Tú** người đã trực tiếp hướng dẫn tác giả, người đã đưa ra những nhận xét lời khuyên sát sao để tác giả có thể hoàn thành đồ án này. Cũng muốn cảm ơn thầy vì đã cho tác giả thấy được khuyết điểm của bản thân để không ngừng cải tiến và nỗ lực. Giúp tác giả học được cách nhìn vấn đề một cách tổng quan hơn và có chiều sâu hơn.

Tác giả cũng trân trọng những ý kiến đóng góp, nhận xét chi tiết của **Ts. Lê Chí Ngọc**, với những kiến thức sâu sắc trong lĩnh vực học máy và trí tuệ nhân tạo thầy đã giúp tác giả rất nhiều cải thiện những thiếu sót về mặt kiến thức của tác giả. Tác giả cũng gửi lời cảm ơn chân thành đến thầy người đã giúp tác giả tìm được hướng đi đúng trên con đường nghiên cứu và công việc sau này. Ngoài ra tác giả cũng gửi lời cảm ơn đến anh **Nguyễn Hồng Sơn** là cựu sinh viên viện toán ứng dụng đã giúp đỡ tác giả rất nhiều và chỉ ra những sai sót nhầm lẫn của tác giả trong quá trình hoàn thiện đồ án.

Tác giả cũng xin gửi lời cảm ơn đến tất cả bạn bè và thầy cô trong viện toán ứng dụng và tin học đã đồng hành cùng tác giả trong suốt quãng thời gian sinh viên tươi đẹp này. Đặc biệt tác giả muốn gửi lời biết ơn đến bố **Nguyễn Văn**

Vịnh và mẹ **Phạm Thị Hoa** người đã hy sinh rất nhiều để tác giả có thể vững bước trên con đường trưởng thành của mình.

Tác giả trân trọng cảm ơn các thầy cô đã đọc, đánh giá và đưa ra nhận xét đối với đồ án này. Ngoài ra, còn rất nhiều người nữa đã luôn ủng hộ giúp đỡ tác giả, nhưng thật khó để nêu tên hết mọi người ở đây, tác giả xin gửi lời cảm ơn sâu sắc đến tất cả mọi người và luôn ghi nhớ mọi người trong tim của mình.

Hà Nội, ngày 02 tháng 07 năm 2021

Tác giả đồ án

Nguyễn Thị Hằng

Tóm tắt nội dung đồ án

Phân tích cảm xúc trong văn bản là một lĩnh vực nhỏ trong xử lý ngôn ngữ tự nhiên có tính ứng dụng và đem lại giá trị thực tế cao. Có nhiều phương pháp tiếp cận và giải quyết bài toán này như trong nghiên cứu trước của tác giả, tác giả đã sử dụng một số thuật toán máy học cơ bản như mô hình naive bayes, Support Vector Machine (SVM) hay một số mô hình học sâu Long Short Term Memory (LSTM), Convolution Neural Network kết hợp LTM (CNN+LSTM) để giải quyết bài toán. Mặc dù kết quả nghiên cứu thu được tương đối tốt nhưng vẫn có những hạn chế về mặt dữ liệu (dữ liệu nhãn nhị phân) trong khi đó thực tế dữ liệu phức tạp và nhiều ý nghĩa hơn nhiều. Ví dụ như về mặt nội dung dữ liệu có thể nhắc đến những thực thể, thuộc tính đặc thù, về mặt ý nghĩa dữ liệu có thể là nhận xét tích cực, tiêu cực nhưng cũng có thể chỉ là nhận xét chung chung không mang bất kỳ màu sắc tình cảm nào. Ngoài ra còn về mặt mô hình như mạng LSTM đã có thể hiểu được ngữ nghĩa của văn bản nhưng không thực sự tốt, trong khi đó các mô hình học chuyển tiếp như BERT, RoBERTa, AlBERT, DisTilBERT đã cho thấy kết quả tuyệt vời trong các tác vụ xử lý ngôn ngữ nói chung và bài toán phân loại văn bản nói riêng.

Để khắc phục một số hạn chế trong nghiên cứu trước đây của mình, tác giả quyết định mở rộng bài toán nhằm khai thác tối đa tri thức mà dữ liệu đem lại. Trong nội dung của đồ án, tác giả tập trung vào bài toán đa nhãn gồm hai bài toán con là phân tích khía cạnh và phân tích cảm xúc. Việc xác định các khía cạnh được nhắc đến trong câu bình luận sẽ tùy thuộc vào từng lĩnh vực sẽ có bộ khía cạnh đặc thù và phân tích cảm xúc mà các khía cạnh thể hiện như tích cực, tiêu cực hay trung tính. Bộ dữ liệu tác giả sử dụng là hai bộ dữ liệu về nhà hàng và khách sạn được cung cấp bởi VLSP2018. Tác giả đã thực hiện nhiều thử nghiệm để giải quyết bài toán đặt ra. Song song với đó, tác giả cũng đã tìm hiểu, nghiên cứu các công trình có liên quan và đưa ra những nhận định về hướng giải quyết, ưu và nhược điểm của các phương pháp để xây dựng

hướng giải quyết và cải tiến. Tác giả tập trung cải thiện độ chính xác của mô hình BERT và áp dụng ý tưởng của bài toán hỏi đáp (QnA) để giải quyết vấn đề đa nhãn và tối ưu ngưỡng của các nghiên cứu đã được công bố trước đây trên cùng bộ dữ liệu này. Các kết quả thử nghiệm cho thấy kết quả tuyệt vời của mô hình BERT cũng như ý tưởng mới của tác giả trong tác vụ phân loại cảm xúc.

Từ khóa: Aspect base, sentiment analysis, BERT.

Hà Nội, ngày 02 tháng 07 năm 2021

Tác giả đồ án

Nguyễn Thị Hằng

Mục lục

Bảng ký hiệu và các từ viết tắt

Từ viết tắt	Ý nghĩa
RV	Review, đánh giá, bình luận
NLP	Natural Language Processing, xử lý ngôn ngữ tự nhiên
AE	Autoencoder, Tự mã hóa
CNN	Convolution neural network, Mạng neuron tích chập
RNN	Recurrent neural network, Mạng neuron hồi tiếp
LSTM	Long short term memory, Mạng bộ nhớ ngắn dài
Seq2Seq	Sequence to sequence, Mô hình chuỗi sang chuỗi
TL	Transfer Learning, học chuyển tiếp
BERT	Bidirectional Encoder Representations from Transformers, Mô hình biểu diễn ngôn ngữ hai chiều
Masked LM	Masked-Language Modeling
NSP	Next Sentence Prediction, nhiệm vụ dự đoán câu tiếp theo
SA	Sentiment analysis
ApB	Aspect base
QnA	Question Answering, Hỏi đáp

Danh sách bảng

Danh sách hình vẽ

Mở đầu

Thu thập và phân tích thông tin phản hồi của khách hàng là một cách giúp cho các doanh nghiệp có được những thông tin giá trị như hiểu được điểm mạnh, điểm yếu trong sản phẩm, dịch vụ của mình để có thể cải thiện chất lượng sản phẩm, dịch vụ đồng thời nhanh chóng nắm bắt được tâm lý và nhu cầu khách hàng nhằm phát triển sản phẩm phù hợp thị hiếu, mang đến cho khách hàng sản phẩm, dịch vụ hoàn hảo nhất.

Ngày nay, với sự phát triển vượt bậc của khoa học và công nghệ, đặc biệt là sự bùng nổ của Internet với các phương tiện truyền thông, mạng xã hội, thương mại điện tử,... việc chia sẻ thông tin, thể hiện thái độ, quan điểm của mình đối với các sản phẩm, dịch vụ và các vấn đề xã hội khác ngày càng trở nên dễ dàng, phong phú dưới nhiều hình thức khác nhau. Vì vậy Internet trở thành nguồn cung cấp một lượng thông tin vô cùng lớn và quan trọng.

Thông qua những dữ liệu được cung cấp qua Internet:

- Người dùng sử dụng nó để tìm kiếm, tham khảo trước khi đưa ra quyết định về sử dụng một sản phẩm hay dịch vụ nào đó.
- Các nhà cung cấp dịch vụ cũng có thể sử dụng những nguồn thông tin này để đánh giá về sản phẩm của mình, từ đó có thể đưa ra những cải tiến phù hợp hơn với người dùng, mang lại lợi nhuận cao hơn, tránh các rủi ro đáng tiếc xảy ra. Đặc biệt, khi một doanh nghiệp có một sản phẩm mới ra mắt thị trường thì việc lấy ý kiến phản hồi là vô cùng cần thiết.
- Các cơ quan chức năng có thể sử dụng những thông tin này để tìm

hiểu xem quan điểm và thái độ của cộng đồng để có thể kịp thời sửa đổi, ban hành các chính sách cho hợp lý hơn.

Đứng trước sự phát triển mạnh mẽ của công nghệ, thương mại điện tử việc phân tích, bóc tách dữ liệu là việc cần thiết và vô cùng quan trọng. Vì vậy tác giả quyết định lựa chọn đề tài **Ứng dụng học chuyển tiếp trong bài toán phân tích khía cạnh - cảm xúc Tiếng Việt"** để tìm hiểu, nghiên cứu với mong muốn phần nào áp dụng bài toán vào thực tế. Trong khuôn khổ của đồ án tác giả sẽ tập trung vào đánh giá khả năng của phương pháp học chuyển tiếp áp dụng cho bài toán phân loại cảm xúc, từ đó đưa ra những kết luận về mức độ hiệu quả cũng như mặt hạn chế của những mô hình học chuyển tiếp.

Đối tượng và phạm vi nghiên cứu của đề tài này bao gồm:

- Đối tượng nghiên cứu: Bài toán phân tích khía cạnh, cảm xúc trong tiếng việt sử dụng mô hình học chuyển tiếp.
- Tập dữ liệu: Tập dữ liệu đánh giá người dùng về khách sạn, nhà hàng được cung cấp bởi VLSP (tập dữ liệu năm 2018).
- Phương pháp nghiên cứu: Thực hiện các thực nghiệm cần thiết.

Cấu trúc đồ án sẽ bao gồm:

Chương 1: Tổng quan bài toán và cơ sở lý thuyết

Giới thiệu tổng quan về bài toán phân tích khía cạnh, cảm xúc trong tiếng việt. Trình bày một số cơ sở lý thuyết và một số hướng tiếp cận được tác giả xem xét từ đó đưa ra kết luận về ưu, nhược điểm của phương pháp.

Chương 2: Học chuyển tiếp trong xử lý ngôn ngữ tự nhiên

Trong chương này tác giả sẽ trình bày chi tiết về mô hình học chuyển tiếp, trong đó có mạng biểu diễn hai chiều tiền huấn luyện cho mô hình ngôn ngữ (BERT) và các kiến thức nền tảng liên quan như cơ chế chú ý (attention) và mô hình transformers. Trình bày chi tiết cách thức triển

khai và áp dụng mô hình học chuyển tiếp cho bài toán phân tích khía cạnh, cảm xúc và một số phương pháp đánh giá hiệu quả của mô hình.

Chương 3: Kết luận

Trong chương ba này tác giả sẽ trình bày các kết quả thử nghiệm và những đề xuất phương hướng cải tiến cho bài toán.

Đồ án được hoàn thành trong chương trình Toán Tin tại Viện toán ứng dụng và Tin học, Đại học Bách Khoa Hà Nội dưới sự hướng dẫn của ThS. Nguyễn Danh Tú. Mặc dù đã cố gắng nhưng do hạn chế về mặt thời gian và kinh nghiệm, đồ án này không thể tránh khỏi những sai sót. Tác giả hy vọng nhận được ý kiến đóng góp quý báu từ thầy cô và các bạn để đồ án được hoàn thiện tốt hơn.

Chương 1

Tổng quan bài toán và cơ sở lý thuyết

1.1 Tổng quan bài toán

Xử lý ngôn ngữ tự nhiên - Natural Language Processing (NLP): Được xây dựng dựa trên ngôn ngữ học phức tạp, các nguyên lý thống kê, và thuật toán mạng neuron (neural network algorithms). Chương trình NLP có khả năng đọc và hiểu được văn bản với tốc độ cao. Do đó, dù bạn có 1000 tài liệu hay thậm chí hàng tỉ văn bản, chương trình NLP có thể "tiêu hoá" nhanh chóng tất cả các thông tin này, từ đó có thể rút trích ra được những tri thức (knowledge) đáng giá cho doanh nghiệp của bạn như: tri thức về các khách hàng, tri thức về những đối thủ cạnh tranh, tri thức về các hoạt động trong doanh nghiệp như điều hành, marketings, sales, kỹ thuật, và sản phẩm.

Bài toán xử lý ngôn ngữ tự nhiên đang được quan tâm:

- Bài toán mức độ đơn giản:
  - Kiểm tra chính tả (Spell checking).
  - Tìm kiếm từ khóa (keyword search).
  - Tìm từ đồng nghĩa (Finding synonyms).
- Bài toán mức độ trung bình:

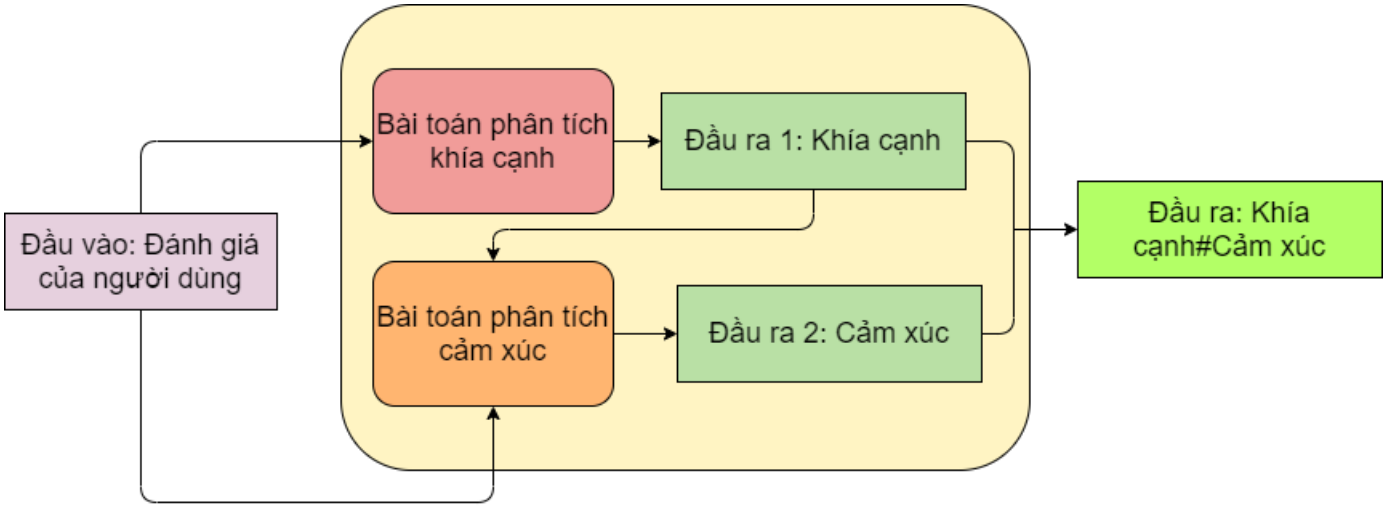
  - Phân tích thông tin từ các tài liệu, websites, ... (Parsing information from websites, documents, etc.).

- Bài toán mức độ khó:
  - Dịch máy (Machine translation).
  - Phân tích ngữ nghĩa (Semantic analysis).
  - Coreference resolution.
  - Trả lời câu hỏi, chatbot, ... (Questions answering).

Hiện nay trong thời đại công nghệ số, các nhà hàng, khách sạn hầu hết đều có những trang web để quảng bá thu hút người tiêu dùng, giúp người dùng có thể đưa ra các nhận xét và đánh giá bày tỏ quan điểm mức độ hài lòng đối với mỗi sản phẩm dịch vụ mà khách hàng được trải nghiệm. Đứng trên phương diện của doanh nghiệp thì việc có được nguồn dữ liệu này có thể giúp doanh nghiệp phân tích để hiểu hơn thị hiếu người dùng, hiểu được điểm chưa tốt mà khách hàng không hài lòng để cải thiện nâng cao chất lượng, hiểu được điểm tốt của mình để tiếp tục duy trì và phát huy. Đứng trên phương diện người dùng, khi muốn đặt phòng khách sạn hay đến một nhà hàng nào đó thì thường sẽ có những thói quen đọc các bình luận của người đã từng dùng để xem xét về nó. Với một số lượng ít vài chục hay vài trăm bình luận thì việc xem xét bằng mắt vẫn hoàn toàn khả thi nhưng với con số lớn hơn thì việc đọc từng câu một rồi thống kê là vô cùng mất thời gian, tốn kém nguồn lực. Chính vì vậy một ý tưởng đặt ra liệu có cách nào tự động thống kê những câu bình luận này hay không? Trong báo cáo này tác giả sẽ giải quyết hai bài toán là phân tích khía cạnh và phân tích cảm xúc trong khuôn khổ phạm vi về lĩnh vực kinh doanh nhà hàng, khách sạn.

Dữ liệu được sử dụng ở đây là hai bộ dữ liệu về nhà hàng và khách sạn được cung cấp bởi VLSP. Đây là nhưng dữ liệu được thu thập từ thực tế và được gán nhãn.

