

Đại Học Bách Khoa Hà Nội  
Trường Công nghệ thông tin và truyền thông

=====oOo=====



## BÁO CÁO

MÔN: Khai phá web

### **Đề tài: Aspect-Based Sentiment Analysis**

Giảng viên hướng dẫn: TS. Nguyễn Duy Hiệp.

Sinh viên thực hiện: Quách Thế Trường -20194196

*Hà Nội, ngày 5 tháng 7 năm 2023*

## Mục lục

<b>1. Giới thiệu chung</b>	<b>3</b>
<b>2. Chi tiết dữ liệu sử dụng</b>	<b>3</b>
2.1 Tổng quan về dữ liệu	3
2.2 Các thống kê về dữ liệu	4
2.3 Xử lý dữ liệu	5
<b>3. Mô hình đề xuất</b>	<b>7</b>
3.1 Tổng quan về bài toán Aspect-Based Sentiment Analysis	7
3.2 Mô hình triển khai	8
3.2.1 Tổng quan về Transfer Learning	8
3.2.2. Tổng quan về Instruction Learning	9
3.2.3. Tổng quan về Transformers	11
3.2.4. Text-to-text Transfer Transformers	15
<b>4. Thực nghiệm mô hình</b>	<b>18</b>
4.1. Môi trường thực nghiệm và mô hình cài đặt	18
4.2 Kết quả thực nghiệm	19
<b>5. Kết luận và hướng phát triển</b>	<b>23</b>
<b>6. Tài liệu tham khảo</b>	<b>24</b>

## 1. Giới thiệu chung

Song song với sự phát triển bùng nổ của các sàn thương mại điện tử, việc thu thập thông tin phản hồi của khách hàng là một cách tuyệt vời giúp cho các doanh nghiệp, các cửa hàng hiểu được điểm mạnh, điểm yếu trong sản phẩm, dịch vụ của mình; đồng thời nhanh chóng nắm bắt được tâm kỳ và nhu cầu khách hàng để mang đến cho họ sản phẩm, dịch vụ hoàn hảo nhất.

Người dùng sử dụng Internet để tìm kiếm, tham khảo trước khi đưa ra quyết định về sử dụng một sản phẩm hay dịch vụ nào đó. Sau khi đã đặt mua và sử dụng sản phẩm, người dùng để lại thông tin phản hồi (feedback). 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 1 doanh nghiệp có 1 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.

Chính vì vậy, bài toán Sentiment Analysis ra đời nhằm giải quyết vấn đề này. Với đầu vào là một câu bình luận của khách hàng dưới dạng văn bản, và đầu ra là nhãn lớp cảm xúc tương ứng (positive, negative, neutral).

- Tuy nhiên, trên thực tế, feedback của khách hàng rất đa dạng và phủ rộng trên nhiều khía cạnh (ví dụ dịch vụ shipper, dịch vụ bảo hành, chất lượng sản phẩm,...) và với mỗi khía cạnh thì thái độ phản hồi của khách hàng có sự khác nhau. Vì vậy, em thực hiện đề tài có tính thực tiễn cao hơn, đó là *“Phân tích cảm xúc dựa trên khía cạnh” (ABSA)*

## 2. Chi tiết dữ liệu sử dụng

### 2.1 Tổng quan về dữ liệu

Dữ liệu được sử dụng là dữ liệu từ 4 tập Laptop14, Restaurant14, Restaurant15, Restaurant16. Bộ dữ liệu bao gồm các đánh giá của khách hàng với chú thích do con người tạo xác định các khía cạnh được đề cập của các thực thể mục tiêu và phân cực tình cảm của từng khía cạnh sẽ được cung cấp. Bộ dữ liệu gồm tập các câu đánh giá bằng tiếng Anh của khách hàng kèm với các nhãn khía cạnh(aspect), ý kiến(opinion), phân cực(polarity), phân loại(category). Mỗi tập dữ liệu được chia thành 2 tập test và training có kích thước như bảng bên dưới:

	Training	Test
--	----------	------

Laptop14	2934	816
Restaurant14	1266	492
Restaurant15	605	322
Restaurant16	1530	583
Chinese_zhang	19485	5567
Synthetic	2356	0

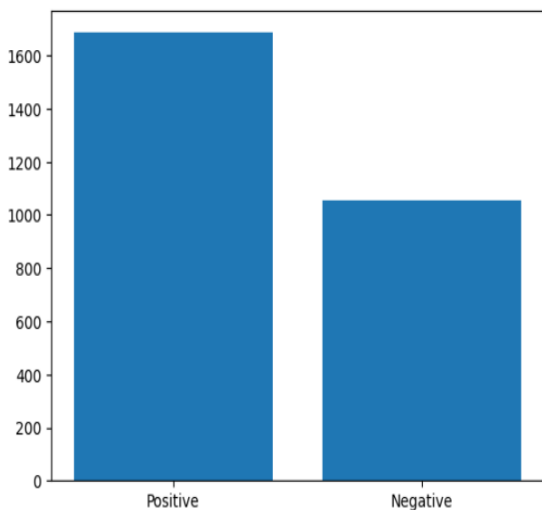
Bảng 1: Kích thước các tập dữ liệu (câu)

