# Sử dụng các Mô hình Ngôn Ngữ Lớn để giải quyết bài toán Aspect-Based Sentiment Analysis và Giải thích kết quả bằng SHAP

Dinh Bao[1,2], Quang Le[1,2], Bac Le[1,2]

[1]Faculty of Information Technology, University of Science, Ho Chi Minh City, Viet Nam.
[2]Vietnam National University, Ho Chi Minh City, Vietnam.

Contributing authors: 19120173@student.hcmus.edu.vn; 19120121@student.hcmus.edu.vn; lhbac@fit.hcmus.edu.vn;

**Tóm tắt nội dung**

Phân-tích-cảm-xúc-theo-từng-khía-cạnh (ABSA) là một bài toán quan trọng, được nghiên cứu rộng rãi trong cả học thuật lẫn industry. BERT là một Mô Hình Ngôn Ngữ có thể tạo ra các vector biểu diễn từ giàu ngữ nghĩa. Nhiều phương pháp sử dụng BERT để giải quyết bài toán ABSA đã được đề xuất. Tuy nhiên, gần đây có nhiều mô hình ngôn ngữ khác đã outperform BERT ở nhiều bài toán NLP khác nhau, tiêu biểu là hai Mô Hình Ngôn Ngữ RoBERTa và DeBERTa. Trong bài nghiên cứu này, chúng tôi tinh chỉnh nhiều Mô Hình Ngôn Ngữ khác nhau để giải quyết bài toán ABSA và đạt được kết quả state-of-the-art trên bộ dữ liệu SemEval 2014-Task 4. Ngoài ra chúng tôi còn sử dụng SHAP để giải thích dự đoán của mô hình.

**Keywords:** ABSA, SHAP, BERT, RoBERTa, DeBERTa

## 1 Introduction

Phân-tích-cảm-xúc (SA) là một bài toán quan trọng trong lĩnh vực Xử lý Ngôn Ngữ Tự nhiên (NLP), được nghiên cứu và sử dụng rộng rãi trong cả học thuật lẫn industry. Mục tiêu của SA là phân tích ra cảm xúc chung của người dùng được biểu thị trong câu chữ về một thực thể xác định.

Tuy nhiên, một thực thể có nhiều khía cạnh khác nhau, và người dùng có thể bày tỏ cảm xúc khác nhau đối với từng khía cạnh. Ví dụ, với câu "Cuốn sách này đọc rất hay nhưng giá thì đắt quá", cảm xúc của người dùng về khía cạnh nội dung của cuốn sách là tích cực trong khi cảm xúc về khía cạnh giá cả của cuốn sách là tiêu cực. Phân-tích-cảm-xúc-dựa-trên-khía-cạnh (ABSA) [10] nhắm đến việc phân tích cảm xúc chính xác của người dùng đối với từng khía cạnh của thực thể được đề cập đến trong câu. Bài toán ABSA được chia thành 4 bài toán con lần lượt là Aspect Term Extraction (ATE), Aspect Term Polarity (ATP), Aspect Category Detection (ACD) và Aspect Category Polarity (ACP). Trong bài nghiên cứu này, chúng tôi tập trung vào hai bài toán cuối là Aspect Category Detection và Aspect Category Polarity.

Aspect Category Detection (ACD): Cho trước một tập hợp cố định các khía cạnh (VD: food, price), mục tiêu bài toán là tìm ra toàn bộ các khía cạnh được đề cập đến trong một câu cho trước. Các khía cạnh này có thể không được đề cập đến như là một từ trong câu. VD: cho trước tập hợp cố đinh các khía cạnh là {food, service, price, ambience, anecdotes/miscellaneous}, ta có nếu với câu "The restaurant was too expensive" thì kết quả trả về là {price}, còn với câu "The restaurant was expensive, but the menu was great" thì kết quả trả về là {price, food}.

Aspect Category Polarity (ACP): Cho trước một tập hợp các khía cạnh đã có trong câu, mục tiêu của bài toán này là tìm ra cảm xúc của người dùng đối với mỗi khía cạnh đó. Cảm xúc người dùng đối với mỗi khía cạnh có thể là positive, neutral, negative, conflict. VD: với câu "The restaurant was too expensive" và tập hợp khía cạnh đã có trong câu là {price} thì kết quả trả về là {price: negative}, còn với câu "The restaurant was expensive, but the menu was great" và tập hợp khía cạnh đã có trong câu là {price, food} thì kết quả trả về là {price: negative, food: positive}.

Các Mô hình Ngôn Ngữ Lớn, đặc biệt là BERT [2], đạt được kết quả state-of-the-art trong nhiều bài toán NLP, bao gồm text classification, reading comprehension, và name entity recognition. Nhiều nghiên cứu về việc sử dụng BERT để giải quyết ACD và ACP đã được thực hiện và thu được các kết quả rất tốt. Tuy nhiên gần đây, nhiều Mô Hình Ngôn Ngữ khác đã được ra đời, tiêu biểu RoBERTa [7] và DeBERTa [3], và đạt được kết quả tốt hơn BERT ở nhiều bài toán khác nhau. Do nhận thấy có thể tận dụng RoBERTa và DeBERTa để giải quyết ABSA nên, ở bài nghiên cứu này, chúng tôi sẽ sử dụng phương pháp Xây dựng câu phụ để chuyển đổi hai bài toán Aspect Category Detection và Aspect Category Polarity thành bài toán Sentence Classification, sau đó tinh chỉnh các Mô Hình Ngôn Ngữ Lớn khác nhau như BERT, RoBERTa và DeBERTa để giải quyết bài toán Sentence Classification, từ đó giải quyết hai bài toán ACD và ACP.