Mô hình bài toán như sau:



Hình 1.1: Tổng quan bài toán

1.1.1 Đặc điểm dữ liệu

Nhãn đánh giá, quan điểm, cảm nhận (opinion polarity) được định nghĩa cho cả bộ dữ liệu khách sạn và nhà hàng 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 tính)

Dữ liệu khách sạn: (được thu thập: https://www.agoda.com/vi-vn/). Hướng dẫn gán nhãn ngữ liệu aspect-sentiment cho chủ đề Khách Sạn (Hotel): Link: VLSP-Hotel

Tập dữ liệu	Số lượng đánh giá	Số lượng khía cạnh
DKS100 - Tập luyện	3000	13949
DKS200 - Tập kiểm tra khớp	2000	7111
DKS300 - Tập kiểm tra	600	2584

Bảng 1.1: Tập dữ liệu khách sạn

Nhãn khía cạnh: gồm bộ thực thể - thuộc tính của thực thể:

- Các loại thực thể (7 loại):
  - HOTEL: đề cập chung đến khách sạn, không nhắc đến feature gì cụ thể.
  - ROOMS: đề cập đến phòng khách sạn về các đặc điểm như kích thước, điều kiện phòng nói chung, đồ nội thất, phòng tắm, giấc ngủ.
  - ROOMS_AMENITIES: đề cập đến các tiện nghi trong phòng như điều hòa, tủ lạnh, lò vi sóng, quầy bar, máy sấy tóc, TV, nhà vệ sinh, ban công, máy pha cà phê, v.v.
  - FACILITIES: đề cập tới một số tiện nghi của khách sạn được phục vụ kèm theo hoặc hoặc cung cấp cho khách hàng như: swimming pool, spa&sauna, beauty salon, restaurants, café, night club, casino, business center, gymnasium, access facility for the differentlyabled, parking, shuttle, laundry, baby sitting or wake up services, sports activities, 24-hour concierge & front desk, information desk, in-room dining, internet access, availability of touristic materialetc.
  - SERVICE: đề cập tới thái độ phục vụ của nhân viên, tính kịp thời, dễ dàng trong giải quyết vấn đề phát sinh, hoặc trong việc tiếp đón, nhận phòng, trả phòng.
  - LOCATION: đề cập tới vị trí của khách sạn, các địa điểm xung quanh khách sạn, v.v.
  - FOOD&DRINKS: đề cập tới ăn sáng, thức ăn, đồ uống nói chung, hoặc các loại thức ăn, đồ uống cụ thể nào đó.
- Các loại thuộc tính (8 loại):
  - GENERAL: đề cập tới một loại thực thể nói chung (hotel, room amenities, rooms, facilities, location, service), view của khách sạn, view của phòng.

- PRICES: đề cập tới giá phòng, giá thức ăn, đồ uống, giá tiện ích, dịch vụ cung cấp bởi khách sạn, hoặc giá khách sạn nói chung.
- DESIGN&FEATURES: đề cập tới thiết kế, trang trí, kích thước của một loại thực thể (hotel, rooms, facilities).
- CLEANLINESS: đề cập tới sự vệ sinh, gọn gàng của phòng, của khu vực chung, và của khách sạn nói chung.
- COMFORT: nói về các thực thể ở góc độ sự thoải mái, tiện nghi cho khách hàng
- QUALITY: nói về chất lượng đồ ăn, thức uống (ví dụ khẩu vị, sự tươi ngon, nhiệt độ, sự chuẩn bị, v.v), hoặc chất lượng các dịch vụ và tiện ích khách sạn cung cấp.
- STYLE&OPTIONS: nói về việc trình bày đồ ăn, thức uống, cách thức phục vụ, các lựa chọn đồ ăn, thực đơn, hoặc tính đa dạng của đồ ăn thức uống phục vụ trong nhà hàng.
- MISCELLANEOUS: cho các attributes không nằm trong những cái ở trên

Dữ liệu nhà hàng: (được thu thập: https://www.foody.vn).

Hướng dẫn gán nhãn dữ liệu aspect-sentiment cho chủ đề Nhà Hàng (Restaurant): Link: VLSP-restaurant

Tập dữ liệu	Số lượng đánh giá	Số lượng khía cạnh
DKS100 - Tập luyện	2961	9297
DKS200 - Tập validation	1290	3443
DKS300 - Tập kiểm tra	500	2414

Bảng 1.2: Tập dữ liệu nhà hàng

Nhãn khía cạnh: gồm bộ thực thể - thuộc tính của thực thể:

- Các loại thực thể (6 loại):
  - restaurant (nhà hàng): nói đến nhà hàng nói chung, tổng kết. Khi không chỉ cụ thể một thực thể nào thì có nghĩa là đề cập chung

đến restaurant. Hầu hết các review chúng ta đều có thể xác định sentiment cho entity này.

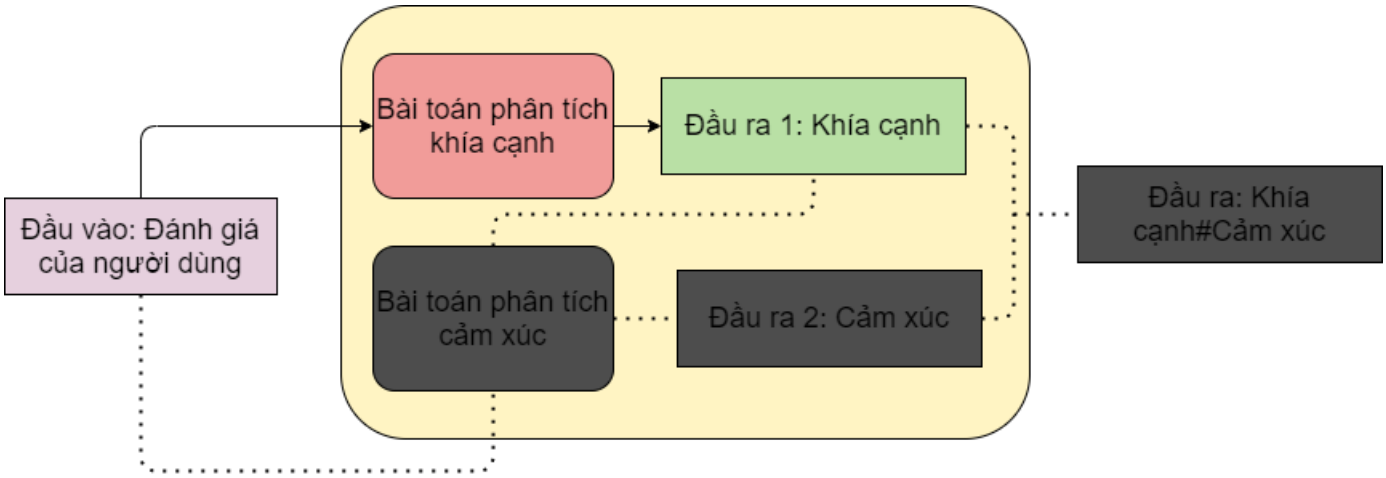
- ambience (không gian): không gian của nhà hàng, liên quan đến giải trí, bên trong có thoáng, dễ chịu không? Ví dụ "nhà hàng bố trí đẹp, có sân vườn rộng, có chỗ cho trẻ em chơi".
- location (vị trí): liên quan vị trí nhà hàng có thuận tiện không, view nhà hàng có ok không, có chỗ gửi xe không, ..
- food (đồ ăn): đồ ăn, các món ăn.
- service (phục vụ): người phục vụ, cách phục vụ khách hàng.
- drinks (đồ uống): đồ uống.

• Các loại thuộc tính (5 loại):

- general (nói chung): thường dùng với restaurant
- quality (chất lượng): thường dùng với food, drink
- price (giá cả): giá của restaurant, hoặc của food, drink, service,
- style_option (kiểu,tùy chọn):dùng cho cách trình bày; kiểu phục vụ; lựa chọn trong thực đơn có phong phú không;
- miscellaneous (thuộc tính khác): không thuộc 4 cái trên

1.1.2 Giới thiệu bài toán phân tích khía cạnh

Bước đầu tiên để giải quyết bài toán đặt ra ta cần giải quyết bài toán nhỏ phân tích khía cạnh theo mô tả như sau:



Hình 1.2: Tổng quan bài toán phân tích khía cạnh

Đối với dữ liệu khách sạn, theo thống kê chúng ta có 34 khía cạnh như sau:

	GENERAL	PRICES	DESIGN&FEATURES	CLEANLINESS	COMFORT	QUALITY	STYLE&OPTIONS	MISCELLANEOUS
HOTEL	✓	✓	✓	✓	✓	✓		✓
ROOMS	✓	✓	✓	✓	✓	✓		✓
ROOMS_AMENITIES	✓	✓	✓	✓	✓	✓		✓
FACILITIES	✓	✓	✓	✓	✓	✓		✓
SERVICE	✓							
LOCATION	✓							
FOOD&DRINKS		✓				✓	✓	✓

Bảng 1.3: Bộ khía cạnh trong lĩnh vực khách sạn

Mục tiêu của bài toán phân tích khía cạnh trên bộ dữ liệu khách sạn là xác định xem trong một câu đánh giá thì người dùng nhắc đến những khía cạnh nào trong 34 khía cạnh trong bảng trên.

Ví dụ: Khách sạn giá rẻ, gần biển. Nhân viên lễ tân thiếu lịch sự với khách, thái_độ khó_chịu.

Chúng ta cần xác định trong đánh giá trên người dùng đã nhắc đến:

HOTEL#PRICES, LOCATION#GENERAL, SERVICES#GENERAL.

Đối với dữ liệu nhà hàng, theo thống kê chúng ta có 12 khía cạnh như sau:

	GENERAL	PRICES	QUALITY	STYLE&OPTIONS	MISCELLANEOUS
RESTAURANT	✓	✓			✓
FOODS		✓	✓	✓	
DRINKS		✓	✓	✓	
AMBIENCE	✓				
SERVICE	✓				
LOCATION	✓				

Bảng 1.4: Bộ khía cạnh trong lĩnh vực nhà hàng

Tương tự như trên bộ dữ liệu khách sạn, với bộ dữ liệu nhà hàng ta cũng cần xác định xem trong một câu đánh giá thì người dùng nhắc đến những khía cạnh nào trong 12 khía cạnh được thống kê ở bảng trên.

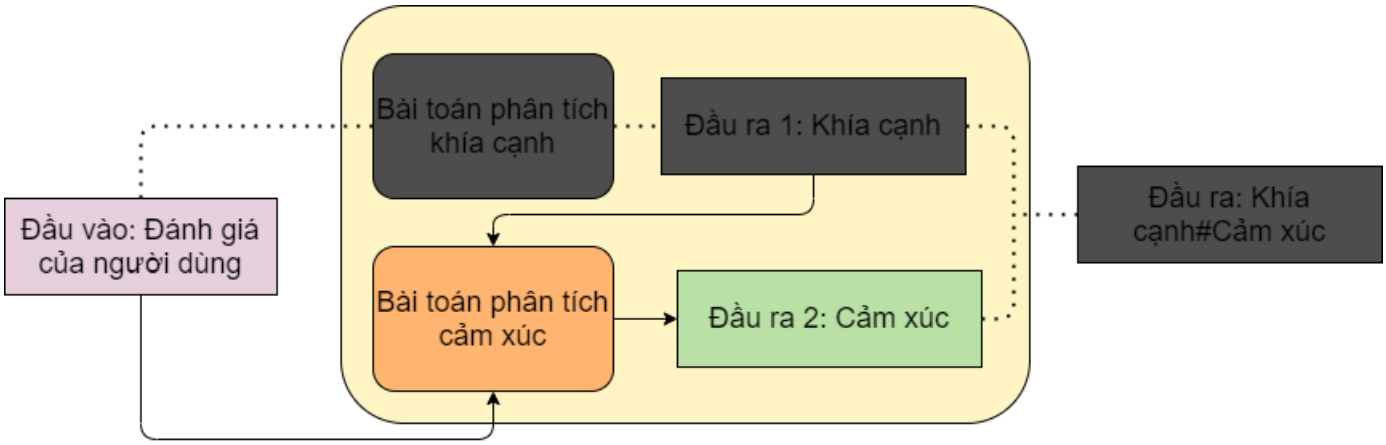
Ví dụ: 180K/suất không phải là cái giá rẻ đối với người Sài Gòn. Nhưng nếu so với giá Chả cá Anh Vũ - Giảng Võ và vị trí ngay khu TT Q3 thì cũng hợp lý. Phải nói là chả có ngon nhất SG mình từng ăn tới lúc này. Bài trí và hương vị khá giống nguyên bản ngoài Hà Nội. Từ chanh ớt, vị mắm tôm cho tới cái chảo rán cá. Duy chỉ có mùi ngũ vị hương hơi đậm là mình ko khoái so với ngoài kia. Creme Brulee khá ngon và ít ngọt. Nói chung sẽ quay lại nhiều.

Đầu ra: restaurant#price, food#quality, food#style_option, restaurant#general là bốn khía cạnh được đề cập đến trong ví dụ.

1.1.3 Giới thiệu bài toán phân tích cảm xúc

Trong bài toán này, sau khi xác định được các khía cạnh của đánh giá thì chúng ta sẽ tiến hành xác định cảm xúc của từng khía cạnh có trong

đánh giá.



Hình 1.3: Tổng quan bài toán phân tích cảm xúc

Trong nghiên cứu trước, tác giả chỉ dừng lại ở việc xác định cảm xúc tích cực và tiêu cực cho một câu đánh giá. Trên thực tế dữ liệu phức tạp hơn nhiều, một câu đánh giá có thể nhắc đến nhiều thực thể, nhiều thuộc tính khác nhau và cũng trên cùng một câu đánh giá người dùng có thể khen chê hoặc chỉ đơn giản là nói một câu không có ý nghĩa gì cả do vậy trong nghiên cứu này tác giả mở rộng bài toán lên bài toán xác định khía cạnh và cảm xúc của các khía cạnh đó.

Ví dụ: Khách sạn giá rẻ, gần biển. Nhân viên lễ tân thiếu lịch sự với khách, thái_độ khó_chịu.

Trong bài toán nhỏ phân tích khía cạnh ở trên đã xác định được các khía cạnh được nhắc đến trong đánh giá là: HOTEL#PRICES, LOCATION#GENERAL, SERVICES#GENERAL. Trong bài toán nhỏ phân tích cảm xúc này sẽ tiến hành xác định đầu ra như sau: HOTEL#PRICES: positive, LOCATION#GENERAL: neutral, SERVICES#GENERAL: negative.

1.2 Các phương pháp tiếp cận thông thường

Có nhiều cách thức tiếp cận đối với bài toán này, tùy thuộc vào đặc trưng của dữ liệu. Tuy nhiên, về mặt tổng quát, ta có thể chia các phương pháp giải quyết cho các bài toán nói trên theo ba nhóm:

- Các mô hình học máy.
- Các mô hình học sâu
- Các mô hình học chuyển tiếp

1.2.1 Các mô hình học máy

Các mô hình học máy mang đến kết quả khá tốt trên các bộ dữ liệu nhỏ, đối với một số bài toán các mô hình học máy cũng có thể cho kết quả tốt hơn các mô hình học sâu. Tuy nhiên, các mô hình này không cho phép học được ngữ nghĩa, quan hệ các từ đối với dữ liệu tuần tự như ngôn ngữ nên không khai thác được hết những đặc trưng quan trọng mà dữ liệu đem lại. Dưới đây là hai mô hình học máy được sử dụng khá nhiều trong xử lý ngôn ngữ tự nhiên:

Mô hình Naïve Bayes Classifier:

Xét bài toán phân loại, phân lớp tập dạng $\Omega = \{X \in \mathbb{R}^d | d \in N\}$ thành C lớp với $C = \{C_i, i = \overline{1,c}\}$ là một phân hoạch của tập dạng Ω. Khi đó cho điểm dữ liệu $X \in \Omega$, việc tính xác suất để điểm dữ liệu rơi vào lớp C_i dựa vào công thức bayes [3]:

$$P\left(C_i|X\right) = \frac{P\left(X|\ C_i\right) P(C_i)}{P(X)}$$

Trong đó: $P\left(X\right) = \sum_{i=1}^{c} P\left(X|C_i\right) P(C_i)$ (Công thức xác suất đầy đủ).

Biểu thức trên nếu tính được, sẽ giúp chúng ta xác định được xác suất để điểm dữ liệu rơi vào mỗi lớp. Từ đó có thể giúp xác định lớp của điểm dữ liệu đó bằng cách chọn ra lớp có xác suất cao nhất:

$$\begin{aligned} c &= arg\max\left(P\left(C_i|X\right)\right),\ i = \overline{1,c} \\ &= arg\max\left(\frac{P\left(X|C_i\right) P\left(C_i\right)}{P\left(X\right)}\right) \\ &= arg\max(P\left(X|C_i\right) P\left(C_i\right)) \end{aligned}$$

(Vì $P(X)$ là một hằng số không phụ thuộc vào C_i nên trong quá trình tính toán có thể bỏ qua $P(X)$).

- $P(C_i)$ là tần suất điểm dữ liệu trong tập huấn luyện rơi vào lớp C_i, hay còn được tính bằng tỉ lệ số điểm dữ liệu trong tập huấn luyện rơi vào lớp C_i chia cho tổng số điểm dữ liệu trong tập huấn luyện.

- Việc tính toán $P(X|C_i)$ thường khá là phức tạp nên để đơn giản hóa việc tính toán người ta thường coi các thuộc tính của vector X là độc lập tuyến tính khi đó ta có công thức:

$$P\left(X|C_i\right) = \prod_{k=1}^{d} P(x_k|C_i) = P\left(x_1|C_i\right) P\left(x_2|C_i\right) \dots P\left(x_d|C_i\right)$$

Việc tính toán $P\left(x_k|C_i\right),\ k = \bar{1,d}; i = \bar{1,c}$ phụ thuộc vào loại dữ liệu, thông thường sẽ có ba quy tắc phân lớp thường dùng phổ biến: Gaussian Naïve Bayes, Multinomial Naïve Bayes, và Bernoulli Naïve Bayes.