1 ví dụ trong bộ restaurant14:

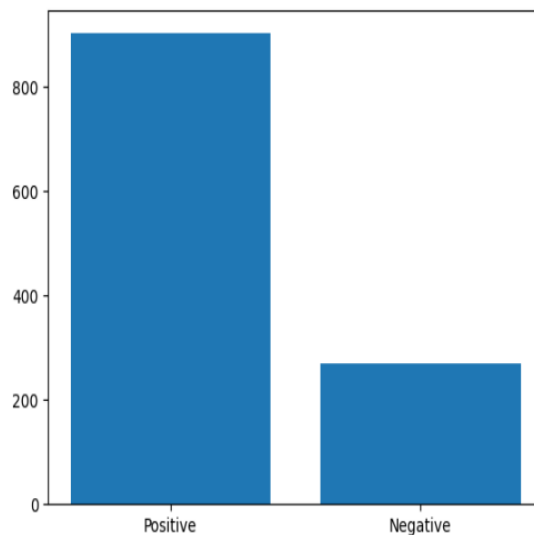
"But the staff was so horrible to us."	[ { "from": "8", "polarity": "negative", "term": "staff", "to": "13" } ]	[ { "category": "service", "polarity": "negative" } ]	"restaurants"	"3121"
--	--	--	---------------	--------