Các mô hình máy học hoạt động như một hộp đen. Tuy chúng cho ra kết quả rất tốt nhưng con người không hiểu được cách chúng hoạt động và ra quyết định. AI-có-thể-giải-thích-được (XAI) là tập hợp các phương pháp để tìm hiểu và giải thích dự đoán của mô hình máy học. Việc giải thích có hai tác dụng: (1) làm mô hình trở nên trong suốt hơn, giúp người dùng tin tưởng vào quyết định của mô hình và (2) giúp ta tìm ra các sự thiên vị (bias) ẩn giấu bên trong mô hình, cải thiện hiệu quả của mô hình. SHAP là một phương pháp nằm trong XAI. SHAP [8] dựa trên giá trị Shapley trong lý thuyết trò chơi để giải thích dự đoán của mô hình. Trong bài nghiên cứu này,

chúng tôi dùng SHAP để giải thích dự đoán của mô hình, giúp người đọc hiểu hơn về cách mô hình ra quyết định.

Tổng kết lại, các đóng góp của chúng tôi trong bài nghiên cứu này là:

- Sử dụng phương pháp Xây dựng câu phụ để chuyển đổi hai bài toán ACD và ACP thành bài toán Sentence Classification
- Tinh chỉnh nhiều Mô hình Ngôn Ngữ Lớn khác nhau như BERT, RoBERTa, DeBERTa để giải quyết bài toán Sentence Classification, từ đó giải quyết hai bài toán ACD và ACP. Chúng tôi đạt được kết quả state-of-the-art với bài toán ACP với accuracy là 90.05.
- Đề xuất 1 phương pháp sử dụng SHAP để giải thích dự đoán của mô hình cho bài toán ACP và ACP.

## 2 Related work

Sun et al [13] đề xuất phương pháp Xây dựng câu phụ để giải quyết hai bài toán ACD và ACP. Trong bài nghiên cứu của mình, Sun đề xuất 4 cách xây dựng câu phụ khác nhau là QA-M, NLI-M, QA-B, NLI-B. Mô hình NLI-B của Sun đã đạt được kết quả state-of-the-art ở bài toán ACD và mô hình QA-B đạt kết quả state-of-the-art ở bài toán ACP.

Li et al [6] sử dụng phương pháp Xây dựng câu phụ của Sun et al [13] để giải quyết hai bài toán ACD và ACP. Điểm cải tiến của Li so với Sun là Li có sử dụng thêm một tầng context-aware embedding để tạo ra context-based knowledge, sau đó sử dụng cơ chế gating là Gated Tanh-RELU Units để kết hợp context-based knowledge và cách biểu diễn câu của BERT, từ đó cải thiện cách biểu diễn câu của BERT và cải thiện hiệu quả của mô hình. Li đạt được kết quả tốt hơn Sun ở cả hai bài toán là ACD và ACP.

Hu et al [5] chuyển đổi bài toán ACD thành bài toán Multi-Label Few-Shot Learning (MFSL). Sau đó, họ đề xuất một phương pháp xử lý bài toán MFSL dựa trên phương pháp Prototypical Network [12], cùng với các cải tiến nhằm giảm đi độ nhiễu trong support set và query set. Về kết quả đạt được, nhóm tác giả đã outperform các cách tiếp cận Few-Shot Learning khác ở bài toán ACD trên nhiều tập dữ liệu khác nhau.

Yan et al [15] đề xuất 1 framework duy nhất để xử lý toàn bộ subtask của bài toán ABSA. Tuy trong bài viết của mình, Yan không đề cập gì đến cách xử lý gì cho 2 bài toán ACD và ACP nhưng chúng tôi nhận thấy có thể sử dụng cách của Yan để xử lý 2 bài toán này. Cách làm của Yan là cách làm rất hứa hẹn trong tương lai.

Về việc sử dụng XAI để giải thích kết quả mô hình, Danilevsky et al [1] đã thực hiện cuộc khảo sát trên 70 bài báo về XAI và đưa ra hướng dẫn cách chọn kỹ thuật XAI sao cho phù hợp với nhu cầu. Andreas et al [9] đã tổng hợp lại các phương pháp trong XAI để giải thích dự đoán của các mô hình NLP, trong đó, tiêu biểu nhất là SHAP, LIME, Anchors. Singh et al [11] đề xuất phương pháp giải thích mô hình bằng ngôn ngữ tự nhiên với sự giúp đỡ của các Mô Hình Ngôn Ngữ Lớn. Phương pháp của Singh trả về lời giải thích cho hoạt động của mô hình bằng ngôn ngữ tự nhiên và điểm số biểu thị độ tin cậy của lời giải thích đó.

# 3 Methodology

## 3.1 Mô tả bài toán

Cho trước 1 câu s, 1 tập hợp cố định các khía cạnh A={food, service, price, ambience, anecdotes/miscellaneous}, nhiệm vụ của ta là với mỗi khía cạnh a ∈ A, dự đoán cảm xúc p ∈ P={positive, neutral, negative, conflict, none}, với p là cảm xúc về khía cạnh a. Cảm xúc p là none nghĩa là câu s không đề cập gì đến khía cạnh a. Bảng 1 là một ví dụ cho bài toán cần giải quyết. Ta thấy rằng cảm xúc người dùng về khía cạnh price là negative, trong khi cảm xúc người dùng về khía cạnh ambience lại là none. Với việc thêm none vào P, chúng tôi có thể giải quyết 2 bài toán là ACD và ACP cùng lúc.