Quy tắc phân lớp Gaussian Naive Bayes [10]:

$$g\left(x, \mu, \sigma\right) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$P\left(X|C_i\right) = g(x_k, \mu_{c_i}, \sigma_{c_i})$$

Bộ tham số $\theta = (\mu_{c_i}, \sigma_{c_i})$ được xác định là kỳ vọng μ_{c_i} và phương sai $\sigma^2_{c_i}$ của các thuộc tính x_k tương ứng.

Tuy nhiên trong thư viện sklearn bộ tham số $\theta = (\mu_{c_i}, \sigma_{c_i})$ được tính toán dựa trên công thức:

$$\theta = (\mu_{c_i}, \sigma_{c_i}) = arg\max \prod_{k=1}^{d} P(x_k^{(c_i)}|\mu_{c_i}, \sigma_{c_i}),\ i = \bar{1,c}$$

Quy tắc phân lớp Multinomial Naïve Bayes [10]:

Mô hình này chủ yếu được sử dụng trong phân loại văn bản mà vector đặc trưng được tính bằng Bags of Words. Lúc này, mỗi văn bản được biểu diễn bởi một vector có độ dài d chính là số từ trong từ điển. Giá trị của thành phần thứ k trong mỗi vector chính là số lần từ thứ k xuất hiện trong văn bản đó. Khi đó, $P\left(x_k|C_i\right)$ tỉ lệ với tần suất từ thứ k (hay đặc trưng thứ k cho trường hợp tổng quát) xuất hiện trong các văn bản của lớp C_i.

Giá trị này có thể được tính bằng cách:

$$\lambda_{c_i}^k = \ P\left(x_k|C_i\right) = \frac{N_{c_i}^k}{N_{c_i}}$$

Trong đó:

- $N_{c_i}^k$ là tổng số từ thứ k xuất hiện trong lớp C_i, nó còn được tính bằng tổng các thành phần thứ k của các vector đặc trưng ứng với lớp C_i.
- N_{c_i} là tổng số từ xuất hiện trong lớp C_i kể cả lặp.
- $N_{c_i} = \ \sum_{k=1}^{d} N_{c_i}^k \ => \ \sum_{k=1}^{d} \lambda_{c_i}^k = 1$

Cách tính này có một hạn chế là nếu có một từ mới chưa bao giờ xuất hiện trong lớp C_i thì giá trị $\lambda_{c_i}^k = 0$ điều này dẫn đến $P\left(X|C_i\right) = \ 0$ bất kể các giá trị còn lại có lớn thế nào. Việc này sẽ dẫn đến kết quả không chính xác, để giải quyết việc này, một kỹ thuật được gọi là Laplace smoothing được áp dụng:

$$\widehat{\lambda_{c_i}^k} = \frac{N_{c_i}^k + \alpha}{N_{c_i} + d\alpha}$$

Với α là một số dương tùy ý, thông thường $\alpha = 1$, để tránh trường hợp $N_{c_i}^k = 0$, và mẫu số được cộng thêm một giá trị $d\alpha$ để đảm bảo $\sum_{k=1}^{d} \widehat{\lambda_{c_i}^k} = 1$. Như vậy mỗi lớp C_i sẽ được mô tả bằng bộ các số dương có tổng bằng 1:

$$\widehat{\lambda_{c_i}} = \left\{\widehat{\lambda_{c_i}^1},\ \widehat{\lambda_{c_i}^2}, \ldots, \widehat{\lambda_{c_i}^d}\right\}$$

Quy tắc phân lớp Bernoulli Naïve Bayes [10]:

Mô hình này được áp dụng cho các loại dữ liệu mà mỗi thành phần là một giá trị binary - bằng 0 hoặc 1. Ví dụ: cũng với loại văn bản nhưng thay vì đếm tổng số lần xuất hiện của 1 từ trong văn bản, ta chỉ cần quan tâm từ đó có xuất hiện hay không. Khi đó $P\left(x_k|C_i\right)$ được tính bằng công thức:

$$P\left(x_k|C_i\right) = P\left(k|C_i\right)^{x_k}(1 - P(k|C_i))^{1-x_k}$$

Trong đó $P(k|C_i)$ xác suất từ thứ k xuất hiện trong các văn bản của lớp C_i.

Mô hình Support Vector Machine (SVM)

Khoảng cách từ một điểm tới một siêu phẳng:

Trong không gian 2 chiều, khoảng cách từ một điểm có toạ độ (x_0, y_0) tới đường thẳng có phương trình $\omega_0 + \omega_1 x_1 + \omega_2 x_2 = 0$ được xác định bởi:

$$\frac{|\omega_0 + \omega_1 x_0 + \omega_2 y_0|}{\sqrt{\omega_1^2 + \omega_2^2}}$$

Trong không gian 3 chiều, khoảng cách từ một điểm có toạ độ (x_0, y_0, z_0) tới một mặt phẳng có phương trình $\omega_0 + \omega_1 x_1 + \omega_2 x_2 + \ \omega_3 x_3 = 0$ được xác định bởi:

$$\frac{|\omega_0 + \omega_1 x_0 + \omega_2 y_0 + \omega_3 z_0 \ |}{\sqrt{\omega_1^2 + \omega_2^2 + \omega_3^2}}$$

Hơn nữa, nếu bỏ dấu trị tuyệt đối ở tử số, chúng ta có thể xác định được điểm đó nằm về phía nào của đường thẳng hay mặt phẳng đang xét. Những điểm làm cho biểu thức trong dấu giá trị tuyệt đối mang dấu dương nằm về cùng một phía (có thể gọi đây là phía dương), những điểm làm cho biểu thức trong dấu giá trị tuyệt đối mang dấu âm nằm về phía còn lại (có thể gọi đây là phía âm). Những điểm nằm trên đường thẳng hoặc mặt phẳng sẽ làm cho tử số có giá trị bằng 0, tức khoảng cách bằng 0.

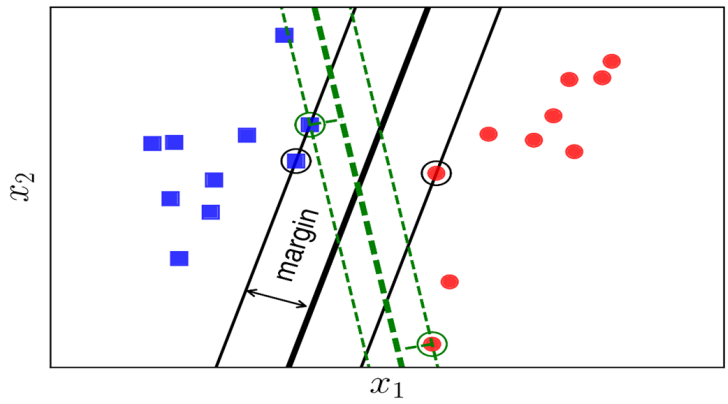
Tổng quát, với không gian n chiều ta có khoảng cách từ một điểm (vector) có tọa độ X_0 đến siêu phẳng có phương trình $\omega_0 + \ W^T X = 0$ được xác định bởi:

$$\frac{|\omega_0 + \ W^T X_0|}{||W||_2}$$

Trong đó: $||W||_2 = \ \sqrt{\sum_{i=1}^{n} \omega_i^2}$ là chuẩn của W theo trọng hàm $u = 2$.

Mô hình phân chia hai lớp sử dụng Support Vector Machine:

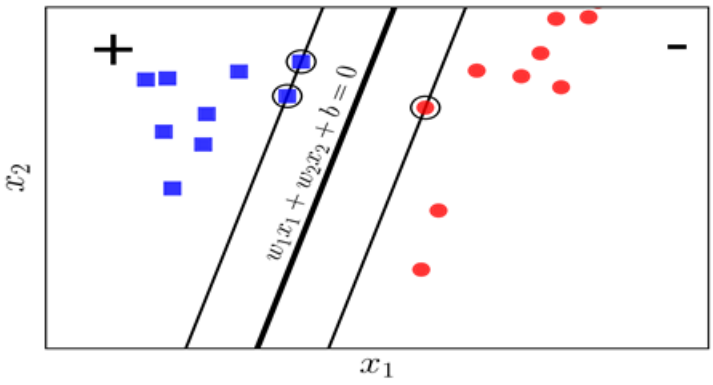
Giả sử rằng có hai lớp khác nhau được mô tả bởi các điểm trong không gian nhiều chiều, hai lớp này là phân biệt tuyến tính, tức tồn tại một siêu phẳng phân chia chính xác hai lớp đó. Việc tìm một siêu mặt phẳng phân chia hai lớp đó, sao cho tất cả các điểm thuộc một lớp nằm về cùng một phía của siêu mặt phẳng đó và ngược phía với toàn bộ các điểm thuộc lớp còn lại. Hoặc một lớp nằm trọn vẹn về phía dương của siêu phẳng hoặc nằm trọn vẹn về phía âm của siêu phẳng.



Hình 1.4: Lề của hai lớp là bằng nhau và lớn nhất có thể

Chúng ta cần một đường phân chia sao cho khoảng cách từ điểm gần nhất của mỗi lớp (các điểm được khoanh tròn) tới đường phân chia là như nhau, khoảng cách như nhau này được gọi là Lề (Margin). Lề của cả hai lớp càng lớn thì việc phân lớp càng tốt (càng rạch ròi càng tốt).

Bài toán tối ưu trong SVM giúp giải quyết vấn đề tìm ra đường phân chia sao cho lề của hai lớp là lớn nhất [13].



Hình 1.5: Phân tích mô hình SVM

Các cặp dữ liệu trong tập huấn luyện là (X_1, y_1) ; (X_2, y_2) ; . . . ; (X_n, y_n) ; với vector (X_i) thể hiện là điểm dữ liệu trong tập huấn luyện và (y_i) thể hiện nhãn của điểm dữ liệu đó.

Khoảng cách từ điểm (X_i, y_i) , $\forall i \in [1, n]$ tới mặt phân chia có phương

trình $\omega_0 + \ W^T X = 0$ được xác định bởi:

$$\frac{y_i(\omega_0 + \ W^T X_i)}{||W||_2}$$

Ta có y_i luôn cùng dấu với phía của X_i, như vậy y_i luôn cùng dấu với $\omega_0 + \ W^T X_i$ và giá trị biểu thức $y_i \left(\omega_0 + \ W^T X_i\right) > 0$. [13] [14]

Với mặt phân chia như trên, lề được tính bằng khoảng cách gần nhất từ một điểm dữ liệu tới mặt phân chia đó (điểm dữ liệu được chọn là điểm bất kỳ thuộc một trong hai lớp):

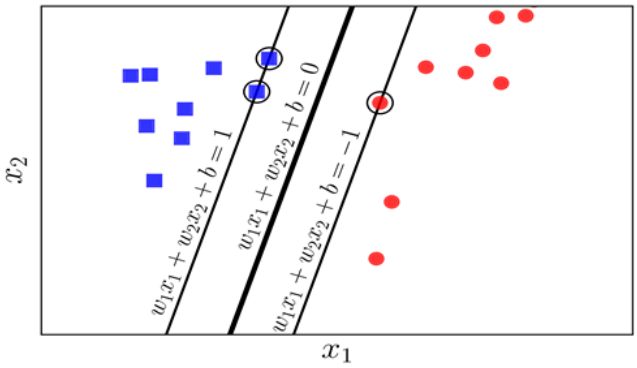
$$margin = \ \min_i \frac{y_i(\omega_0 + \ W^T X_i)}{||W||_2}, \ \forall i \in [1, n]$$

Bài toán tối ưu trong SVM chính là bài toán tìm bộ giá trị tham số $(W, \ \omega_0)$ sao cho lề đạt giá trị lớn nhất:

$$(W, \ \omega_0) = arg \max_{W, \ \omega_0} \{\min_i \frac{y_i(\omega_0 + \ W^T X_i)}{||W||_2}\}$$

$$= arg \max_{W, \omega_0} \frac{1}{||W||_2} \min_i y_i \left(\omega_0 + \ W^T X_i\right)$$

Giả sử $y_i \left(\omega_0 + \ W^T X_i\right) = 1$ khi đó ta có:



Hình 1.6: các điểm gần mặt phân cách nhất được khoanh tròn

Như vậy $\forall i \in [1, n]$ ta có: $y_i \left(\omega_0 + \ W^T X_i\right) \geq 1$. Bài toán trở thành bài toán quy hoạch tuyến tính được phát biểu như sau:

$$(W,\ \omega_0) = arg \max_{(W,\ \omega_0)} \frac{1}{||W||_2}$$
$$\text{v.đ.k:}\ \ y_i(\omega_0 + W^T X_i) \geq 1,\ \forall i \in [1, n]$$

1.2.2 Các phương pháp học sâu

Các mô hình học sâu như CNN, RNN hay LSTM được áp dụng rộng rãi trong lĩnh vực thị giác máy tính và xử lý ngôn ngữ tự nhiên. Nó đang cho thấy hiệu quả vượt trội so với các mô hình học máy trong nhiều bài toán. Có hai lý do cơ bản để mô hình học sâu đạt hiệu quả:

- Có thể khai thác được các đặc trưng quan trọng của ngôn ngữ.
- Có khả năng khai thác ngữ nghĩa, quan hệ giữa các từ dựa trên tính chất tuần tự của dữ liệu.

Mặc dù kết quả của các mô hình học sâu rất tốt, nhưng nhược điểm của các mô hình này là đòi hỏi lượng dữ liệu huấn luyện rất lớn, mà hầu như là bất khả thi trong thực tế.

Trong nghiên cứu trước tác giả có nêu chi tiết về các mô hình CNN, RNN và LSTM trong xử lý ngôn ngữ tự nhiên. Dù kết quả mang lại tốt nhưng cũng tồn tại nhược điểm:

- Các mạng CNN có thể dễ dàng được thực hiện song song ở một tầng nhưng không có khả năng nắm bắt các phụ thuộc chuỗi có độ dài biến thiên.
- Các mạng RNN có khả năng nắm bắt các thông tin cách xa nhau trong chuỗi có độ dài biến thiên, nhưng không thể thực hiện song song trong một chuỗi.

Ta có thể thấy rằng hai phương pháp học sâu và học máy hay chính trong những mô hình học sâu đều có những ưu và nhược điểm riêng biệt tương phản nhau. Trong khi học máy thống kê có khả năng đem lại kết

quả khá tốt với tập dữ liệu nhỏ thì học sâu cần tập dữ liệu rất lớn để phát huy tính hiệu quả. Thay vào đó, kết quả của các mô hình học sâu mang tính tổng quát lớn hơn nhiều.

Như vậy, câu hỏi được đặt ra là làm thế nào kết hợp được điểm mạnh của các mô hình học máy và các mô hình học sâu như CNN, RNN, LSTM ... Một phương pháp đem lại kết quả mang tính tổng quát cao nhưng không đòi hỏi quá nhiều dữ liệu huấn luyện mà tốc độ xử lý nhanh sẽ có tính ứng dụng rất cao. Phương pháp mà tác giả đang nói đến chính là phương pháp học chuyển tiếp. Cụ thể phương pháp sẽ được trình bày ở Chương 2.

Chương 2

Phương pháp học chuyển tiếp trong xử lý ngôn ngữ tự nhiên

Học chuyển tiếp (Transfer learning) là một phương pháp học tập trung vào việc khai thác các kiến thức thu được trong quá trình giải quyết một vấn đề lớn và áp dụng nó vào một bài toán nhỏ hơn nhưng có liên quan. Phương pháp học chuyển tiếp có liên quan mật thiết đến vấn đề học đa tác vụ và chuyển đổi ngữ cảnh, mặt khác học chuyển tiếp thường sử dụng các mô hình học sâu nhưng bản thân học chuyển tiếp không phải là một lĩnh vực của học sâu. Có nhiều yếu tố để phân chia học chuyển tiếp thành các loại khác nhau. Tuy nhiên, nói riêng trong lĩnh vực xử lý ngôn ngữ tự nhiên, học chuyển tiếp thường được áp dụng bằng cách sử dụng các mô hình tiền huấn luyện [8]. Mục tiêu của các mô hình này là học đặc trưng của ngôn ngữ, mối quan hệ của từ, ngữ cảnh trong câu văn cũng như mối quan hệ của các câu văn trong văn bản nhờ vào các tác vụ cụ thể. Khi miền dữ liệu huấn luyện đủ lớn, mô hình được kỳ vọng có khả năng biểu diễn hiệu quả ý nghĩa của từ, ngữ cảnh và câu văn.

Trong khuôn khổ đồ án này, tác giả tập trung vào mô hình BERT. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding [4] là mô hình biểu diễn hai chiều cho ngôn ngữ tuân theo mô hình tự mã hóa (autoencoder) dựa trên khả năng của cơ chế tự chú ý (self-attention). Các kết quả mới nhất cho thấy khả năng ấn tượng của

mô hình này với các bộ dữ liệu quan trọng của xử lý ngôn ngữ tự nhiên.

2.1 Mô hình học chuyển tiếp

Để hiểu được mô hình học chuyển tiếp, tác giả sẽ trình bày tổng quan về cơ chế chú ý (attention mechanism) - nền tảng cốt lõi của các mô hình học chuyển tiếp và mô hình transformer.

2.1.1 Cơ chế chú ý - Attention mechanism

Trong thực tế, khi xét một câu văn mỗi từ trong câu văn đó chỉ liên quan mật thiết đến một số từ trong câu chứ không nhất thiết là toàn bộ các từ có mặt trong đó. Do đó cơ chế chú ý được đưa ra nhằm mục đích ghi nhớ và giải mã ý nghĩa của câu văn, một trong những vấn đề của dịch máy. Cơ chế chú ý cho phép mô hình tập trung vào một vài từ, một vài ngữ cảnh địa phương trong câu thay vì việc coi tất cả các từ có ý nghĩa như nhau. Mục tiêu của cơ chế chú ý là đưa ra các trọng số tương ứng với từng trạng thái đầu vào đại diện cho sự ảnh hưởng của trạng thái đó lên ý nghĩa toàn cục của câu văn hoặc ý nghĩa cục bộ tại thời điểm mà mô hình đang xem xét. Các khái niệm mã hóa - giải mã là các khái niệm cơ bản được sử dụng trong máy dịch, và sẽ được sử đụng để giải thích phương pháp hoạt động của mô hình attention thông thường. Theo đó, bộ mã hóa cho phép mã hóa một chuỗi thành một vector, trong khi bộ giải mã thực hiện việc giải mã bộ vector đó thành chuỗi tương ứng.