## 2.2 Các thống kê về dữ liệu

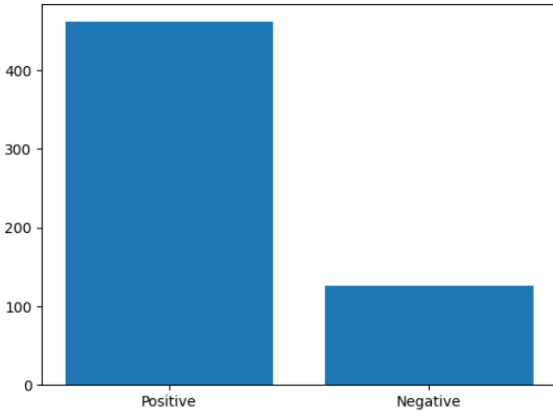
- Phân bố polarity



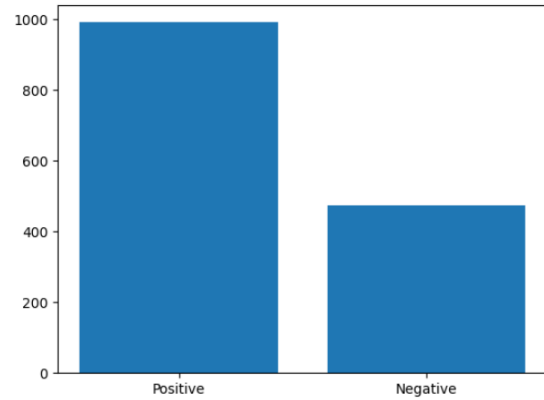
Hình 1: Laptop14



Hình 2: Restaurant14



Hình 3: Restaurant15

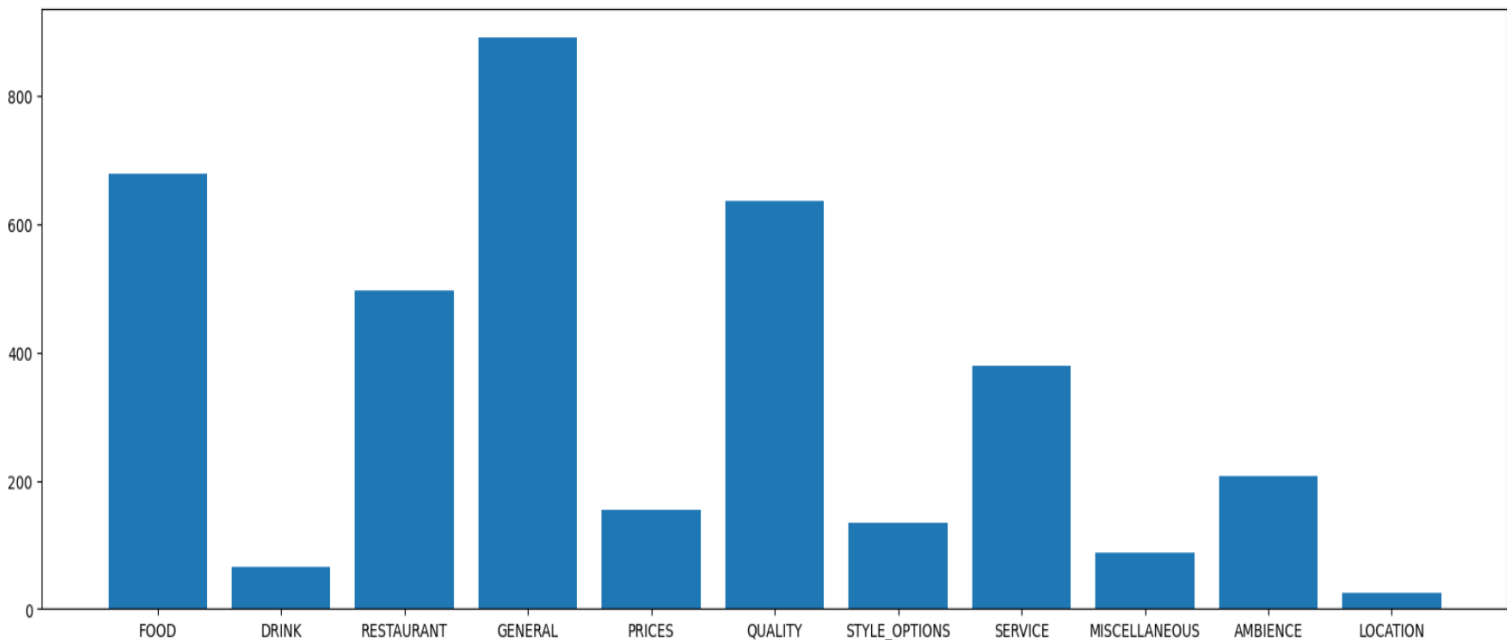


Hình 4: Restaurant16

Có thể thấy tỉ lệ phân cực ở 4 tập này đều giống nhau, có lượng positive cao hơn negative.

- Phân bố về loại

Phân loại đánh giá xuất hiện ở tập Restaurant16 bao gồm: FOOD, DRINK, RESTAURANT, GENERAL, PRICES, QUALITY, STYLE\_OPTIONS, SERVICE, MISCELLANEOUS, AMBIENCE, LOCATION. Một câu nói có thể phân bố thành nhiều loại tập trung chủ yếu ở GENERAL, FOOD và QUALITY. Trong khi đó, ta có thể thấy DRINK và LOCATION là thấp nhất.



Hình 5: Phân bố phân loại ở Restaurant16

## 2.3 Xử lý dữ liệu

Dữ liệu sau khi đọc dưới dạng thô được xử lý qua phương thức “prepare\_instruction\_dataloader” trong lớp “InstructDatasetLoader”. Dữ liệu đi qua 4 luồng nhiệm vụ trong đó có 2 luồng chính là ATE task xác định khía cạnh của thuật ngữ(aspect) và APC task xác định phân cực (polarities) ngoài ra còn có opinion task và category task.

Ví dụ về tập dữ liệu ban đầu:

text	labels
the duck confit is always amazing and the foie gras terrine with figs was out of this world .	{'aspect': 'foie gras terrine with figs', 'opinion': 'out of this world', 'polarity': 'positive', 'category': 'FOOD#QUALITY'},{'aspect': 'duck confit', 'opinion': 'amazing', 'polarity': 'positive', 'category': 'FOOD#QUALITY'}
the wine list is interesting and has many good values .	{'aspect': 'wine list', 'opinion': 'interesting', 'polarity': 'positive', 'category': 'DRINKS#STYLE_OPTIONS'},{'aspect': 'wine list', 'opinion': 'good values', 'polarity': 'positive', 'category': 'DRINKS#PRICES'}
for the price , you can not eat this well in manhattan .	{'aspect': 'NULL', 'opinion': 'well', 'polarity': 'positive', 'category': 'RESTAURANT#PRICES'},{'aspect': 'NULL', 'opinion': 'well', 'polarity': 'positive', 'category': 'FOOD#QUALITY'}
i was very disappointed with this restaurant .	{'aspect': 'restaurant', 'opinion': 'disappointed', 'polarity': 'negative', 'category': 'RESTAURANT#GENERAL'}
ive asked a cart attendant for a lotus leaf wrapped rice and she replied back rice and just walked away .	{'aspect': 'cart attendant', 'opinion': 'NULL', 'polarity': 'negative', 'category': 'SERVICE#GENERAL'}

Hình 6: Dữ liệu nhận ban đầu

Dữ liệu gồm phần text là các đánh giá bằng tiếng anh, phần labels gồm khía cạnh(aspect), ý kiến(opinion), phân cực(polarity), phân loại(category) được ghi chú lại.