**Bảng 1** Một ví dụ về tập dữ liệu Semeval 2014 - Task 4

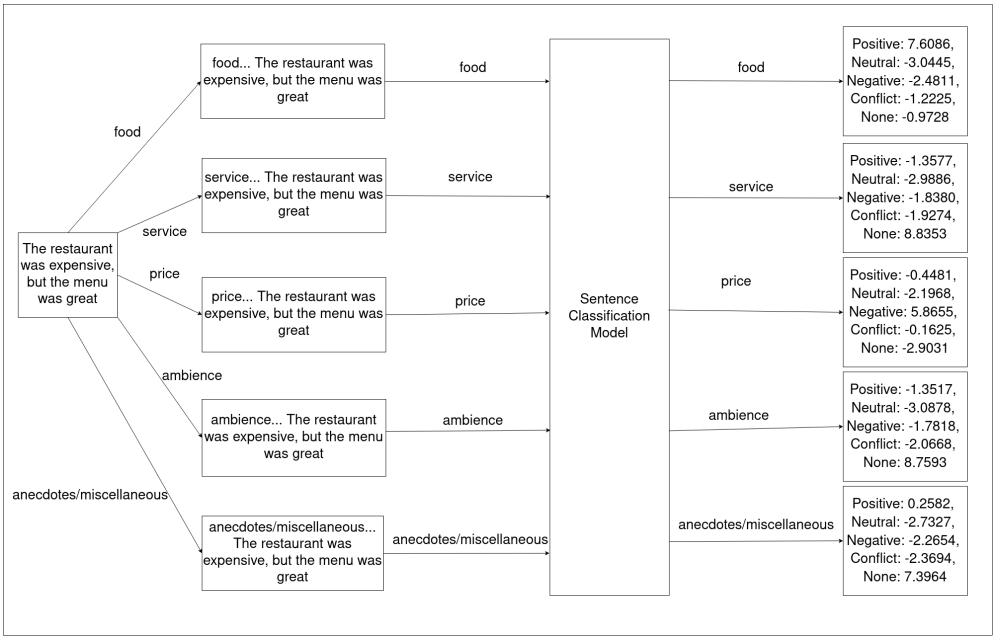
| **Ví dụ:** The restaurant was expensive, but the menu was great | |
|---|---|
| Khía cạnh | Cảm xúc |
| food | positive |
| service | none |
| price | negative |
| ambience | none |
| anecdotes/miscellaneous | none |

## 3.2 Xây dựng câu phụ

Với mỗi khía cạnh a ∈ A, ta tạo ra 1 câu phụ $B_a$ có cấu trúc là "{khía cạnh a}... {câu s}". Tập hợp A có 5 phần tử, vì vậy ta sẽ tạo ra 5 câu phụ, mỗi câu phụ ứng với 1 khía cạnh. Hình 1 là ví dụ minh họa cách tạo ra các câu phụ từ câu s là "The restaurant was expensive, but the menu was great". Như ta thấy, với khía cạnh food, ta tạo ra câu phụ $B_{food}$ là "food... The restaurant was expensive, but the menu was great", hay như với khía cạnh service, ta tạo ra câu phụ $B_{service}$ là "service... The restaurant was expensive, but the menu was great". Mục đích của việc xây dựng câu phụ là cho mô hình biết ta đang quan tâm đến khía cạnh nào của thực thể.

## 3.3 Sử dụng câu phụ

Như đã nói ở trên, với mỗi khía cạnh a ∈ A, ta tạo ra câu phụ $B_a$. Sau đó, ta truyền câu phụ $B_a$ vào Mô Hình Sentence Classification. Mô hinh Sentence Classification là mô hình được tinh chỉnh từ 1 trong 3 Mô Hình Ngôn Ngữ là BERT, RoBERTa và DeBERTa, có nhiệm vụ giải quyết bài toán Sentence Classification. Mô hình Sentence Classification giúp ta phân lớp câu phụ $B_a$ vào lớp p ∈ P, tương ứng với cảm xúc người dùng về khía cạnh a. VD: với câu "The restaurant was expensive, but the menu was great" và khía cạnh a là price, ta tạo ra câu phụ $B_{price}$ là "price... The restaurant was expensive, but the menu was great". Sau đó ta truyền câu phụ $B_{price}$ vào mô hình Sentence Classification, mô hình sẽ gán câu phụ $B_{price}$ thuộc về lớp negative, nghĩa là cảm xúc người dùng về khía cạnh giá cả là tiêu cực.




**Hình 1** Ví dụ về phương pháp của chúng tôi để giải quyết bài toán ACD và ACP

Trên thực tế, mô hình Sentence Classification sẽ không đưa ra chính xác câu phụ $B_a$ thuộc về lớp nào mà sẽ đưa ra điểm số mà mô hình gán cho từng cảm xúc. Sau đó ta dùng hàm softmax để tìm ra cảm xúc có điểm số cao nhất, và đó chính là cảm xúc của người dùng về khía cạnh a. Hình 1 là ví dụ cho cách chúng tôi tạo ra và sử dụng các câu phụ. Ta thấy với câu phụ $B_{price}$ là "price... The restaurant was expensive, but the menu was great", mô hình trả về ta các điểm số như sau: {Positive: -0.4481, Neutral: -2.1968, Negative: 5.8655, Conflict: -0.1625, None: -2.9031}, ta thấy cảm xúc Negative có số điểm cao nhất, nghĩa là cảm xúc người dùng về khía cạnh giá cả là tiêu cực.