Cơ chế chú ý ban đầu được đề xuất để giải quyết bài toán sinh chuỗi (seq2seq) thông thường bằng cơ chế mã hóa - giải mã. Cơ chế chú ý được đưa vào để chỉnh sửa trọng số của vector trong phiên giải mã, giúp mô hình tập trung vào những thành phần quan trọng trong chuỗi thay vì toàn bộ chuỗi. Dưới đây là mô tả toán học của hai quá trình mã hóa và giải mã:

- Mã hóa:

$$h_t = f(x_t, h_{t-1})$$
$$c = q(\{h_1, ..., h_{T_x}\})$$

- Giải mã:

$$p(y) = \prod_{t=1}^{T} p(y_t|\{y_1, ..., y_{t-1}\}, c)$$

Trong đó, ở bước mã hõa, h_t là trạng thái ẩn tương ứng với dữ liệu đầu vào thứ t, f là hàm số được biểu diễn bởi mạng neuron, c đại diện cho ngữ cảnh, ý nghĩa của toàn bộ câu văn. Mục tiêu của bước này là tìm ra các vector đại diện cho ý nghĩa của từng từ ngữ cũng như ý nghĩa của toàn bộ câu văn. Trong bước giải mã, xác suất của mỗi từ $p(y_t|\{y_1, ..., y_{t-1}\}, c)$ được tính toán dựa trên các từ phía trước y_{t-1}, vector ngữ cảnh s_t và vector ý nghĩa của câu văn c được tạo ra trong quá trình mã hóa.

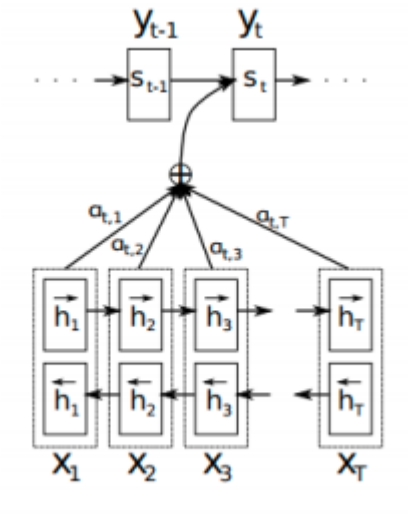
Mô hình attention đầu tiên được đề xuất bởi Bahdanau và cộng sự vào năm 2015 [2] được gọi là soft attention hay attention mềm. Mô hình toán học của phiên bản attention này được thể hiện như sau:

- Trong hàm phân phối xác suất của bước giải mã, một tham số điều kiện tương ứng với trạng thái t được sử dụng để kiểm soát thông tin ảnh hưởng đến bước giải mã này:

  $$p(y_t|\{y_1, ..., y_{t-1}\}, x) = g(y_{t-1}, s_t, c_t)$$

  trong đó $s_t = f(s_{t-1}, y_{t-1}, c_t)$

- Vecto c_t tương ứng với từng thành phần của chuỗi y_t được tính toán tuần tự là tổng có trọng số các vector ẩn tương ứng với từng vị trí đầu vào của chuỗi.



Hình 2.1: Soft attention

$$e_{tj} = a(s_{t-1}, h_t)$$

$$\alpha_{tj} = \frac{exp(e_{tj})}{\sum_{k=1}^{T_x} exp(e_{tk})}$$

$$c_t = \sum_{j=1}^{T_x} \alpha_{tj} h_j$$

Các biến thể của cơ chế chú ý

- Chú ý toàn cục - Global attention: Ý tưởng của mô hình chú ý toàn cục là xem xét tất cả các trạng thái ẩn của bước mã hóa khi tính toán vector ngữ cảnh c_t [7].

$$a_t(s) = align(h_t, \bar{h_s})$$

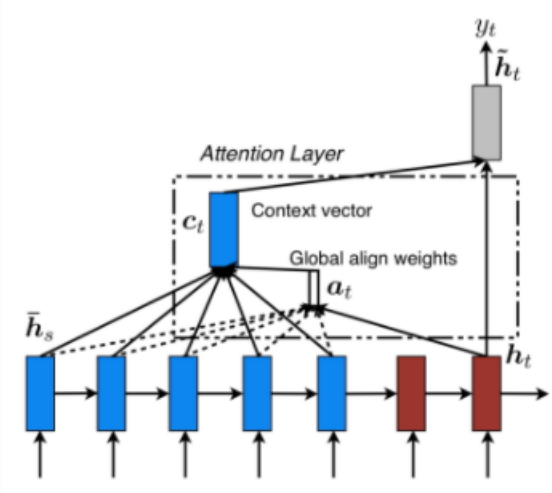
$$= \frac{exp(score(h_t, \bar{h_s}))}{\sum_{s'} exp(score(h_t, \bar{h_{s'}}))}$$

Trong đó:

– $\bar{h_s}$: trạng thái ẩn nguồn

– h_t: trạng thái ẩn đích

– score(): hàm tính toán mối liên hệ ngữ cảnh

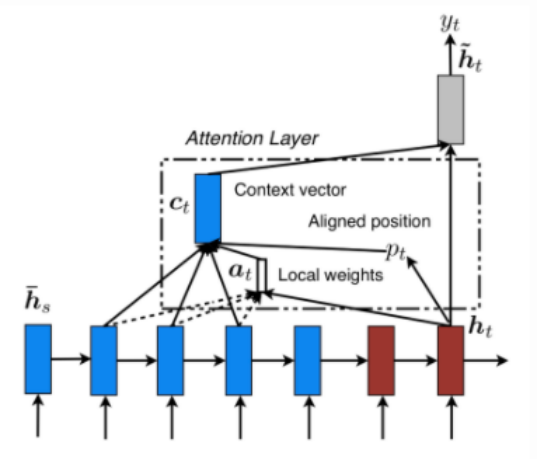


Hình 2.2: Cơ chế chú ý toàn cục - Global attention

Chúng ta xem xét ba công thức tính toán hàm score() sau:

$$score(h_t, \bar{h_s}) = \begin{cases} h_t^T \bar{h_s} \text{ hàm nhân} \\ h_t^T W_a \bar{h_s} \text{ hàm tổng quát} \\ v_a^T tanh(W_a[h_t; \bar{h_s}]) \text{ hàm nối} \end{cases}$$

- Chú ý địa phương - Local attention: Nhược điểm của chú ý toàn cục là nó phải chú ý đến tất cả các từ phía nguồn (phía đầu vào), điều này có thể nắm bắt toàn bộ câu văn tuy nhiên với những câu dài có số lượng từ lớn thì việc này rất tốn kém và không khả thi. Do đó chú ý địa phương được tạo ra để giải quyết vấn đề này [7].



Hình 2.3: Cơ chế chú ý địa phương - Local attention

Đầu tiên mô hình tạo ra một vector vị trí p_t cho mỗi từ đích tại thời điểm t. Vector ngữ cảnh c_t được tính toán như giá trị trung bình có trọng số trên tập hợp $[p_t + D, p_t + D]$; Trong đó D được chọn dựa theo kinh nghiệm (là tham số có thể được chọn tùy thuộc từng bài toán khác nhau). Khi đó vector vị trí căn chỉnh dự đoán - Predictive alignment (local-p):

$$p_t = S.sigmoid(v_p^T tanh(W_p h_t))$$

W_p, v_p là tham số mà mô hình cần học để dự đoán vector vị trí. $p_t \in [0, S]$. Sau đó ta có:

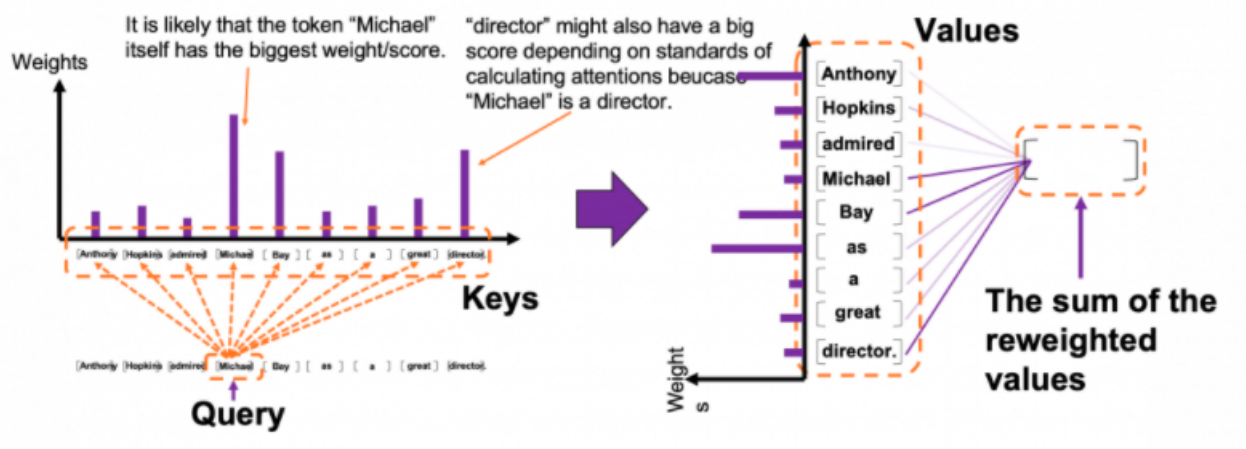
$$a_t(s) = align(h_t, \bar{h_s}) exp(-\frac{(s - p_t)^2}{2\sigma^2}$$
$$= \frac{exp(score(h_t, \bar{h_s}))}{\sum_{s'} exp(score(h_t, \bar{h_{s'}}))} exp(-\frac{(s - p_t)^2}{2\sigma^2}$$

- Cơ chế tự chú ý - Self attention: cơ chế tự chú ý là quy trình áp dụng cơ chế chú ý được mô tả ở phần trên cho tất cả các vị trí của chuỗi đầu vào. Điều này được thực hiện bằng cách tạo ra bộ ba vector (query, key, value) cho tất cả các vị trí của chuỗi, sau đó áp dụng cơ chế chú ý cho mỗi vị trí x_i. Các vector key và query ở vị trí x_i được sử dụng cho tất cả các vị trí khác [1].
  Hàm chú ý được định nghĩa $f : (q, k)\&v \longrightarrow$ Attention score
  Hàm tính điểm đo mối quan hệ ngữ cảnh giữa vector query và vector key tương ứng: $score(q, k_i) = \alpha(q, k_i)$
  Hàm $\alpha()$ (scaled Dot-product attention):

$$\alpha(q, k) = \frac{< q, k >}{\sqrt{d}}$$

Trong đó: $q, k \in \mathbb{R}^d$. Việc giảm giá trị của hàm α nhằm mục đích tăng sự ổn định khi chiều của các vector key, value cũng như query tăng lên. Sau đó tính toán kết quả đầu ra:

$$Attention_i(q, k_i, v_i) = Softmax(\alpha(q, k_i) * v_i)$$ [1].



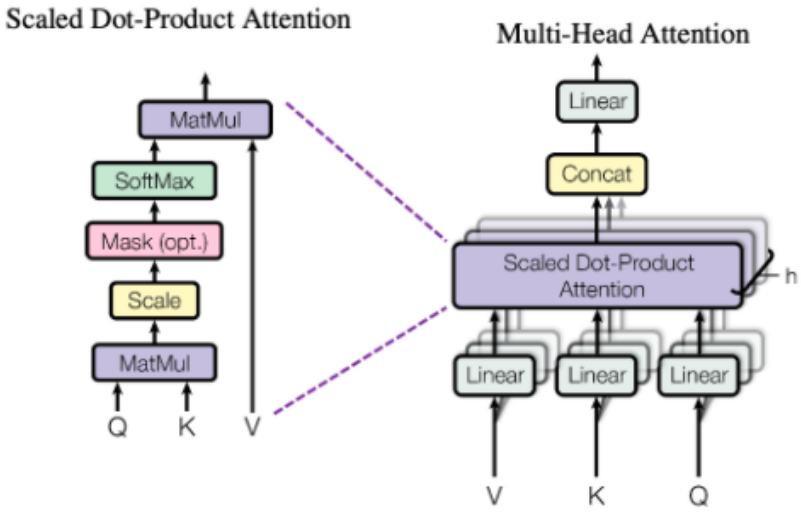
Hình 2.4: Cơ chế tự chú ý - Self attention

Cơ chế chú ý có thể tính toán song song nhiều bộ query và key và value do đó giúp tăng tốc độ tính toán (thay vì tính toán tuần tự như mạng RNN, LSTM . . .). Tuy nhiên cơ chế chú ý sẽ không lưu giữ được thứ tự, yếu tố thời gian của chuỗi đầu vào.

$$\alpha(Q, K) = \frac{QK^T}{\sqrt{(d)}}$$

$$Attention(Q, K, V) = Softmax(\alpha(Q.K))V$$

- Cơ chế chú ý đa diện - Multi-Head Attention:



Hình 2.5: Cơ chế chú ý đa diện - Multi head attention

Thay vì như tự chú ý với mỗi bộ (Q, K, V) được tính toán chú ý một lần với số chiều d_{model} thì cơ chế chú ý đa diện cho phép mô hình tính toán chú ý h lần với không gian có số chiều tương ứng d_{model}/h (head). Với mỗi head, bộ ma trận (Q, K, V) được chiếu riêng biệt lên không gian d_{model}/h chiều và tính toán tự chú ý. Kết quả của mỗi head sau đó được nối lại và áp dụng một phép chiếu tuyến tính để đưa về không gian có số chiều tương ứng với bộ (Q, K, V) ban đầu [1].

Mô hình tính toán được mô tả như sau:

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O$$

$$\text{Trong đó: } head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

$$W_i^Q \in \mathbb{R}^{d_{model}\text{x}d_k} \text{ và } W_i^K \in \mathbb{R}^{d_{model}\text{x}d_k}$$

$$W_i^V \in \mathbb{R}^{d_{model}\text{x}d_v} \text{ và } W^O \in \mathbb{R}^{hd_v\text{x}d_{model}}$$

2.1.2 Mô hình transformer

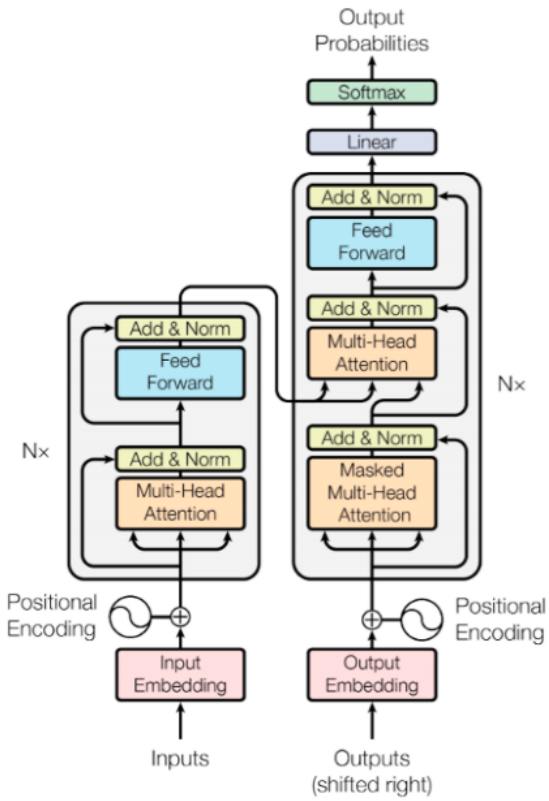
Các mạng CNN có thể dễ dàng được thực hiện song song ở một tầng nhưng không có khả năng nắm bắt các phụ thuộc chuỗi có độ dài biến thiên.

Các mạng RNN có khả năng nắm bắt các thông tin cách xa nhau trong chuỗi có độ dài biến thiên, nhưng không thể thực hiện song song trong một chuỗi.

Để kết hợp các ưu điểm của CNN và RNN, [Vaswani et al., 2017] [1] đã thiết kế một kiến trúc mới gọi là Transformer, song song hóa bằng cách học chuỗi hồi tiếp với cơ chế tập chú ý, đồng thời mã hóa vị trí của từng phần tử trong chuỗi. Kết quả là ta có một mô hình tương thích với thời gian huấn luyện ngắn hơn đáng kể.

Mô hình học chuyển tiếp lần đầu tiên được giới thiệu trong bài báo gốc có tên "Attention Is All You Need" [1], và từ tiêu đề, có thể dễ dàng thấy rằng cơ chế chú ý đóng vai trò quan trọng trong mô hình này.

Hình bên dưới, được sử dụng trong tài liệu gốc về học chuyển tiếp.



Hình 2.6: Kiến trúc mô hình transformer

Kiến trúc mô hình học chuyển tiếp có hai phần là mã hóa - encoder và phần giải mã - decoder.

- Encoder: bao gồm N layer liên tiếp xếp chồng lên nhau. Mỗi layer bao gồm 2 sub_layer trong nó:
  - Sub layer 1: Cơ chế tự chú ý đa diện - Multi-head self-attention.
  - Sub layer 2 Fully-connected feed-forward.

  Chúng ta sẽ sử dụng một kết nối dư - residual conection ở mỗi sub-layer ngay sau layer normalization thiết kế kết nối dư giống với kết nối dư trong mạng ResNet [6].