Ví dụ về dữ liệu sau khi xử lý bằng lớp “InstructDatasetLoader”:

text	labels
Definition: The input are sentences about a product or service. The task is to extract the aspects and their corresponding polarity. Here are some examples: example 1- input: I charge it at night and skip taking the cord with me because of the good battery life. The aspects are: battery life, cord None battery life:good cord:NULL example 2- input: Great food, good size menu, great service and an unpretentious setting. The aspects are: food, menu, service, setting None food:great menu:good service:great setting:unpretentious Now extract opinions for the following example: input:love their drink menu .The aspects are: drink menu let us extract opinions one by one:	drink menu:love
Definition: The input are sentences about a product or service. The task is to extract the aspects and their corresponding categories. Here are some examples: example 1- input: I charge it at night and skip taking the cord with me because of the good battery life. The aspects are: battery life, cord None battery life:POWER_SUPPLY#GENERAL cord:NULL example 2- input: Great food, good size menu, great service and an unpretentious setting. The aspects are: food:FOOD#QUALITY  menu:RESTAURANT#GENERAL service:SERVICE#GENERAL setting:SERVICE#GENERAL None food:FOOD#QUALITY, menu:RESTAURANT#GENERAL, service:SERVICE#GENERAL, setting:SERVICE#GENERAL Now extract categories for the following example: input: love their drink menu .The aspects are: drink menu let us extract categories one by one:	drink menu:DRINKS#STYLE_OPTIONS
Definition: The input are sentences about a product or service. The task is to extract the aspects. Here are some examples: example 1- input: I charge it at night and skip taking the cord with me because of the good battery life. None aspect:battery life aspect:cord example 2- input: Great food, good size menu, great service and an unpretentious setting. None aspect:food aspect:menu aspect:service aspect:setting Now extract aspects from the following example: input: however , if you want great food at a great price and do n ' t mind the decor , you ca n ' t beat this place . let us extract aspects one by one:	food decor
Definition: The input are sentences about a product or service. The task is to extract the aspects and their corresponding polarity. Here are some examples: example 1- input: I charge it at night and skip taking the cord with me because of the good battery life. The aspects are: battery life, cord None battery life:positive cord:positive example 2- input: Great food, good size menu, great service and an unpretentious setting. The aspects are: food, menu, service, setting None food:positive menu:positive service:positive setting:positive Now predict aspect sentiments from the following example: input: however , if you want great food at a great price and do n ' t mind the decor , you ca n ' t beat this place .The aspects are: food decor let us predict sentiments one by one:	food:positive decor:neutral
Definition: The input are sentences about a product or service. The task is to extract the aspects and their corresponding polarity. Here are some examples: example 1- input: I charge it at night and skip taking the cord with me because of the good battery life. The aspects are: battery life, cord None battery life:good cord:NULL example 2- input: Great food, good size menu, great service and an unpretentious setting. The aspects are: food, menu, service, setting None food:great menu:good service:great setting:unpretentious Now extract opinions for the following example: input:however , if you want great food at a great price and do n ' t mind the decor , you ca n ' t beat this place .The aspects are: food decor let us extract opinions one by one:	food:great food:great decor:NULL

Hình 7: Dữ liệu sau khi xử lý

### 3. Mô hình đề xuất

#### 3.1 Tổng quan về bài toán Aspect-Based Sentiment Analysis

Phân tích tình cảm dựa trên khía cạnh (ABSA) là một bài toán quan trọng trong việc hiểu cảm xúc một cách chi tiết trong các bình luận văn bản của người dùng. Đây là một bài toán cấp cao của bài toán phân tích cảm xúc (Sentiment Analysis) khi bài toán yêu cầu dự đoán cảm xúc cho từng khía cạnh (aspect) trong câu bình luận.

Cụ thể, ABSA trích xuất các khía cạnh và phân loại độ phân cực cảm xúc (polarity) của khía cạnh bằng học sâu để hiểu ý kiến của người dùng. Nói cách khác, bài toán này gồm 2 bài toán con:

- + ATE (Aspect Term Extraction): trích xuất thông tin khía cạnh/thực thể trong của review người dùng
- + APC/ATSC(Asspect Term Sentiment Classification): từ đó đưa ra nhận xét mức độ hài lòng của khách hàng tới từng khía cạnh cụ thể

Bài toán tập trung phân loại các thành phần thành 2 nhãn:

- + Negative
- + Positive

Ví dụ:

$S_i$ : The $\overset{a^1}{\text{price}}$ was $\overset{o^1}{\text{too high}}$ , but the $\overset{a^2}{\text{cab}}$ was $\overset{o^2}{\text{amazing}}$ .		
Subtask	Input	Output
Aspect Term Extraction (ATE)	$S_i$	$a^1, a^2$
Aspect Term Sentiment Classification (ATSC)	$S_i + a^1, S_i + a^2$	$sp^1, sp^2$
Joint Task	$S_i$	$(a^1, sp^1), (a^2, sp^2)$

Với ví dụ này, ta có thể thấy rõ đầu vào của bài toán là một câu văn bản "The cab ride was awesome, but the driver was rude". Khi đó, mô hình cần học và dự đoán dduwwocj đầu ra như sau:  $\{ \text{'term': 'cab ride', 'polarity': 'positive'} \}$ ,  $\{ \text{'term': 'driver', 'polarity': 'negative'} \}$ .