Tóm lại, khi ta truyền câu phụ $B_a$ vào mô hình Sentence Classification, ta nhận được $L_a = [L_{a-p}\ for\ p\ \in\ P]$, với $L_{a-p}$ là điểm số của cảm xúc p. Cảm xúc p có giá trị $L_{a-p}$ lớn nhất sẽ là cảm xúc của người dùng về khía cạnh a.

## 3.4 Tinh chỉnh các Mô Hình Ngôn Ngữ Lớn

### 3.4.1 Các Mô Hình Ngôn Ngữ Lớn

Các mô hình truyền thống như RNN, LSTM [4], Gated RNN đều là các mô hình tốt, và đạt được nhiều kết quả khả quan trong các bài toán NLP. Tuy nhiên, chúng có điểm hạn chế là tại mỗi thời điểm t, các mô hình này sẽ tạo ra một hidden state $h_t$ như là một hàm với 2 tham số là $h_{t-1}$ và input tại vị trí t. Điều này có nghĩa là nếu muốn tính hidden state $h_t$, ta phải tính hidden state $h_{t-1}$ trước, khiến việc huấn luyện không thể thực hiện một cách song song được.

Kiến trúc Transformers [14] đã bỏ đi tính liên tục của các mô hình truyền thống, cho phép huấn luyện mô hình một cách song song. Điều này cho phép ta tạo ra các Mô Hình Ngôn Ngữ Lớn (hàng trăm triệu tham số), huấn luyện các Mô Hình Ngôn Ngữ đó trên các tập dữ liệu lớn (hàng trăm GB), mà không tốn quá nhiều thời gian, tài nguyên và công sức. Nhờ việc có số lượng tham số lớn và được huấn luyện trên một tập dữ liệu lớn, các Mô Hình Ngôn Ngữ Lớn này hiểu ngôn ngữ tự nhiên rất tốt, và outperform các mô hình truyền thống trên nhiều bài toán NLP.

Ở bài nghiên cứu này, chúng tôi tinh chỉnh 3 Mô Hình Ngôn Ngữ Lớn khác nhau là BERT, RoBERTa và DeBERTa.

BERT [2] có cấu trúc là nhiều lớp Transformers Block được xếp chồng lên nhau. Bằng việc được huấn luyện bởi hai bài toán là Masked Language Modelling và Next Sentence Prediction, BERT tạo ra được các vector biểu diễn từ sử dụng ngữ cảnh từ hai chiều của từ. Mô hình BERT có thể được tinh chỉnh và thêm một tầng output layer để tạo ra mô hình state-of-the-art cho nhiều bài toán khác nhau như là Question Answering, Language Inteference, ... mà không cần thay đổi quá nhiều kiến trúc của mô hình.

RoBERTa [7] là mô hình BERT sau khi được áp dụng một vài sự cải tiến. Các sự cải tiến này bao gồm: (1) Chỉnh sửa lại các tham số khi huấn luyện mô hình và (2) Huấn luyện mô hình với bộ dữ liệu lớn hơn. RoBERTa có hiệu suất tốt hơn BERT ở cả 3 bộ dữ liệu benchmark là GLUE, SQuAD và RACE.

DeBERTa [3] là kiến trúc mô hình mới. DeBERTa cải thiện BERT và RoBERTa bằng hai kỹ thuật chính là disentangled attention mechanism và enhanced mask decoder. Kỹ thuật disentangled attention mechanism là việc biểu diễn một từ bằng 2 vector riêng biệt, 1 vector biểu diễn ngữ nghĩa từ và 1 vector biểu diễn vị trí từ, thay vì biểu diễn cả ngữ nghĩa từ và vị trí từ trong cùng 1 vector như BERT và RoBERTa. Kỹ thuật enhaced mask decoder là việc kết hợp thêm thông tin vị trí tuyệt đối của từ vào tầng decoding trong khi đang huấn luyện mô hình với bài toán Masked Language Modelling. DeBERTa có kết quả tốt hơn RoBERTa ở nhiều bài toán NLP khác nhau, đặc biệt ở benchmark SuperGLUE, DeBERTa đã vượt qua con người với kết quả là 89.9 so với 89.8.

### 3.4.2 Thủ tục tinh chỉnh

BERT, RoBERTa và DeBERTa còn được gọi là các Pretrained Model, và chúng đều được huấn luyện bởi hàng trăm GB dữ liệu, nên chúng có thể tạo ra các vector biểu diễn từ giàu ngữ nghĩa. Tinh chỉnh là quá trình ta tiếp tục huấn luyện các Pretrained Model bằng dữ liệu liên quan đến bài toán mà ta quan tâm. Điều này giúp ta tận dụng được lượng kiến thức phong phú của Pretrained Model vào bài toán của mình. Trong bài nghiên cứu này, chúng tôi chuyển đổi hai bài toán Aspect Category Detection và Aspect Category Polarity thành bài toán Sentence Classification, sau đó tinh chỉnh 3 Mô Hình Ngôn Ngữ là BERT, RoBERTa và DeBERTa bằng bộ dữ liệu SemEval-2014 Task 4 để tạo ra mô hình Sentence Classification, từ đó giải quyết hai bài toán Aspect Category Detection và Aspect Category Polarity.