- Decoder: Decoder cũng bao gồm N layer xếp chồng lên nhau. Mỗi layer sẽ bao gồm 3 sub_layer trong nó.

- Sub layer 1: Mask multi-head self-attention: cơ chế giống hệt sub layer thứ nhất (Encoder) tuy nhiên nó sẽ che đi từ tương lai (nghĩa là sẽ không để model đào tạo từ tương lai). Tại bước thứ i của decoder chúng ta chỉ biết được các từ ở vị trí nhỏ hơn i nên việc điều chỉnh đảm bảo attention chỉ áp dụng cho những từ nhỏ hơn vị trí thứ i.
- Sub layer 2: Cơ chế tự chú ý đa diện - Multi-head self-attention.
- Sub layer 3: Fully-connected feed-forward.

Cơ chế kết nối dư - residual connection cũng được áp dụng tương tự như trong Encoder.

Lưu ý là chúng ta luôn có một bước cộng thêm Positional Encoding vào các input của encoder và decoder nhằm đưa thêm yếu tố thời gian vào mô hình làm tăng độ chuẩn xác. Đây chỉ đơn thuần là phép cộng vector mã hóa vị trí của từ trong câu với vector biểu diễn từ. Chúng ta có thể mã hóa dưới dạng onehot vector vị trí hoặc sử dụng hàm sin, cos.

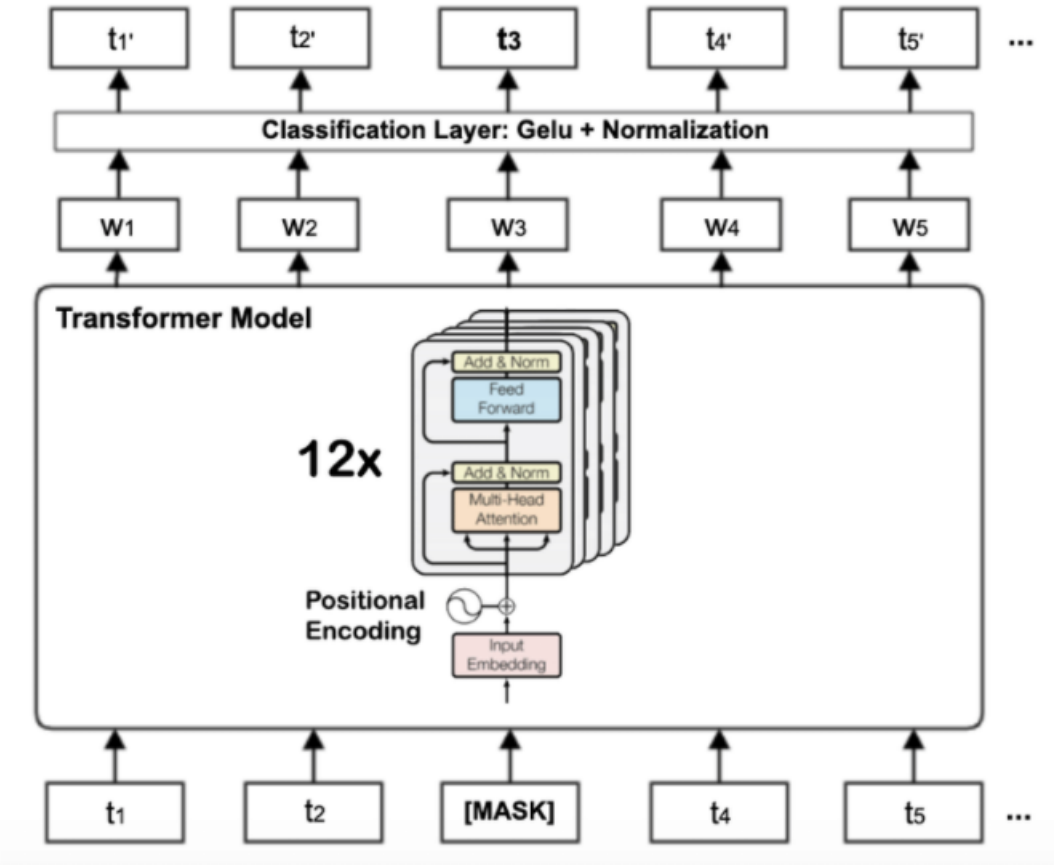
2.2 Mô hình BERT - Mô hình biểu diễn hai chiều tiền huấn luyện

Mô hình BERT được giới thiệu vào cuối năm 2018 [4], đến hiện tại mô hình này vẫn được coi là một trong những mô hình tốt nhất cho các tác vụ xử lý ngôn ngữ tự nhiên. Nó cho phép đào tạo mô hình với thách thức về thiếu hụt dữ liệu do nó đã học tập mỗi quan hệ ngữ nghĩa giữa các từ trong tập dữ liệu rất lớn nhờ các tác vụ học tập tự giám sát (self-supervised), nhờ đó mô hình có thể học được kiến thức tổng quan của ngôn ngữ và áp dụng vào những tác vụ khác cụ thể hơn.

Kiến trúc mô hình BERT

Mô hình BERT là mô hình mã hóa sử dụng nhiều lớp Transformer mã hóa, trong đó các lớp Transformer được sử dụng hoàn toàn tương tự như mô hình được đề xuất và thực thi gốc của Vasawani [1]. Kiến trúc của

Transformer được trình bày chi tiết ở phần 2.1.2.



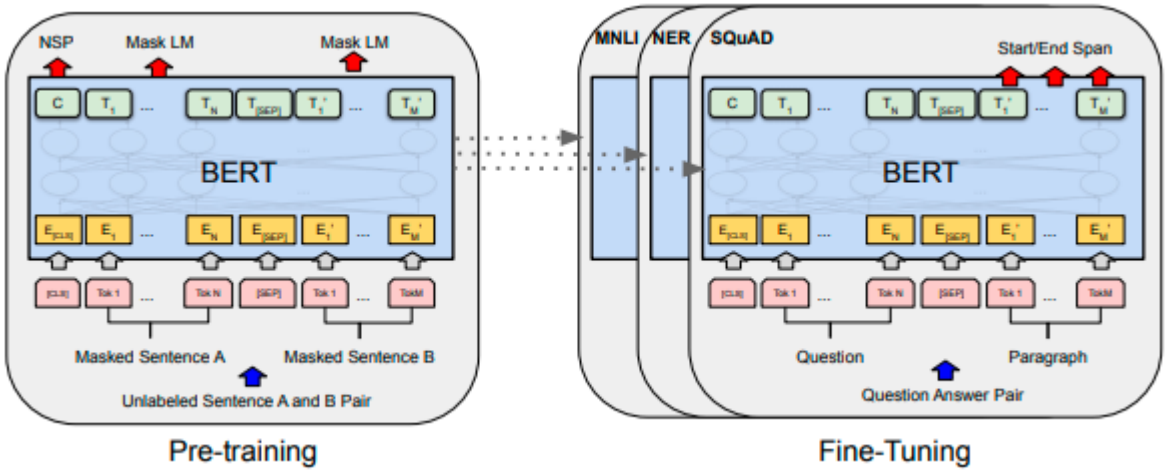
Hình 2.7: Kiến trúc mô hình BERT

Mô hình BERT định nghĩa số lượng lớp (khối Transformer mã hóa) là L, kích thước lớp ẩn là H và số lượng head của self-attention là A. Trong tất cả các cài đặt, BERT sử dụng kích thước của lớp truyền thẳng là 4H, tức là 3072 với H = 768 và 4096 với H = 1024 [4]. BERT có hai kiến trúc mô hình cơ bản như sau:

- BERT base: Bao gồm 12 khối transformer, 12 head self-attention heads,and the hidden size of 768. Với tổng cộng 110 triệu tham số.
- BERT large: Bao gồm 24 khối transformer, 24 head self-attention heads,and the hidden size of 1024. Với tổng cộng 340 triệu tham số.

Triển khai chi tiết mô hình BERT sẽ gồm hai phần: tiền huấn luyện - pretrain và tinh chỉnh finetune. Trong quá trình tiền huấn luyện mô

hình được đào tạo trên bộ dữ liệu lớn với hai nhiệm vụ Masked-Language Modeling (Masked LM) và Next Sentence Prediction (NSP) [4]. Còn trong quá trình tinh chỉnh mô hình, bước đầu tiên là khởi tạo các tham số được tiền huấn luyện và các tham số tinh chỉnh bằng các sử dụng dữ liệu được gán nhãn từ nhiệm vụ cụ thể.



Hình 2.8: Quá trình đào tạo trước và tinh chỉnh của mô hình BERT

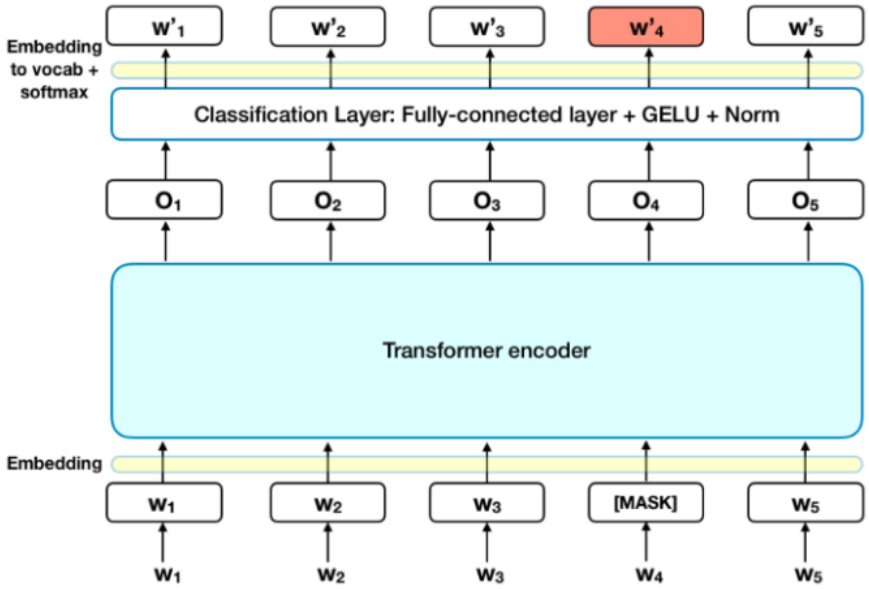
2.2.1 Mô hình BERT tiền huấn luyện

Task #1: Masked Language Modeling (Masked LM) [4]:

Khoảng 15% các token của câu input trong đó 80% số đó được thay thế bởi [MASK] token, 10% thay thế bằng từ bất kỳ, 10% giữ nguyên từ đúng trước khi truyền vào model đại diện cho những từ bị che dấu (masked). Mô hình sẽ dựa trên các từ không được che dấu (non-masked) xung quanh [MASK] và đồng thời là ngữ cảnh của [MASK] để dự báo giá trị gốc của từ được che dấu. Số lượng từ được che dấu được lựa chọn là một số ít (15%) để tỷ lệ ngữ cảnh chiếm nhiều hơn (85%). Bản chất của kiến trúc BERT vẫn là một mô hình seq2seq gồm hai pha mã hóa giúp nhúng các từ ở đầu vào và giải mã giúp tìm ra phân phối xác suất của các từ ở đầu ra.

Kiến trúc Transfomer encoder được giữ lại trong tác vụ Masked LM. Sau khi thực hiện self-attention và feed forward ta sẽ thu được các vector nhúng ở output là $O_1, O_2, ..., O_n$ Hàm softmax có tác dụng tính toán phân

phối xác suất cho từ đầu ra, chúng ta thêm một Fully connect layer ngay sau Transformer Encoder. Số lượng units của fully connected layer phải bằng với kích thước của từ điển. Cuối cùng ta thu được vector nhúng của mỗi một từ tại vị trí MASK sẽ là embedding vector giảm chiều của vector O_i khi đi qua fully connected layer như mô tả hình vẽ sau:



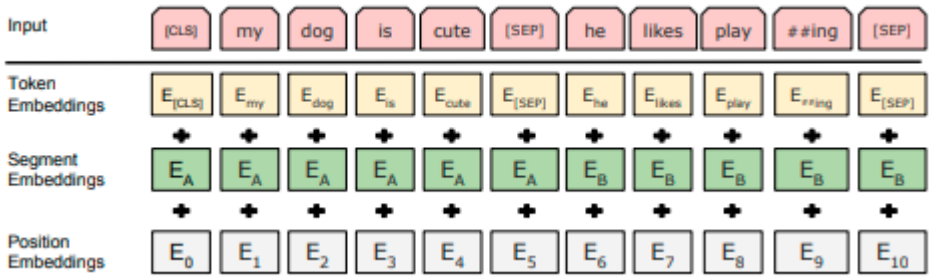
Hình 2.9: Masked Language Model

Hàm loss function của BERT sẽ bỏ qua mất mát của những từ không bị che dấu và chỉ đưa vào mất mát của những từ bị che dấu. Do đó mô hình sẽ hội tụ lâu hơn nhưng đây là đặc tính bù trừ cho sự gia tăng quan hệ về ngữ cảnh của câu. Việc lựa chọn ngẫu nhiên 15% số lượng các từ bị che dấu cũng tạo ra vô số các kịch bản đầu vào cho mô hình huấn luyện nên mô hình sẽ cần phải huấn luyện rất lâu mới học được toàn diện các khả năng. (Bert explained)

Task#2: BERT cho nhiệm vụ dữ đoán câu tiếp theo - Next sententce predict (NSP) [4]:

Trong quá trình đào tạo, 50% đầu vào là một cặp, trong đó câu thứ hai là câu tiếp theo trong tài liệu gốc, trong khi 50% còn lại, một câu ngẫu nhiên từ ngữ liệu được chọn làm câu thứ hai. Giả định là câu ngẫu nhiên

sẽ bị ngắt kết nối với câu đầu tiên. Để giúp mô hình phân biệt giữa hai câu trong đào tạo, đầu vào được xử lý theo cách sau trước khi đưa vào mô hình.



Hình 2.10: Biểu diễn đầu vào mô hình BERT

- BERT sử dụng WordPiece [16] để tách các câu thành các từ nhỏ với bộ từ điển bao gồm 32000 từ đối với tiếng Anh và hơn 108000 từ đối với mô hình đa ngôn ngữ.
- Mã thông báo [CLS] được chèn vào đầu câu đầu tiên và mã thông báo [SEP] được chèn vào cuối mỗi câu.
- Một câu nhúng cho biết Câu A hoặc Câu B được thêm vào mỗi mã thông báo.
- Một nhúng vị trí được thêm vào mỗi mã thông báo để chỉ ra vị trí của nó trong chuỗi. Khái niệm và cách thực hiện nhúng vị trí được trình bày trong bài báo Transformer.

Để dự đoán xem câu thứ hai có thực sự được kết nối với câu đầu tiên hay không, các bước sau được thực hiện:

- Toàn bộ chuỗi đầu vào đi qua mô hình transformer encoder.
- Đầu ra của mã thông báo [CLS] được chuyển thành vectơ có hình dạng 2×1, sử dụng lớp phân loại đơn giản (ma trận trọng số và độ lệch đã học).
- Tính xác suất của IsNextSequence với softmax.

Khi đào tạo mô hình BERT, Masked LM và NSP được đào tạo cùng nhau, với mục tiêu giảm thiểu hàm tổn thất kết hợp của hai chiến lược.

2.2.2 Tinh chỉnh mô hình BERT - Fine-tuning BERT

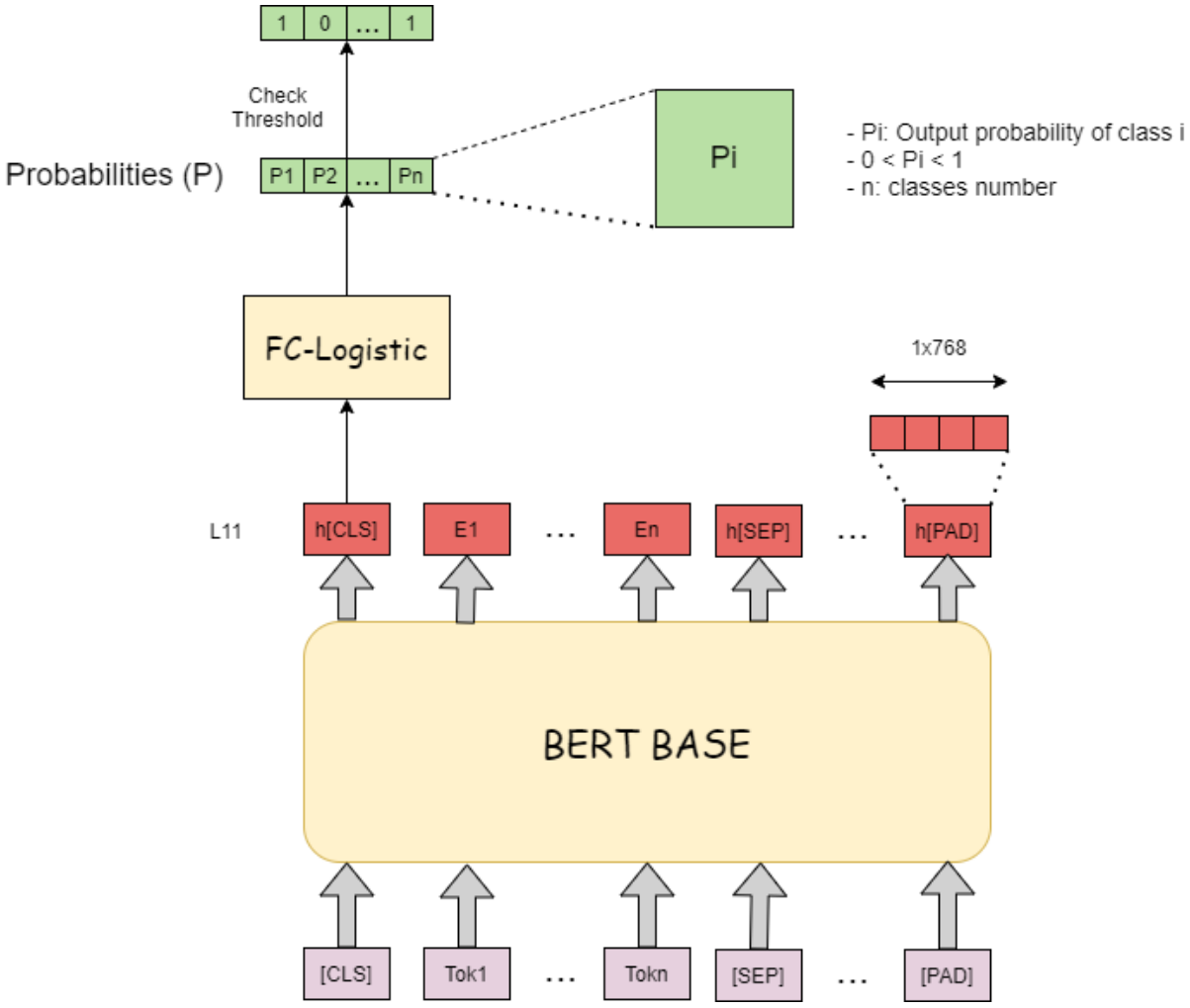
Việc tinh chỉnh rất đơn giản vì cơ chế tự tích chú ý trong transformer cho phép BERT mô hình hóa nhiều tác vụ con cụ thể cho dù chúng liên quan đến văn bản đơn lẻ hay cặp văn bản. Đối với mỗi tác vụ, chúng ta chỉ cần truyền các đầu vào và đầu ra của nhiệm vụ cụ thể vào BERT và tinh chỉnh tất cả các tham số từ đầu đến cuối.