## 3.2 Mô hình triển khai

### 3.2.1 Tổng quan về Transfer Learning

Học chuyển giao (Transfer Learning) là tác vụ mà trong đó một mô hình được huấn luyện trước với một kho dữ liệu giàu có trước khi được điều chỉnh (finetune) trong một nhiệm vụ tiếp theo.

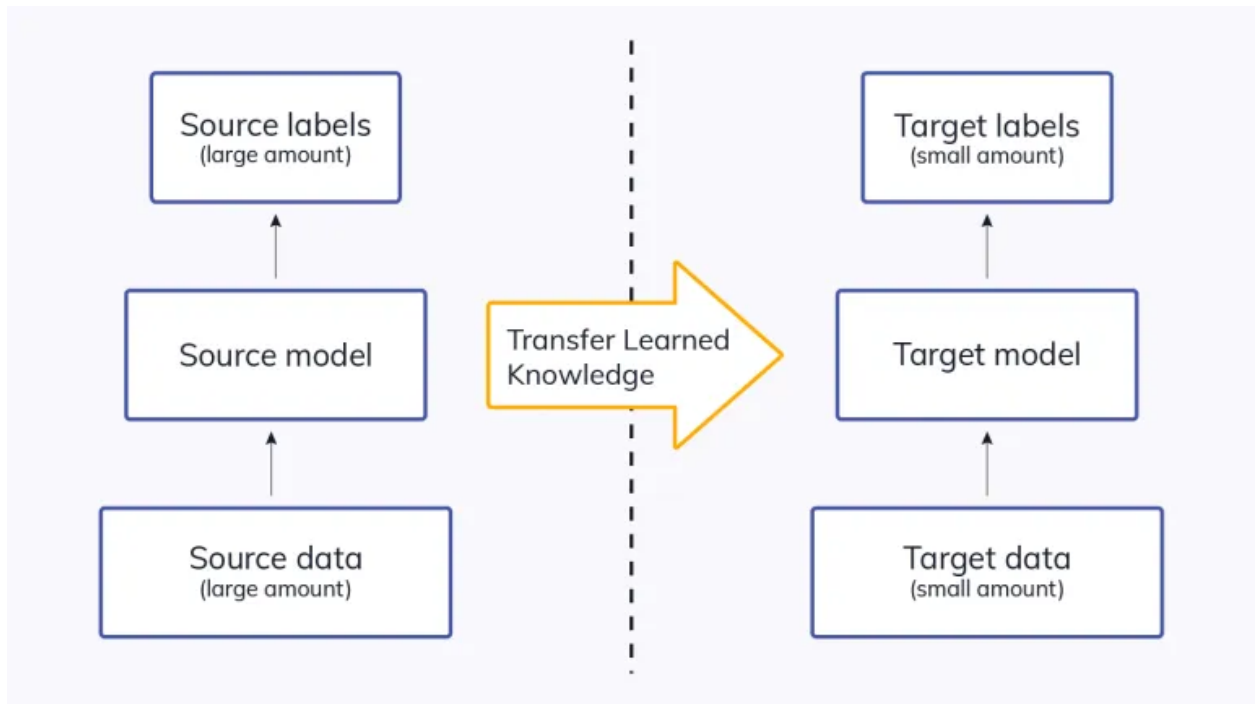
Học chuyển giao đã nổi lên như một kỹ thuật mạnh mẽ trong ngôn ngữ tự nhiên xử lý (NLP). Hiệu quả của học tập chuyển giao đã dẫn đến sự đa dạng của cách tiếp cận, phương pháp và thực hành đối với các tác vụ NLP như: dịch máy (Machine Translation), trích rút thực thể (NER), phân tích cảm xúc (Sentiment Analysis),...

Phương pháp này rất hữu ích khi ta chỉ có một lượng nhỏ dữ liệu huấn luyện cho tác vụ mới hoặc khi tác vụ mới có một lượng dữ liệu khác nhau so với tác vụ gốc mà mô hình đã được huấn luyện. Thay vì bắt đầu từ đầu và huấn luyện một mô hình từ đầu, Transfer Learning cho phép chúng ta tận dụng kiến thức đã được học từ mô hình gốc và chỉ điều chỉnh hoặc "chuyển giao" một số trọng số trong mô hình cho tác vụ mới.