## 3.5 Dùng SHAP để giải thích dự đoán của mô hình

### 3.5.1 Chuẩn hóa kết quả trả về của mô hình

Với mỗi khía cạnh a, ta tạo ra câu phụ $B_a$. Sau khi truyền câu phụ $B_a$ vào mô hình Sentence Classification, mô hình trả về $L_a = [L_{a-p} \, for \, p \in P]$, với $L_{a-p}$ là điểm số của cảm xúc p. Cảm xúc p có điểm số cao nhất sẽ là cảm xúc người dùng về khía cạnh a. Tuy nhiên, $L_a$ có 2 điểm hạn chế như sau: (1) các số $L_{a-p}$ là các số thực bất kỳ, giá trị của chúng có thể là số âm và (2) tổng của các phần tử trong $L_a$ cũng là một số thực bất kỳ nào đó. Do đó, ta chuẩn hóa $L_a$ thành $C_a = [C_{a-p} \, for \, p \in P]$ với $C_{a-p}$ là xác suất cảm xúc người dùng về khía cạnh a là p. Ta chuẩn hóa $L_a$ thành $C_a$ theo công thức 1.

$$C_{a-p} = \left(\frac{e^{L_{a-p}}}{\sum_{p2}^{P} e^{L_{a-p2}}} \times 100\right) for \, p \in P \tag{1}$$

Công thức 1 giúp ta đảm bảo các điều sau: (1) $L_{a-p}$ càng lớn thì $C_{a-p}$ càng lớn, (2) $C_{a-p}$ đều có giá trị từ 0 đến 100 và (3) Tổng của các phần tử trong $C_a$ là 100. Do đó, ta có thể nói $C_{a-p}$ là xác suất cảm xúc người dùng về khía cạnh a là p.

### 3.5.2 Dùng SHAP để giải thích kết quả sau khi chuẩn hóa

Sau khi chuẩn hóa kết quả trả về của mô hình thì với mỗi cặp {khía cạnh a - cảm xúc p} ta có thông tin $C_{a-p}$ là xác suất cảm xúc người dùng về khía cạnh a là p. Vì vậy với mỗi cặp {a-p}, ta dùng SHAP giải thích tại sao mô hình đưa ra số $C_{a-p}$.

SHAP là một phương pháp để giải thích dự đoán của một mô hình. SHAP tính giá trị SHapley cho từng thuộc tính. Giá trị SHapley của thuộc tính đó càng lớn thì thuộc tính càng quan trọng. Trong bài nghiên cứu này, các thuộc tính là các từ trong câu s. Với mỗi cặp {khía cạnh a-cảm xúc p}, ta dùng SHAP tính giá trị SHapley cho từng từ trong câu s. Giá trị SHapley của mỗi từ thể hiện từ đó ảnh hưởng thế nào đến kết quả $C_{a-p}$. Nếu giá trị SHapley của một từ là dương thì từ đó làm tăng xác suất cảm xúc người dùng về khía cạnh a là p, còn nếu là âm thì nghĩa là từ đó làm giảm xác suất cảm xúc người dùng về khía cạnh a là p.

## 3.6 Experiments

### 3.6.1 Datasets

Chúng tôi đánh giá phương pháp của mình bằng bộ dữ liệu SemEval 2014-Task 4. Bộ dữ liệu này là tập hợp các review của người dùng về các nhà hàng, bao gồm 3042 câu trong tập train và 800 câu trong tập test. Bộ dữ liệu này sử dụng tập hợp A và P như đã đề cập ở phần Miêu tả bài toán. Để đánh giá phương pháp, chúng tôi sử dụng độ đo micro-F1 ở bài toán Aspect Category Detection và độ đo accuracy ở bài toán Aspect Category Polarity.

### 3.6.2 Tinh chỉnh các Mô Hình Ngôn Ngữ

Chúng tôi tinh chỉnh lần lượt 3 Mô Hình Ngôn Ngữ là BERT, RoBERTa và DeBERTa bằng bộ dữ liệu SemEval 2014-Task 4 theo các tham số như bảng 2. Cột Mô Hình Sentence Classification chính là tên của mô hình sau khi được tinh chỉnh, cột Mô Hình Ngôn Ngữ là tên của Mô hình Ngôn Ngữ mà chúng tôi dùng để tinh chỉnh, các cột còn lại là tham số dùng để tinh chỉnh.

**Bảng 2** Bảng thể hiện các tham số chúng tôi dùng để tinh chỉnh mô hình BERT, RoBERTa, DeBERTa

| Mô hình Sentence Classification | Mô Hình Ngôn Ngữ | Learning Rate | Batch Size | Epoch |
|---|---|---|---|---|
| A-BERT | BERT | 2e-5 | 24 | 5 |
| A-RoBERTa | RoBERTa | 2e-5 | 24 | 5 |
| A-DeBERTa | DeBERTa | 2e-5 | 16 | 5 |

### 3.6.3 Kết quả hai bài toán Aspect Category Detection và Aspect Category Polarity

Kết quả của mô hình chúng tôi và sự so sánh đến các mô hình khác ở hai bài toán Aspect Category Detection và Aspect Category Polarity được thể hiện trong bảng 3 và 4. 3 mô hình ngôn ngữ sau khi được chúng tôi tinh chỉnh lần lượt được đặt tên là A-BERT, A-RoBERTa và A-DeBERTa. Ngoài ra chúng tôi còn so sánh hiệu suất của 3 mô hình đó với các mô hình khác của Sun et al [13] và Li et al [6]. Ta thấy rằng mô hình A-DeBERTa có kết quả tốt hơn so với các mô hình của Sun et al [13] nhưng lại thua các mô hình của Li et al [6] ở bài toán Aspect Category Detection. Tuy nhiên, cả mô hình A-RoBERTa và A-DeBERTa đều outperform tất cả mô hình của Sun et al [13] và Li et al [6] ở bài toán Aspect Category Polarity.