2.3 Mô hình BERT cho bài toán phân tích khía cạnh và phân tích cảm xúc

2.3.1 Mô hình BERT cho bài toán phân loại đa nhãn

Trong bài toán phân tích khía cạnh, cảm xúc của các đánh giá, bình luận người dùng, như đã phân tích thì một câu bình luận sẽ có nhiều ý nghĩa như nhắc đến nhiều thực thể hoặc mang cùng lúc nhiều cảm xúc như tích cực, tiêu cực hoặc trung tính. Đây là bài toán phân loại đa nhãn và đa lớp. Mô hình BERT đã được áp dụng cho bài toán này và đã đạt được kết quả vượt trội so với các mô hình máy học và mô hình học sâu khác, kết quả được Ngoc Lê và cộng sự công bố [9] vào năm 2020.

Mô hình BERT áp dụng cho bài toán như sau:



Hình 2.11: Mô hình BERT cho bài toán phân loại đa nhãn

Tuy nhiên đối với bài toán đa nhãn sẽ cần một ngưỡng - threshold để xác định xem câu đánh giá thuộc nhãn nào. [9]

Tập dữ liệu	Mô hình phân tích khía cạnh	Mô hình phân tích cảm xúc
Nhà hàng	0.65	0.55
Khách sạn	0.25	0.45

Bảng 2.1: Bảng chọn ngưỡng phân lớp

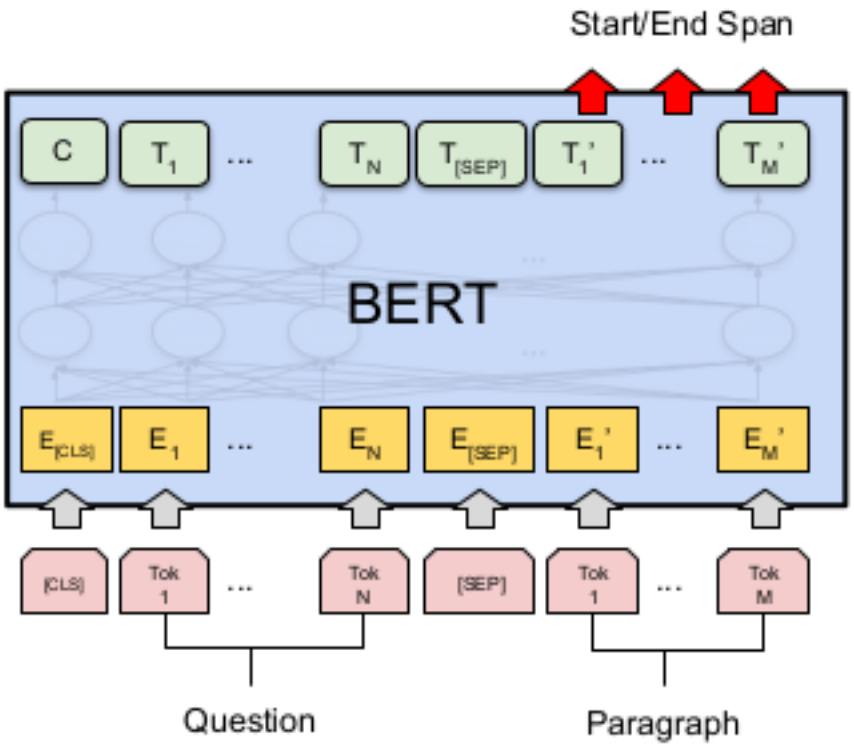
Như vậy có một nhược điểm là với tập dữ liệu khác nhau thì cần chọn ngưỡng khác nhau để cho bài toán có kết quả tốt. Việc chọn ngưỡng là chưa có phương pháp nào có thể tối ưu ngoài việc chọn ngẫu nhiên và thử kết quả. Do đó tác giả đưa ra một ý tưởng chuyển đổi bài toán đa nhãn sang bài toán đa lớp bằng cách giải quyết bài toán phân lớp với ý tưởng

của bài toán Question Answering - QnA. Mô hình và ý tưởng bài toán được trình bày chi tiết trong phần tiếp theo.

2.3.2 Giải quyết bài toán phân loại đa nhãn với Mô hình BERT - QnA

Bài toán QnA là bài toán thuộc lớp bài toán hiểu ngữ nghĩa của câu văn và trích xuất thông tin phù hợp với yêu cầu. Mô hình BERT đang cho thấy hiệu quả tuyệt vời của nó trong bài toán QnA này.

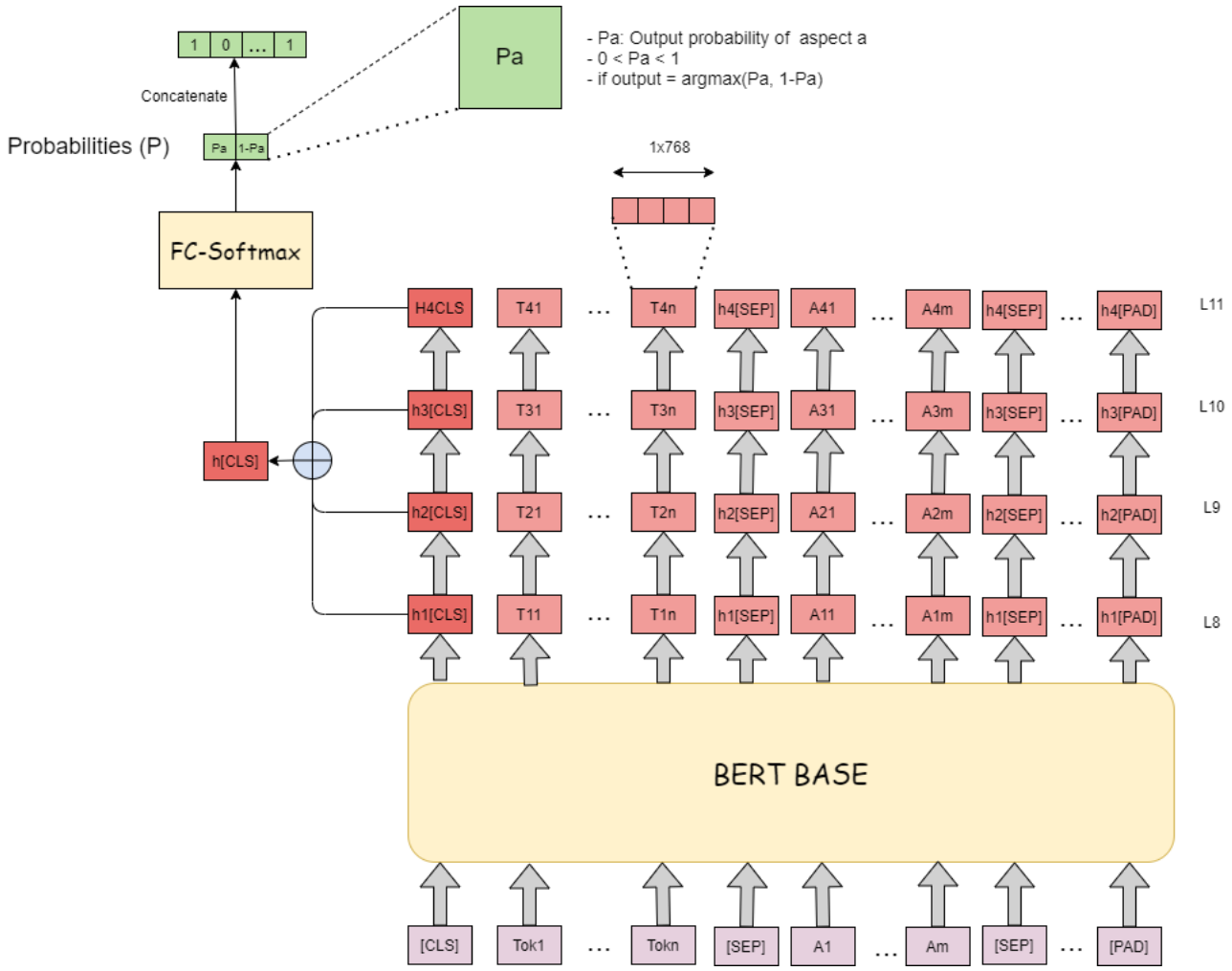
Mô hình BERT cho bài toán QnA như sau [4]:



Hình 2.12: Mô hình BERT cho bài toán QnA

Dựa trên những nghiên cứu về bài toán QnA và nắm bắt nhược điểm của bài toán phân loại đa nhãn nêu trên tác giả đã đưa ra ý tưởng chuyển đổi bài toán đa nhãn sang bài toán đa lớp dựa trên ý tưởng giải quyết bài toán QnA sử dụng mô hình BERT.

Kiến trúc mô hình được thiết kế như sau:



Hình 2.13: Mô hình BERT cho bài toán phân loại đa lớp với ý tưởng bài toán QnA

Đầu ra của mô hình BERT thay vì chỉ sử dụng vector embedding của token [CLS] ở layer cuối cùng rồi đem qua mô hình phân lớp như trong bài báo Ngoc Lê [9] tác giả quyết định ghép vector embedding của token [CLS] ở bốn layer cuối cùng lại rồi mới đưa qua mô hình phân lớp, trong quá trình thử nghiệm kết quả của việc này cho kết quả tương đối khả thi.

Đối với hai tập dữ liệu khách sạn và nhà hàng, các câu đánh giá đều là tiếng việt trong khi đó các khía cạnh là cặp ghép của thực thể và thuộc tính đều là tiếng anh. Do đó tác giả tiến hành dịch các khía cạnh sang tiếng việt để khớp với câu đánh giá, điều này cho thấy kết quả mô hình được cải thiện nhưng không nhiều.

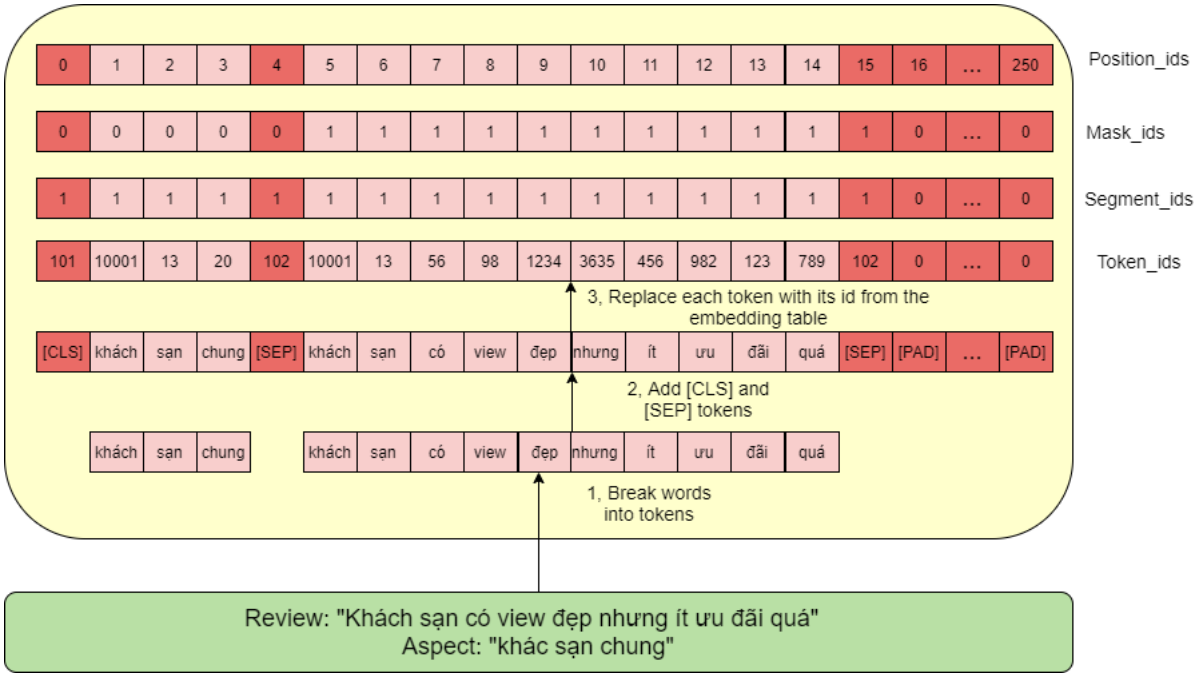
Ví dụ thiết kế đầu vào tương tự đầu vào của bài toán QnA:

- RV: "Khách sạn có view đẹp nhưng ít ưu đãi quá"
- Asp: "HOTEL#GENERAL"

Khi đó RV và Aps được tách thành các token bởi Tokenization của BERT, cặp RV và Aps được biểu diễn làm đầu vào như sau (giả sử các từ sau khi tách thành token vẫn không có thay đổi):

[CLS] Khách sạn có view đẹp nhưng ít ưu đãi quá [SEP] khách sạn chung [SEP]

Đầu vào mô hình được biểu diễn như sau:



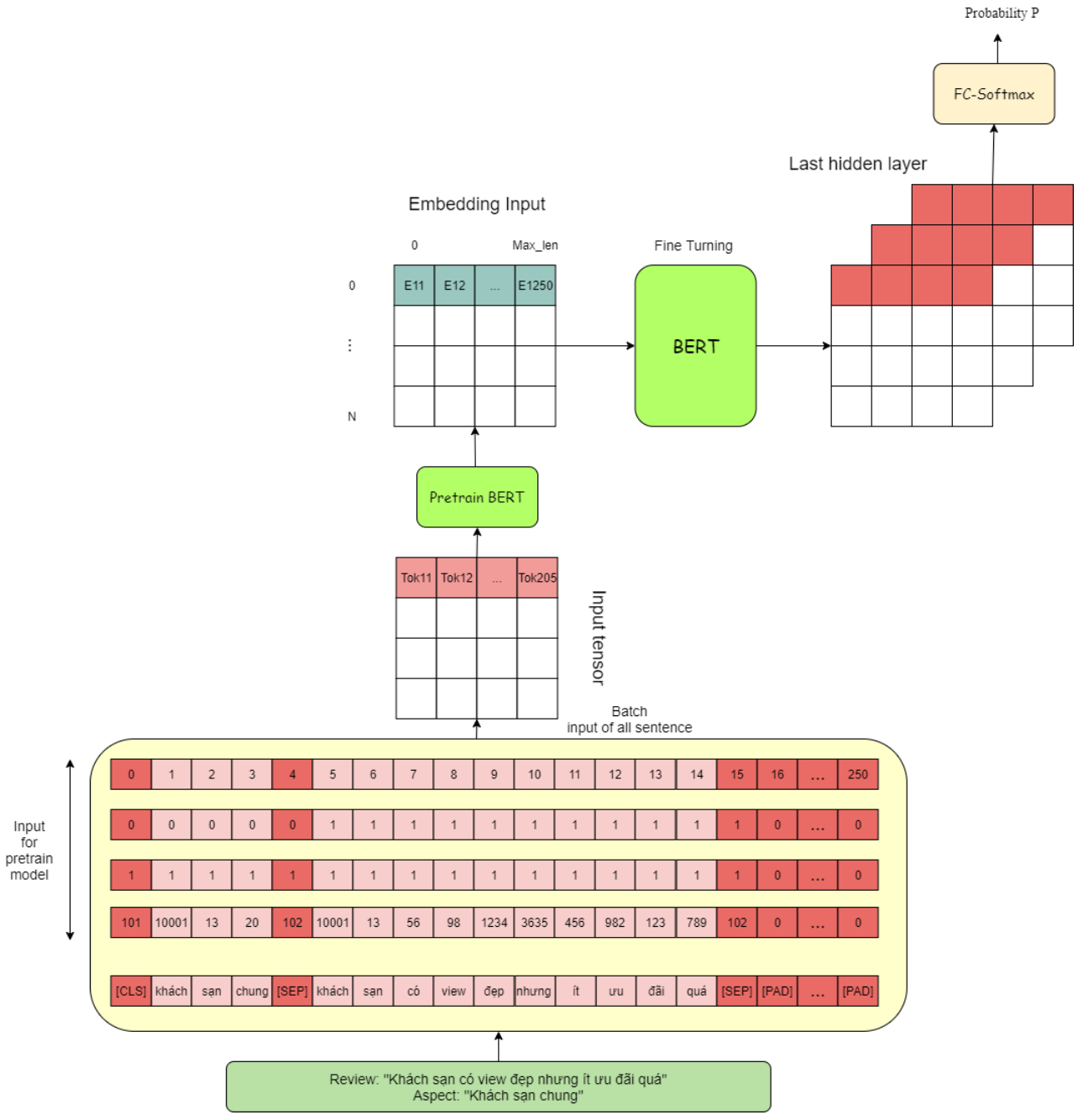
Hình 2.14: Đầu vào mô hình BERT với ý tưởng bài toán QnA

Trong đó:

- Token_ids: Vector id của danh sách token đầu vào theo trình tự được ánh xạ vào bộ từ điển của BERT
- Segment_ids: Vector id phân đoạn để phân biệt câu thứ nhất và thứ hai trong cặp đầu vào
- Mask_ids: Vector id phân đoạn phân biệt các token đầu vào với token padding được thêm vào.