Cách thông thường để thực hiện Transfer Learning là sử dụng mô hình đã được huấn luyện trước đó (thường được huấn luyện trên một tác vụ lớn và phức tạp như ImageNet) và sau đó thay đổi một số lớp cuối cùng của mô hình và huấn luyện lại chúng trên tác vụ mới. Việc này giúp mô hình học được các đặc trưng chung và trừu tượng hơn từ dữ liệu gốc và áp dụng chúng vào tác vụ mới.

Transfer Learning giúp tăng tốc quá trình huấn luyện, giảm yêu cầu về dữ liệu huấn luyện và cải thiện khả năng tổng quát hóa của mô hình trong các tác vụ mới.





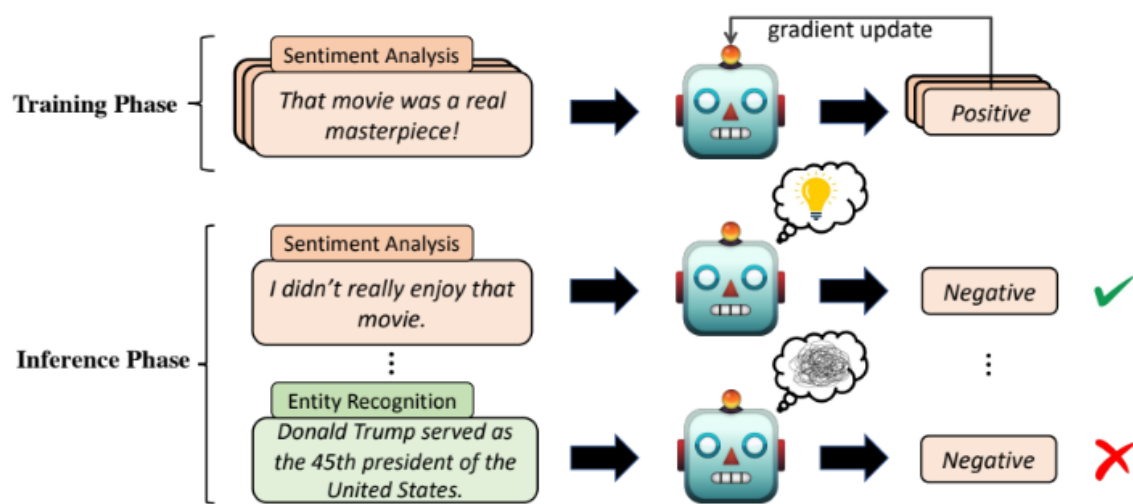
*Hình ảnh mô tả cơ chế học Transfer Learning*

### 3.2.2. Tổng quan về Instruction Learning

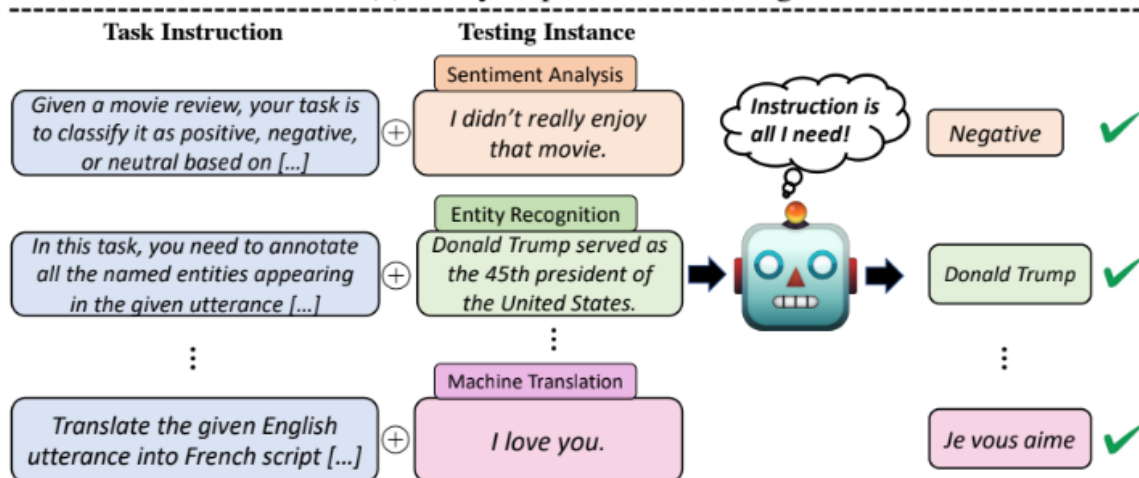
Instruction Learning trong xử lý ngôn ngữ tự nhiên (NLP) đề cập đến nhiệm vụ huấn luyện các mô hình học máy để hiểu và thực hiện các hướng dẫn bằng ngôn ngữ tự nhiên. Mục tiêu là phát triển các thuật toán có thể hiểu các hướng dẫn từ con người và thực hiện các hành động mong muốn tương ứng.