**Bảng 3** Bảng thể hiện hiệu suất 3 mô hình A-BERT, A-RoBERTa, A-DeBERTa và so sánh với các mô hình của Li et al [6] và Sun et al [13] ở bài toán Aspect Category Detection trên bộ dữ liệu SemEval 2014-Task 4

| Mô hình | F1 | Precision | Recall |
|---|---|---|---|
| A-BERT | 89.18 | 91.66 | 86.83 |
| A-RoBERTa | 91.99 | 93.01 | 90.93 |
| A-DeBERTa | 92.39 | 93.59 | 91.22 |
| BERT-pair-QA-M [13] | 91.54 | 92.87 | 90.24 |
| BERT-pair-NLI-M [13] | 91.67 | 93.15 | 90.24 |
| BERT-pair-QA-B [13] | 91.47 | 93.04 | 89.95 |
| BERT-pair-NLI-B [13] | 92.18 | 93.57 | 90.83 |
| BERT-pair-QA-M-GBCN [6] | 92.44 | 93.59 | 91.32 |
| BERT-pair-NLI-M-GBCN [6] | **92.89** | 94.26 | 91.55 |

**Bảng 4** Bảng thể hiện hiệu suất 3 mô hình A-BERT, A-RoBERTa, A-DeBERTa và so sánh với các mô hình của Li et al [6] và Sun et al [13] ở bài toán Aspect Category Polarity trên bộ dữ liệu SemEval 2014-Task 4

| Mô hình | Accuracy |
|---|---|
| A-BERT | 82.15 |
| A-RoBERTa | 89.27 |
| A-DeBERTa | **90.15** |
| BERT-pair-QA-M [13] | 85.2 |
| BERT-pair-NLI-M [13] | 85.1 |
| BERT-pair-QA-B [13] | 85.9 |
| BERT-pair-NLI-B [13] | 84.6 |
| BERT-pair-QA-M-GBCN [6] | 86.4 |
| BERT-pair-NLI-M-GBCN [6] | 86.0 |

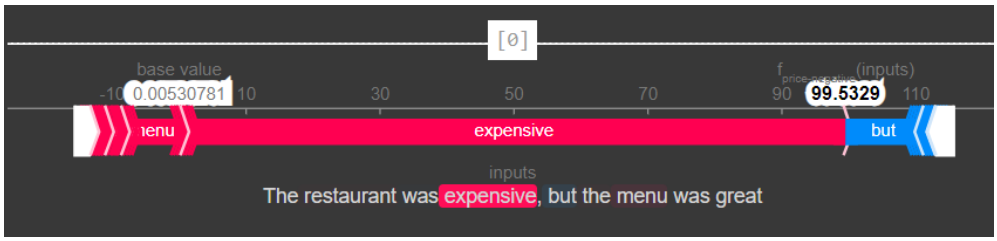
### 3.6.4 Dùng SHAP để giải thích dự đoán mô hình

Chúng tôi dùng package SHAP trong Python để giải thích và visualize dự đoán của mô hình. Với mỗi cặp {khía cạnh a-cảm xúc p}, package SHAP trả về 3 thông tin sau: (1) Base Value là xác suất cảm xúc người dùng về khía cạnh a là cảm xúc p nếu toàn bộ từ trong câu s bị che đi, (2) Prediction Value là xác suất cảm xúc người dùng về khía cạnh a là cảm xúc p và (3) là giá trị SHapley của từng từ trong câu. Ta có tổng giá trị SHapley của các từ trong câu bằng hiệu giữa Prediction Value và Base Value. Ta hình dung các từ như là các lực để đẩy giá trị dự đoán của mô hình từ Base Value đến Prediction Value. Package SHAP còn có thể dùng 3 thông tin này để visualize việc giải thích dự đoán của mô hình.

VD: với câu s là "The restaurant was expensive, but the menu was great" và cặp {khía canh-cảm xúc} là {price-negative}, mô hình trả về ta 3 thông tin: (1) Base Value là 0.0053, nghĩa là xác suất cảm xúc người dùng về khía cạnh thức ăn là tiêu cực khi toàn bộ từ trong câu s bị che đi là 0.0053%, (2) Prediction Value là 99.5329, tức xác suất cảm xúc người dùng về khía cạnh thức ăn là tiêu cực là 99.5329% và (3) là giá trị SHapley của từng từ trong câu s, các giá trị này được chúng tôi biểu diễn ở bảng 5. Ta thấy từ "expensive" có giá trị SHapley là 99.041, tức là từ "expensive" làm tăng xác suất cảm xúc người dùng về khía cạnh giá cả là tiêu cực lên 99.041%. Hay từ "but" có giá trị SHapley là -11.1652, nghĩa là từ "but" làm giảm xác suất cảm xúc người dùng về khía cạnh giá cả là tiêu cực đi 11.1652%.