- Postion_ids: Vector id vị trí của token trong position embedding.

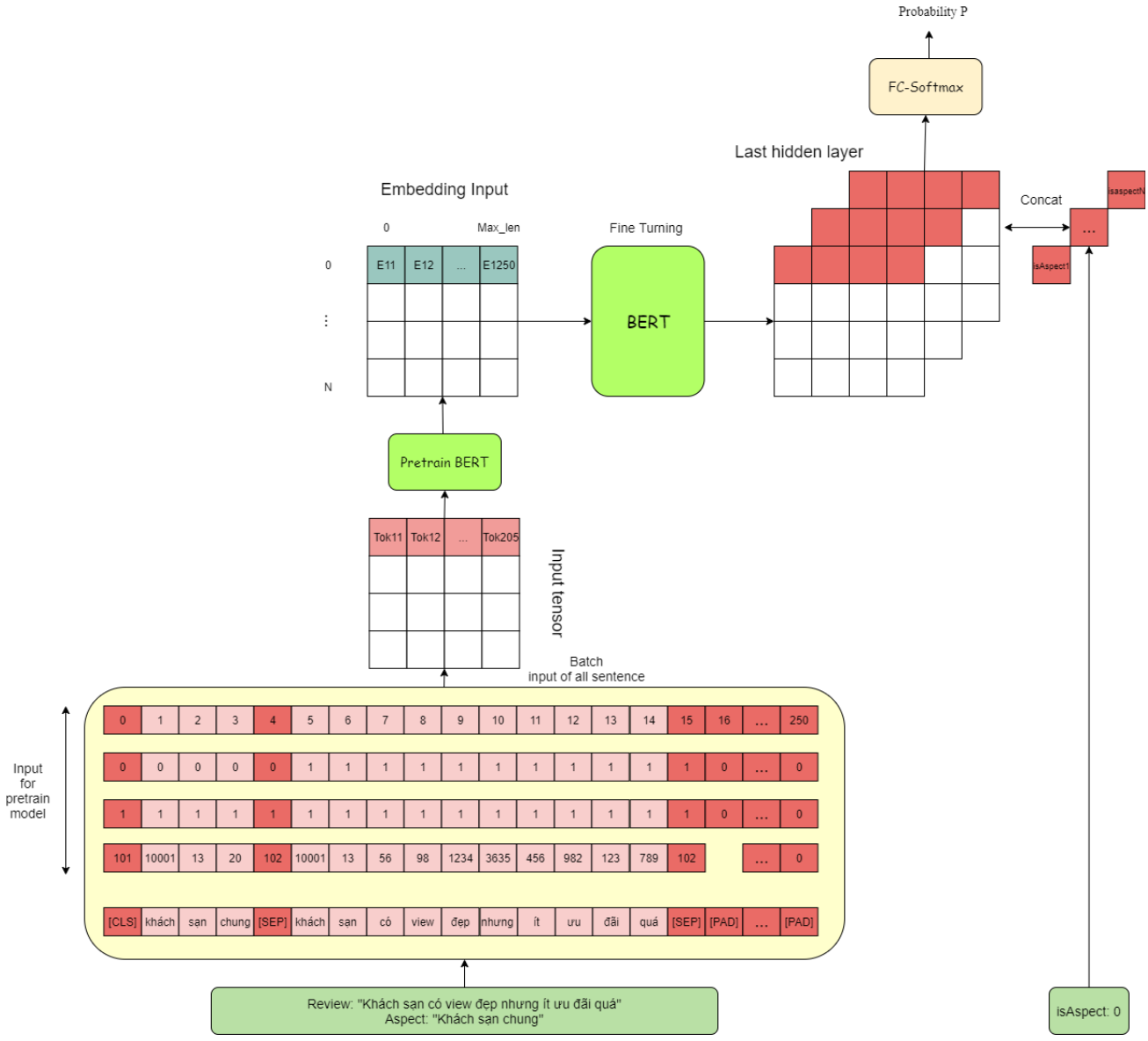
Mô hình BERT chi tiết cho bài toán phân tích khía cạnh như sau:



Hình 2.15: Mô hình bài toán phân tích khía cạnh

Kiến trúc mô hình bài toán phân tích cảm xúc tương tự bài toán phân tích khía cạnh tuy nhiên đầu vào sẽ có thêm feature về khía cạnh (đầu ra của bài toán phân tích khía cạnh) và trước khi đi qua lớp FC-Softmax tác giả nối thêm feature về khía cạnh vào đầu ra của BERT. Mô hình chi tiết

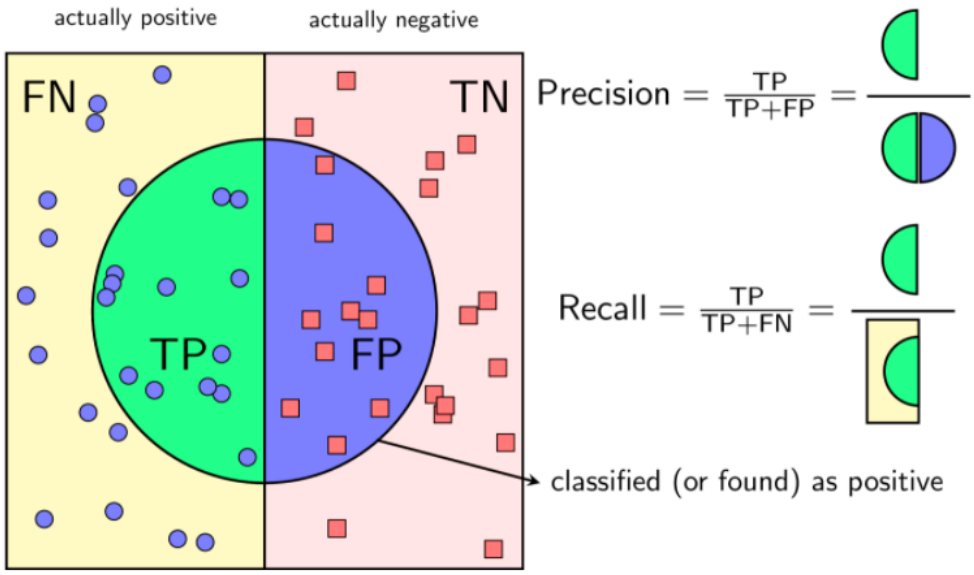
cho bài toán phân tích cảm xúc như sau:



Hình 2.16: Mô hình bài toán phân tích cảm xúc

2.4 Phương pháp đánh giá

Đối với bài toán phân lớp, mô hình được đánh giá dựa trên F1-score. Chỉ số này đảm bảo được sự cân bằng về tính chính xác trong dự đoán của mô hình và thường được dùng để đo khả năng dự đoán đúng của các bộ dữ liệu có sự mất cân bằng nhãn lớn. Giá trị F1-score được tính toán dựa trên ma trận nhầm lẫn (Confusion matrix).



Hình 2.17: Cách tính precision và recall

Hình 2.17 mô tả rất rõ ràng cách thức tính toán confusion matrix . Giả sử mô hình đang cố gắng dự đoán các thực thể có liên quan đến một lớp A nào đó, khi đó selected elements là các phần tử được mô hình dự đoán rằng có liên quan đến lớp A và relevant elements là các phần tử thực sự có liên quan đến lớp ta đang xét. Ta có bốn giá trị cơ bản lần lượt là true positive (TP), false positive (FP), true negative (TN) và false negative (FN).

Precision: được định nghĩa là tỉ lệ số điểm true positive trong số những điểm được phân loại là positive (TP + FP).

$$Precision = \frac{TP}{TP + FP}$$

Recall: được định nghĩa là tỉ lệ số điểm true positive trong số những điểm thực sự là positive (TP + FN).

$$Precision = \frac{TP}{TP + FN}$$

F1-score: là harmonic mean của precision và recall (giả sử rằng hai đại lượng này khác không):

$$\frac{2}{F1} = \frac{1}{Precision} + \frac{1}{Recall} = 2\frac{Precision.Recall}{Precision + Recall}$$

Trong trường hợp tổng quát F_β được xác định bằng công thức sau:

$$F_\beta = (1 + \beta^2)\frac{Precision.Recall}{\beta^2(Precision + Recall)}$$

Đối với các bài toán đa lớp hoặc đa nhãn chúng ta sẽ quan tâm đến các độ đo micro-average và macro-average.

Micro-average Precision:

$$Micro - averagePrecision = \frac{\sum_{c=1}^{C} TP_c}{\sum_{c=1}^{C}(TP_c + FP_c)}$$

Micro-average Recall:

$$Micro - averageRecall = \frac{\sum_{c=1}^{C} TP_c}{\sum_{c=1}^{C}(TP_c + FN_c)}$$

Macro-average Precision:

$$Macro - averagePrecision = \frac{\sum_{c=1}^{C} Precision_c}{C}$$

Macro-average Recall:

$$Macro - averageRecall = \frac{\sum_{c=1}^{C} Recall_c}{C}$$

Chương 3

Kết quả

3.1 Một số nghiên cứu thực nghiệm

Tác giả đã thử nghiệm một số thực nghiệm cần thiết để giải quyết bài toán và cải thiện kết quả

Đầu tiên tác giả tiến hành khai thác thông tin tổng quan của dữ liệu. Với nhãn cảm xúc có thể thấy tỷ lệ nhãn giữa các tập đào tạo, tập kiểm tra khớp và tập kiểm tra là cùng phân phối.

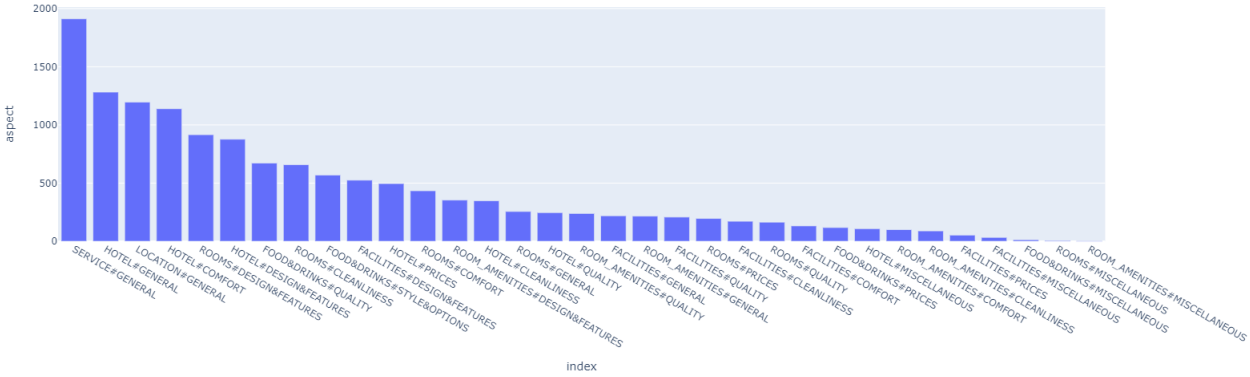
Link trực quan hóa nhãn cảm xúc với tập khách sạn: (visualize sentiment hotel)

Hình 3.1: Phân phối nhãn cảm xúc trên ba tập dữ liệu về khách sạn

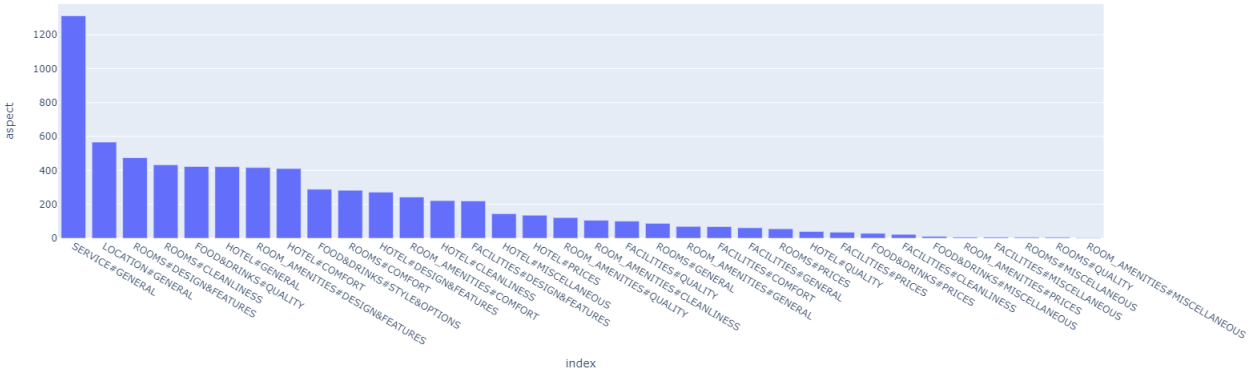
Link trực quan hóa nhãn cảm xúc với tập dữ liệu nhà hàng: (visualize sentiment restaurant)

Hình 3.2: Phân phối nhãn cảm xúc trên ba tập dữ liệu về nhà hàng

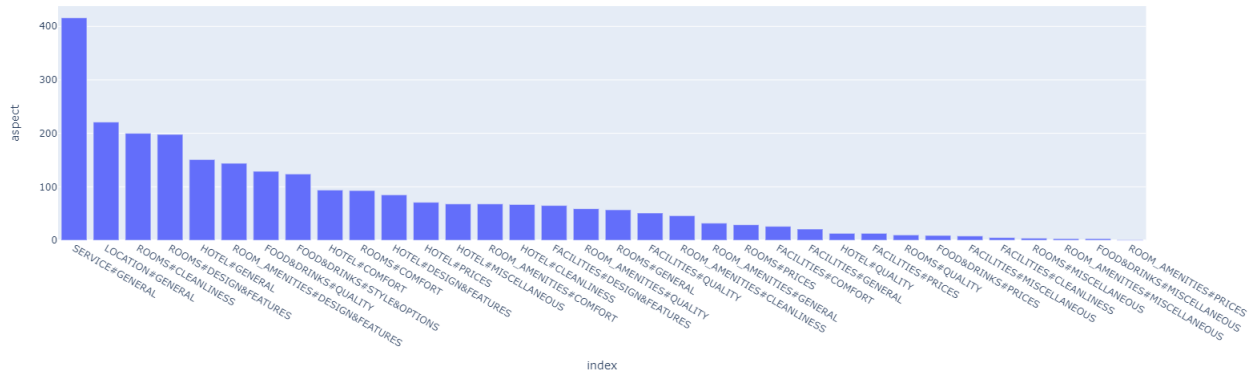
Đối với nhãn khía cạnh có thể thấy dữ liệu bị mất cân bằng nhãn rất nhiều, song phân bố dữ liệu nhãn khía cạnh của các tập đào tạo, tập kiểm tra khớp, tập kiểm tra không có sự đồng đều ở các nhãn. Có những nhãn có trong tập đào tạo nhưng không có trong tập kiểm tra hoặc ngược lại dẫn đến sự khó khăn trong xây dựng mô hình. Dữ liệu thiếu hụt cũng là một nguyên nhân rất lớn ảnh hưởng đến mô hình:



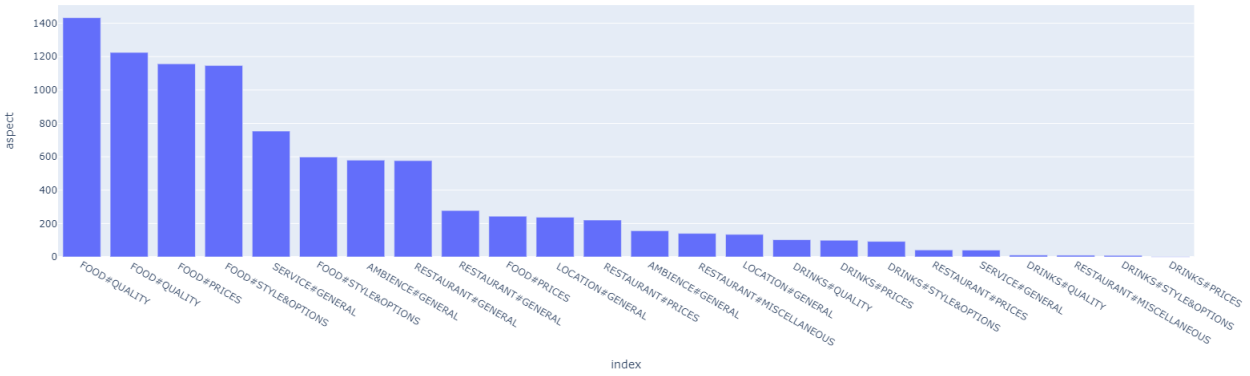
Hình 3.3: Phân phối nhãn khía cạnh tập đào tạo về khách sạn