Instruction Learning bao gồm các công việc phụ như hiểu ngôn ngữ tự nhiên (NLU), phân tích ngữ nghĩa và sinh ngôn ngữ. NLU tập trung vào việc trích xuất ý nghĩa từ các hướng dẫn bằng cách xác định các thực thể, mối quan hệ và hành động chính được mô tả trong văn bản. Phân tích ngữ nghĩa liên quan đến ánh xạ các hướng dẫn đã được phân tích thành một biểu diễn có cấu trúc, chẳng hạn như các biểu thức logic hoặc các chương trình có thể thực thi được bởi máy. Sinh ngôn ngữ liên quan đến việc tạo ra câu trả lời hoặc hành động phù hợp và mạch lạc dựa trên các hướng dẫn đã được phân tích.

Quá trình Instruction Learning thường liên quan đến việc sử dụng dữ liệu huấn luyện bao gồm các cặp hướng dẫn và hành động hoặc các minh chứng tương ứng. Các tập dữ liệu này là cơ sở để huấn luyện các mô hình học máy, chẳng hạn như các mạng nơ-ron sâu, để học cách ánh xạ giữa các hướng dẫn bằng ngôn ngữ tự nhiên và các hành động mong muốn.



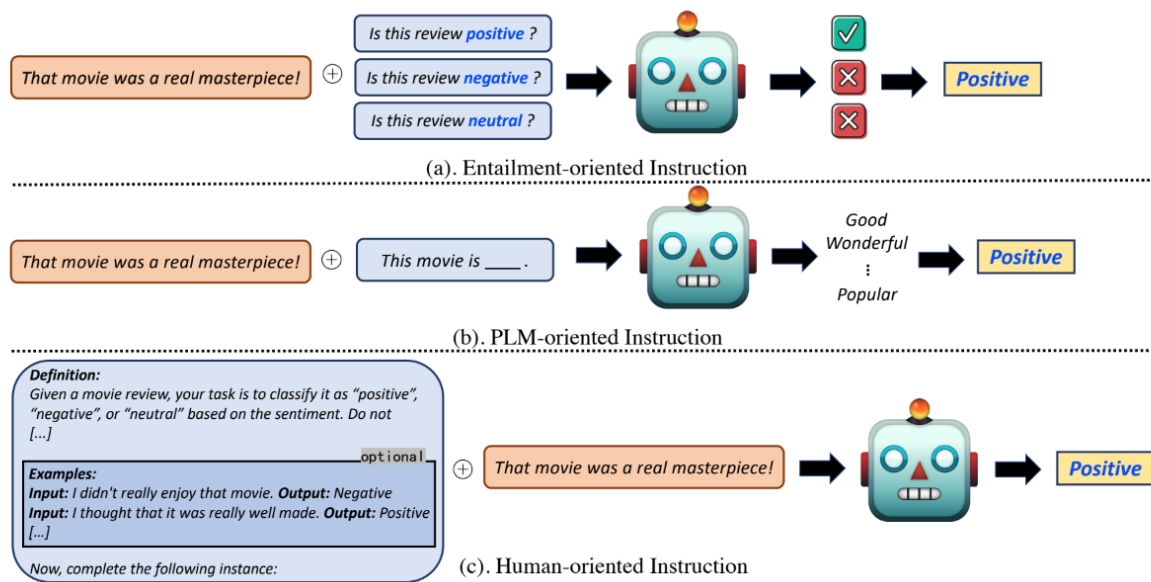
(a). Fully Supervised Learning



(b). Instruction Learning

Hình ảnh so sánh giữa Fully Supervised Learning và Instruction Learning

Có 3 hướng tiếp cận chính trong Instruction Learning, được thể hiện trong hình dưới đây:



### 3 hướng tiếp cận chính trong Instruction Learning

Trong bài toán này, em sử dụng cách tiếp cận số 3 bằng cách sử dụng InstructDatasetLoader