Package SHAP còn giúp ta tạo ra các biểu đồ biểu diễn cách mô hình hoạt động dựa trên 3 thông tin SHAP trả về. Hình 2 là ví dụ minh họa cho cách package SHAP tạo biểu đồ biểu diễn cách mô hình hoạt động. Ta có thể thấy SHAP đưa ra hai biểu đồ, biểu đồ lực ở trên và biểu đồ nhiệt ở dưới. Ở biểu đồ lực, ta thấy các từ như là các lực có độ lớn là giá trị SHapley của từ đó, và các lực này sẽ đẩy dần quyết định của mô hình từ Base Value đến Prediction Value. Còn biểu đồ nhiệt giúp ta nhanh chóng xác định các từ có tác động lớn đến quyết định của mô hình. Ta thấy từ "expensive"

có độ đậm rất lớn, nghĩa là từ "expensive" có tác động rất lớn đến quyết định xác suất cảm xúc người dùng về khía cạnh giá cả là tiêu cực là 99.5329%.

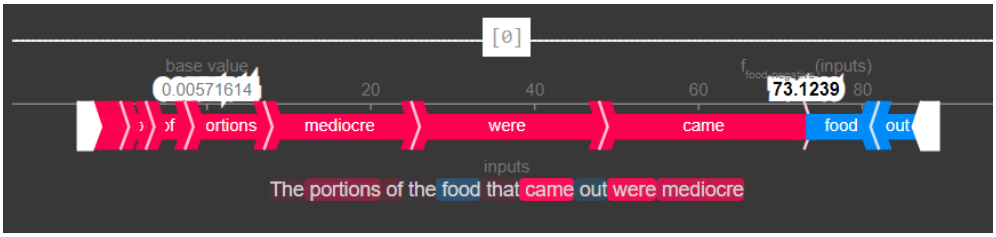



**Hình 2** Package SHAP tạo ra biểu đồ biểu diễn cách mô hình hoạt động

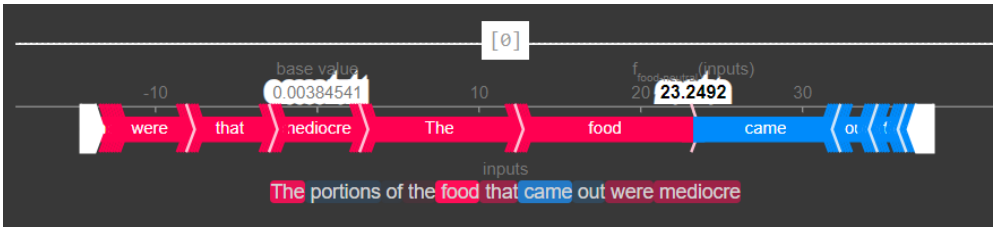
**Bảng 5** Giá trị SHapley của các từ với cặp {price-negative}

| Từ | Giá trị SHapley |
|---|---|
| The | 0.126 |
| restaurant | 2.182 |
| was | -1.781 |
| expensive | 99.041 |
| but | -11.1652 |
| the | 2.402 |
| menu | 8.844 |
| was | -0.026 |
| great | -0.095 |

Việc giải thích còn giúp ta hiểu tại sao mô hình lại đưa ra quyết định sai lầm. VD như với câu s là "The portions of the food that came out were mediocre", cảm xúc đúng của người dùng về khía cạnh thức ăn là neutral, nhưng mô hình lại cho rằng cảm xúc của người dùng về khía cạnh thức ăn là negative. Chi tiết hơn, với khía cạnh thức ăn, sau khi chuẩn hóa kết quả trả về của mô hình ta thu được $C_{food}$={Positive: 0.1587, Neutral: 23.2492, Negative: 73.1239, Conflict: 3.40423, None: 0.0639721}. Ta thấy rằng $C_{food-neutral}$ thấp hơn $C_{food-negative}$ rất nhiều. Hình 3 cho thấy cách SHAP giải thích tại sao mô hình lại quyết định $C_{food-negative}$ = 73.1239%. Nhìn vào biểu đồ nhiệt, ta thấy rằng từ "came" có ảnh hưởng rất lớn, làm tăng đáng kể $C_{food-negative}$. Hình 4 cho thấy cách SHAP giải thích về quyết định $C_{food-neutral}$=23.2492% của mô hình. Ta thấy từ "came" làm giảm xác suất $C_{food-neutral}$ đi. Từ 2 thông tin đó, ta có thể cho rằng từ "came" có ảnh hưởng lớn đến sự sai lệch trong việc ra quyết định của mô hình.




**Hình 3** SHAP giải thích lý do mô hình quyết định $C_{food-negative}$=73.1239%




**Hình 4** SHAP giải thích lý do mô hình quyết định $C_{food-neutral}$=23.2492%

# 4 Discussion

Ta thấy, kết quả của mô hình A-BERT thua xa 2 mô hình A-RoBERTa và A-DeBERTa ở cả hai bài toán là Aspect Category Detection và Aspect Category Polarity. Điều này cũng là dễ hiểu vì các tác giả của RoBERTa và DeBERTa đã đưa ra các sự cải tiến so với BERT trong bài nghiên cứu của mình. Ngoài ra, RoBERTa và DeBERTa đều outperform BERT trên nhiều bộ dữ liệu benchmark khác nhau.