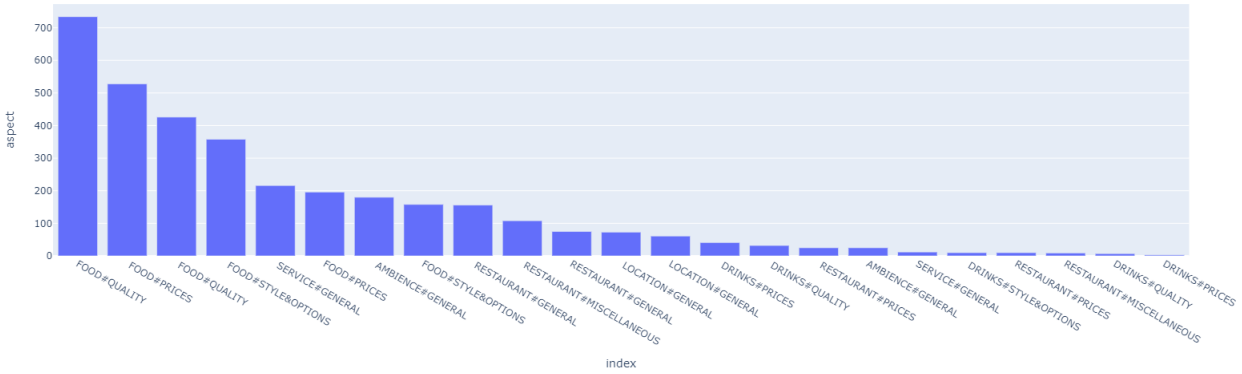
Hình 3.4: Phân phối nhãn khía cạnh tập kiểm tra khớp về khách sạn



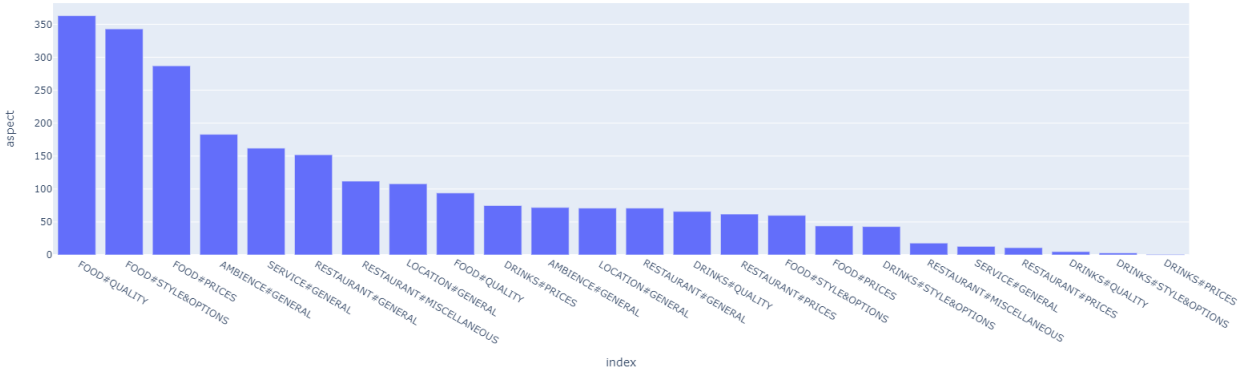
Hình 3.5: Phân phối nhãn khía cạnh tập kiểm tra về khách sạn



Hình 3.6: Phân phối nhãn khía cạnh tập đào tạo về nhà hàng



Hình 3.7: Phân phối nhãn khía cạnh tập kiểm tra khớp về nhà hàng



Hình 3.8: Phân phối nhãn khía cạnh tập kiểm tra về nhà hàng

Ngoài một số vấn đề về thiếu hụt dữ liệu và mất cân bằng nhãn còn tồn tại vấn đề về định dạng chữ không giống nhau, do đó tác giả tiến hành chuẩn hóa dữ liệu bằng cách chuyển toàn bộ dữ liệu về dạng unicode dựng sẵn sử dụng thư viện ftfy.

Đồng thời tác giả cài đặt cứng max-length của câu đầu vào (vì trong quá trình khảo sát chiều dài lớn nhất của các RV là 240 và độ dài của các khía cạnh luôn nhỏ hơn 10 do đó max-length = 250

Dưới đây là chi tiết siêu tham số - hyperparameter trong quá trình xử lý mô hình:

STT	Tên mô hình	Thông số (Siêu tham số)	Tiêu chí đánh giá	Chú giải
1	Bert	1. batch_size =8 2. lr = 1e-5 3. optimize = Adam 4. use_pooler = True 5. epoch =10	Kết quả chưa được tốt => có thể do hàm tối ưu chưa ổn	F1 = 0.6427; Recall = 0.651; Precision =0.6348
2	Bert	1. batch_size =8 2. lr = 1e-5 3. optimize = AdamW 4. use_pooler = True 5. epoch =10	Thử với AdamW vì trong bài báo gốc về BERT đề xuất => kết quả cao hơn đáng kể, tuy nhiên vẫn chưa tốt so với baseline	F1 = 0.7613; Recall = 0.7584; Precision =0.7645
3	Bert	1. batch_size =8 2. lr = 1e-5 3. optimzer = AdamW 4. use_pooler = True 5. epoch = 10 6. num_warmup_step = 100 7. num_training_steps = 5000	Sử dụng thêm num_warmup_step để set scheduler cho lr => kết quả tốt hơn	F1 = 0.7708; Recall = 0.7923; Precision =0.7503
4	Bert	1. batch_size =8 2. lr = 1e-5 3. optimzer = AdamW 4. epoch = 10 6. num_warmup_step = 100 7. num_training_steps = 5000 8. class_weight = True 9. use_pooler = True	Thêm class_weight cho quá trình tính loss mục đích giải quyết vấn đề mất cân bằng nhãn => kết quả không mấy thay đổi	F1 = 0.771; Recall = 0.7851 Precision =0.7577

STT	Tên mô hình	Thông số (Siêu tham số)	Tiêu chí đánh giá	Chú giải
5	Bert	1. batch_size =8 2. lr = 1e-5 3. optimzer = AdamW 4. epoch = 10 6. num_warmup_step = 100 7. num_training_steps = 5000 8. class_weight =True 9. use_pooler = False	Thực hiện ghép vector embedding của token [CLS] ở 4 layer cuối cùng => kết quả có cải tiến vượt trội	F1 = 0.7858; Recall = 0.7862; Precision =0.7855
6	Bert	1. batch_size =8 2. lr = 1e-5 3. optimzer = AdamW 4. epoch = 10 6. num_warmup_step = 100 7. num_training_steps = 5000 8. class_weight =True 9. use_pooler = False 10. sử dụng thư viện fix_tex để chuyển RV về unicode dựng sẵn	Thực hiện chuẩn hóa dữ liệu về dạng unicode dựng sẵn => kết quả có sự cải thiện	F1 = 79.2; Recall = 77.92; Precision =80.54

Hình 3.9: Chi tiết các siêu tham số của mô hình trong quá trình cái tiến mô hình

Nhận xét: Đây là thử nghiệm trên bài toán phân tích khía cạnh kết qủa đánh giá dựa trên tập kiểm tra khớp của bộ dữ liệu khách sạn. Kết quả cho thấy việc sử dụng ghép bốn vector embedding của token [CLS] ở bốn layer cuối và sử dụng thư viện chuẩn hóa dữ liệu sang kiểu unicode dựng sẵn cho kết quả tốt đáng kể.

3.2 Kết quả của đồ án

Dưới đây là bảng so sánh kết quả thực nghiệm được công bố trước đó và kết quả nghiên cứu của tác giả:

Domain	Team	Precision	Recall	F1-Score
Restaurant	The 3rd team [1]			
	The 2nd team [11]	0.78	0.65	0.71
	The 1st team [5]	0.75	0.85	0.79
	Dang [12]			
	Ngoc C.Lê [9]	0.7684	0.8806	0.8207
	Kết quả NC	0.82	0.88	0.8489
Hotel	The 3rd team [1]			
	The 2nd team [11]	0.83	0.51	0.63
	The 1st team [5]	0.75	0.64	0.69
	Dang [12]			
	Ngoc C.Lê [9]	0.7940	0.7874	0.7907
	Kết quả NC	0.8200	0.7500	0.7834

Bảng 3.1: Kết quả so sánh hiệu xuất nghiên cứu của tác giả với một số nghiên cứu khác trên tập DEV - bài toán phân tích khía cạnh

Domain	Team	Precision	Recall	F1-Score
Restaurant	The 3rd team	0.88	0.38	0.54
	The 2nd team [11]	0.62	0.62	0.62
	The 1st team [5]	0.79	0.76	0.77
	Dang [12]	0.8475	0.7648	0.8040
	Ngoc C.Lê [9]	0.7916	0.8367	0.8135
	Kết quả NC	0.81	0.82	0.8149
Hotel	The 3rd team	0.85	0.42	0.56
	The 2nd team [11]	0.83	0.58	0.68
	The 1st team [5]	0.76	0.66	0.7
	Dang [12]	0.8235	0.5975	0.6925
	Ngoc C.Lê [9]	0.7960	0.7972	0.7966
	Kết quả NC	0.8054	0.7792	0.792

Bảng 3.2: Kết quả so sánh hiệu xuất nghiên cứu của tác giả với một số nghiên cứu khác trên tập TEST - bài toán phân tích khía cạnh

Domain	Team	Precision	Recall	F1-Score
Restaurant	The 3rd team			0.59
	The 2nd team [11]	0.71	0.59	0.64
	The 1st team [5]	0.63	0.71	0.67
	Dang [12]			
	Ngoc C.Lê [9]	0.6616	0.7145	0.6871
	Kết quả NC	0.68	0.72	0.6994
Hotel	The 3rd team [1]			0.59
	The 2nd team [11]	0.78	0.48	0.6
	The 1st team [5]	0.67	0.58	0.62
	Dang [12]			
	Ngoc C.Lê [9]	0.7943	0.5901	0.6772
	Kết quả NC	0.80	0.61	0.69

Bảng 3.3: Kết quả so sánh hiệu xuất nghiên cứu của tác giả với một số nghiên cứu khác trên tập DEV - bài toán phân tích cảm xúc

Domain	Team	Precision	Recall	F1-Score
Restaurant	The 3rd team	0.79	0.35	0.48
	The 2nd team [11]	0.52	0.52	0.52
	The 1st team [5]	0.62	0.6	0.61
	Dang [12]			
	Ngoc C.Lê [9]	0.6327	0.6503	0.6414
	Kết quả NC [9]	0.6478	0.6624	65.5
Hotel	The 3rd team	0.8	0.39	0.53
	The 2nd team [11]	0.71	0.49	0.58
	The 1st team [5]	0.66	0.57	0.61
	Dang [12]			
	Ngoc C.Lê [9]	0.7983	0.5882	0.6774
	Kết quả NC	0.8009	0.5842	0.6755

Bảng 3.4: Kết quả so sánh hiệu xuất nghiên cứu của tác giả với một số nghiên cứu khác trên tập TEST - bài toán phân tích cảm xúc

Nhận xét: Từ kết quả được trình bày ở bảng trên có thể cho thấy nghiên cứu của tác giả vẫn giữ được kết quả tương đương với nghiên cứu trước đó cũng sử dụng phương pháp học chuyển tiếp mà không cần dựa vào tính ngẫu nhiên của ngưỡng. Bài toán được đưa về bài toán đơn giản phổ quát

hơn.

Ngoài ra, về mặt dữ liệu có thể thấy có một số nhãn khía cạnh xuất hiện rất ít trong bộ dữ liệu, có những nhãn chỉ xuất hiện trên tập kiểm tra mà không có trên tập đào tạo. Do đó tác giả đề xuất giữ lại bốn nhãn khía cạnh có nhiều dữ liệu (có thể trong thực tế là bốn khía cạnh mà cả doanh nghiệp lẫn người dùng quan tâm nhất) trong tập đào tạo và mở rộng thêm nhãn **Khác** để chuyển các dữ liệu có nhãn không nằm trong bốn nhãn chính kia thành nhãn **Khác**. Dưới đây là kết quả thử nghiệm:

Domain	Team	Precision	Recall	F1-Score
Restaurant	Phân tích khía cạnh	0.8123	0.8765	0.8431
	Phân tích cảm xúc	0.692	0.7205	0.7059
Hotel	Phân tích khía cạnh	0.8156	0.7498	0.7813
	Phân tích cảm xúc	0.8152	0.6077	0.6963

Bảng 3.5: Kết quả trên tập DEV

Domain	Team	Precision	Recall	F1-Score
Restaurant	Phân tích khía cạnh	0.8006	0.8189	0.8096
	Phân tích cảm xúc	0.6456	0.6534	.6495
Hotel	Phân tích khía cạnh	0.7922	0.781	0.7856
	Phân tích cảm xúc	0.8112	0.5820	0.6777

Bảng 3.6: Kết quả trên tập TEST

3.3 Nhược điểm và hướng phát triển của đồ án trong tương lai

3.3.1 Nhược điểm

Trong qua trình nghiên cứu bài toán, tác giả nhận thấy một số nhược điểm như sau:

- Với bài toán xác định khía cạnh bị giới hạn trong một lĩnh vực nhỏ, không thể tổng quát cho nhiều lĩnh vực.
- Với ý tưởng QnA cho bài toán tuy giải quyết được vấn đề về chọn

ngưỡng ngẫu nhiên như một số nghiên cứu về bài toán đa nhãn khác nhưng với ý tưởng này có nhược điểm về việc tăng số lượng dữ liệu, làm tốn bộ nhớ, tốn kém tính toán.

3.3.2 Hướng phát triển trong tương lai

Do hạn chế về mặt thời gian tác giả chưa thể giải quyết hết nhược điểm nêu trên của bài toán do đó tác giả có đề xuất một số hướng phát triển cho tương lai như sau:

- Mở rộng từ bài toán phân tích khía cạnh thông thường sang bài toán trích xuất khía cạnh như vậy có thể áp dụng trên nhiều lĩnh vực.
- Xử lý vấn đề gia tăng dữ liệu bằng cách lọc bỏ dữ liệu từ bên phân tích khía cạnh.

Kết luận

Trong nghiên cứu đồ án tốt nghiệp với đề tài **"Ứng dụng học chuyển tiếp trong bài toán phân tích khía cạnh - cảm xúc tiếng việt"** cơ bản đã hoàn thành các mục tiêu đề ra gồm:

- Tìm hiểu và bổ sung các kiến thức cơ sở cho bài toán phân tích khía cạnh, cảm xúc cho văn bản tiếng việt.
- Tìm hiểu các mô hình và phương pháp học chuyển tiếp áp dụng cho bài toán và nghiên cứu kỹ lưỡng các công trình liên quan để đưa ra giải pháp giải quyết nhược điểm của các giải pháp được công bố trước đó.
- Nghiên cứu và áp dụng các phương pháp trên vào bài toán phân tích khía cạnh, cảm xúc cho tiếng việt cụ thể là trong lĩnh vực nhà hàng, khách sạn.
- Thử nghiệm hiệu quả ý tưởng xử lý bài toán đa nhãn sang bài toán đa lớp đơn giản hơn, giải quyết tốt vấn đề chọn ngưỡng ngẫu nhiên mà vẫn giữ được kết quả tương đương với mô hình bài toán đa nhãn ban đầu.

Cuối cùng, tác giả đề xuất một số hướng giải quyết tiềm năng cho bài toán:

- Mở rộng từ bài toán phân tích khía cạnh thông thường sang bài toán trích xuất khía cạnh như vậy có thể áp dụng trên nhiều lĩnh vực.
- Xử lý vấn đề gia tăng dữ liệu bằng cách lọc bỏ dữ liệu từ bên phân tích khía cạnh.

Tài liệu tham khảo

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan Gomez, Lukasz Kaiser & Illia Polosukhin, "Attention Is All You Need", 2017.

[2] Dzmitry Bahdanau, Kyunghyun Cho & Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate", *ArXiv*, Vol 1409, 2014.

[3] J. H. Martin D. Jurafsky, "Chapter 4 Naive Bayes and Sentiment Classification - T. C. a. N. Bayes, Stanford", 2020.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee & Kristina Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", *CoRR*, Vol abs/1810.04805, 2018.

[5] K. V. Nguyen T. D. Van & N. L.-T. Nguyen, "NLP@UIT at VLSP 2018: A Supervised Method for Aspect Based Sentiment Analysis", 2018.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren & Jian Sun, "Deep Residual Learning for Image Recognition", 2016, 770–778.

[7] Minh-Thang Luong, Hieu Pham & Christopher Manning, "Effective Approaches to Attention-based Neural Machine Translation", 2015.

[8] Minh-Tien Nguyen, Anh Phan, Linh Le, Nguyen Son, Dung Le, Miku Hirano & Hajime Hotta, *Transfer Learning for Information Extraction with Limited Data*, 2020.

[9] Ngoc Lê, Nguyen Lam, Son Nguyen & Duc Nguyen, "On Vietnamese Sentiment Analysis: A Transfer Learning Method", 2020, 1–5.

[10] Pramod Mathapati, Arati Shahapurkar & Kavita Hanabaratti, "Sentiment Analysis using Naïve bayes Algorithm", *International Journal of Computer Sciences and Engineering*, Vol 5, 2017, 75–77.

[11] T. A. Nguyen & P. Q. N. Minh, "Using Multilayer Perceptron for Aspect-based Sentiment Analysis at VLSP-2018 SA Task", 2018.

[12] Thin Dang, Vu Duc, Kiet Nguyen & Ngan Nguyen, "Deep Learning for Aspect Detection on Vietnamese Reviews", 2018, 104–109.

[13] Thomas Burr, "Pattern Recognition and Machine Learning. Christopher M. Bishop", *Journal of the American Statistical Association*, Vol 103, 2008, 886–887.

[14] Xiao-Xia Yin, Sillas Hadjiloucas & Yanchun Zhang, *Pattern Classification*, 2017.