Ta thấy, kết quả của mô hình A-BERT thua các mô hình được tinh chỉnh từ BERT khác của Li et al [6] và Sun et al [13] ở cả 2 bài toán là Aspect Category Detection và Aspect Category Polarity. Điều này có nghĩa là nếu ta áp dụng cách tạo câu phụ của Li et al [6] và Sun et al [13] lên mô hình DeBERTa thì có thể thu được kết quả tốt hơn mô hình A-DeBERTa của chúng tôi.

Về việc tại sao mô hình A-RoBERTa và A-DeBERTa không thể hiện quá tốt ở bài toán Aspect Category Detection (chỉ hơn được Sun et al [13] 0.21 điểm F1, và thua Li et al [6] 0.6 điểm F1 nhưng lại outperform cả Sun và Li ở bài toán Aspect Category Polarity, chúng tôi cho rằng do (1) RoBERTa và DeBERTa là những mô hình tốt và outperform BERT ở nhiều bài toán NLP nên mô hình được tinh chỉnh từ RoBERTa và DeBERTa của chúng tôi mới có thể outperform mô hình từ BERT của Sun và Li và (2) có thể RoBERTa và DeBERTa mạnh ở việc xác định cảm xúc của người dùng về từng khía cạnh chứ không mạnh về việc xác định xem người dùng có đề cập đến khía cạnh hay không. Để giúp người đọc tin tưởng hơn vào kết quả nghiên cứu, chúng tôi đã public source code tại: https://github.com/Lang0808/Paper.

## 5 Conclusion

Trong bài nghiên cứu này, chúng tôi đã chuyển đổi hai bài toán là Aspect Category Detection và Aspect Category Polarity thành bài toán Sentence Classification. Sau đó chúng tôi tinh chỉnh các Mô Hình Ngôn Ngữ như BERT, RoBERTa và DeBERTa để giải quyết bài toán Sentence Classification, từ đó giải quyết hai bài toán Aspect Category Detection và Aspect Category Polarity. Cuối cùng, chúng tôi dùng SHAP để giải thích dự đoán của mô hình. Trong tương lai, chúng tôi sẽ tìm thêm các cách xây dựng câu phụ khác để có thể tạo ra các mô hình có hiệu suất tốt hơn nữa.

## References


[1] Marina Danilevsky et al. "A Survey of the State of Explainable AI for Natural Language Processing". In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*. Suzhou, China: Association for Computational Linguistics, Dec. 2020, pp. 447–459. URL: https://aclanthology.org/2020.aacl-main.46.

[2] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423.

[3] Pengcheng He et al. "DeBERTa: Decoding-enhanced BERT with Disentangled Attention". In: *arXiv e-prints*, arXiv:2006.03654 (June 2020), arXiv:2006.03654. DOI: 10.48550/arXiv.2006.03654. arXiv: 2006.03654 [cs.CL].

[4] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-term Memory". In: *Neural computation* 9 (Dec. 1997), pp. 1735–80. DOI: 10.1162/neco.1997.9.8.1735.

[5] Mengting Hu et al. "Multi-Label Few-Shot Learning for Aspect Category Detection". In: *CoRR* abs/2105.14174 (2021). arXiv: 2105.14174. URL: https://arxiv.org/abs/2105.14174.

[6] Xinlong Li et al. "Enhancing BERT Representation With Context-Aware Embedding for Aspect-Based Sentiment Analysis". In: *IEEE Access* 8 (2020), pp. 46868–46876. DOI: 10.1109/ACCESS.2020.2978511.

[7] Yinhan Liu et al. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. cite arxiv:1907.11692. 2019. URL: http://arxiv.org/abs/1907.11692.

[8] Scott M. Lundberg and Su-In Lee. "A unified approach to interpreting model predictions". In: *CoRR* abs/1705.07874 (2017). arXiv: 1705.07874. URL: http://arxiv.org/abs/1705.07874.

[9] Andreas Madsen, Siva Reddy, and Sarath Chandar. "Post-hoc Interpretability for Neural NLP: A Survey". In: *CoRR* abs/2108.04840 (2021). arXiv: 2108.04840. URL: https://arxiv.org/abs/2108.04840.

[10] Maria Pontiki et al. "SemEval-2014 Task 4: Aspect Based Sentiment Analysis". In: *Proceedings of the 8th International Workshop on Semantic Evaluation*

*(SemEval 2014)*. Dublin, Ireland: Association for Computational Linguistics, Aug. 2014, pp. 27–35. DOI: 10.3115/v1/S14-2004. URL: https://aclanthology.org/S14-2004.

[11] Chandan Singh et al. *Explaining black box text modules in natural language with language models*. 2023. arXiv: 2305.09863 [cs.AI].

[12] Jake Snell, Kevin Swersky, and Richard S. Zemel. "Prototypical Networks for Few-shot Learning". In: *CoRR* abs/1703.05175 (2017). arXiv: 1703.05175. URL: http://arxiv.org/abs/1703.05175.

[13] Chi Sun, Luyao Huang, and Xipeng Qiu. "Utilizing BERT for Aspect-Based Sentiment Analysis via Constructing Auxiliary Sentence". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 380–385. DOI: 10.18653/v1/N19-1035. URL: https://aclanthology.org/N19-1035.

[14] Ashish Vaswani et al. "Attention Is All You Need". In: *CoRR* abs/1706.03762 (2017). arXiv: 1706.03762. URL: http://arxiv.org/abs/1706.03762.

[15] Hang Yan et al. "A Unified Generative Framework for Aspect-Based Sentiment Analysis". In: *CoRR* abs/2106.04300 (2021). arXiv: 2106.04300. URL: https://arxiv.org/abs/2106.04300.