#### 3.2.3. Tổng quan về Transformers

##### a, Kiến trúc mô hình Transformers

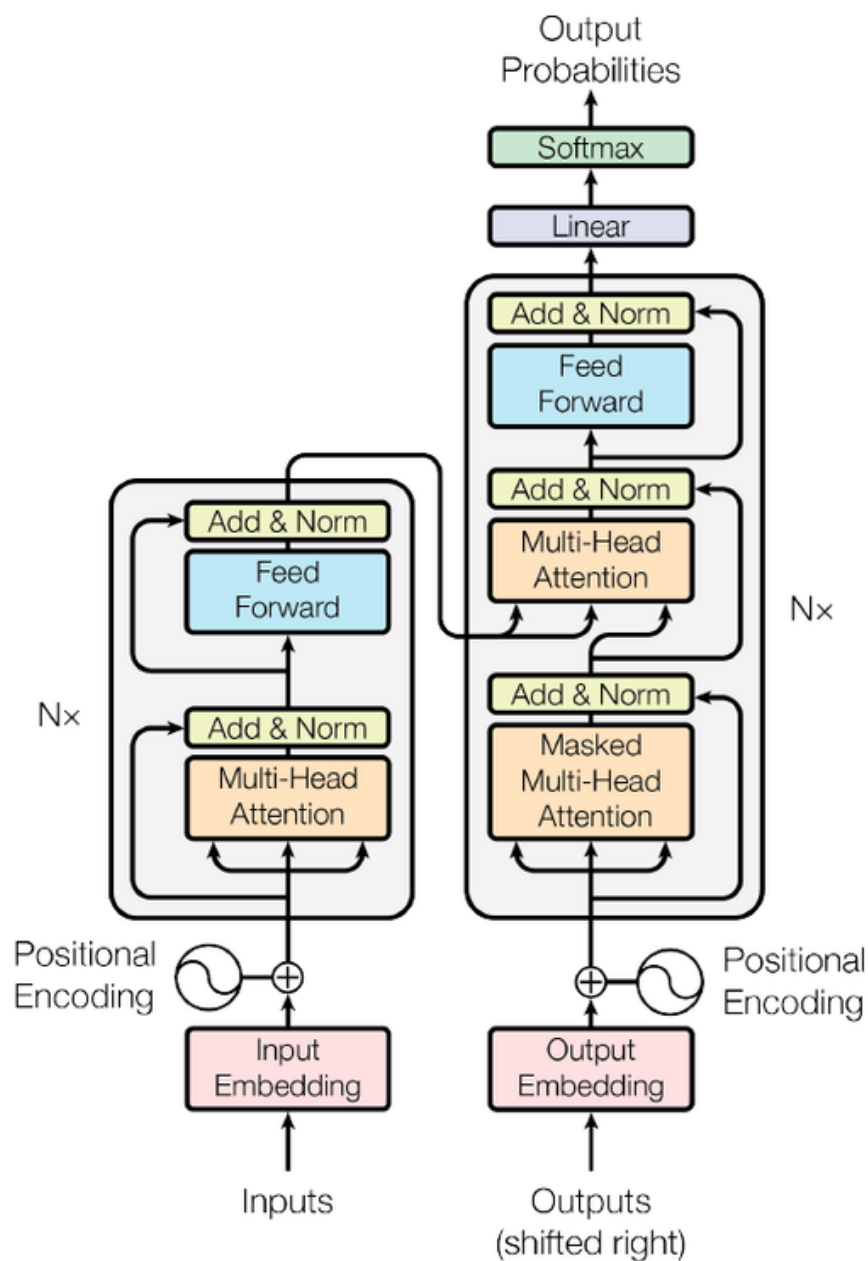
Giống như một mô hình ngôn ngữ Sequence2Sequence, kiến trúc tổng quan của mô hình transformer bao gồm 2 phần lớn là encoder và decoder. Encoder dùng để học vector biểu của câu với mong muốn rằng vector này mang thông tin hoàn hảo của câu đó. Decoder thực hiện chức năng chuyển vector biểu diễn kia thành ngôn ngữ đích.

- Encoder: là tổng hợp xếp chồng lên nhau của 6 layers xác định. Mỗi layer bao gồm 2 layer con (sub-layer) trong nó. Sub-layer đầu tiên là multi-head self-attention mà lát nữa chúng ta sẽ tìm hiểu. Layer thứ 2 đơn thuần chỉ là các fully-connected feed-forward layer. Chắc mình sẽ không giải thích thêm về fully-connected feed-forward layer nữa vì chúng là khái niệm quá cơ bản trong machine learning. Các bạn có thể tìm hiểu thêm ở bài viết chi tiết này Multi-layer Perceptron của machinelearningcoban. Một lưu ý là chúng ta sẽ sử dụng một kết nối residual ở mỗi sub-layer ngay sau layer normalization. Kiến trúc này có ý tưởng tương tự như mạng resnet trong CNN. Đầu ra của mỗi sub-layer là có số chiều là 512 theo như bài viết.

- Decoder: Decoder cũng là tổng hợp xếp chồng của 6 layers. Kiến trúc tương tự như các sub-layer ở Encoder ngoại trừ thêm 1 sub-layer thể hiện phân phối attention ở vị trí đầu tiên. Layer này không gì khác so với multi-head self-attention layer ngoại trừ được điều chỉnh để không đưa các từ trong tương lai vào attention. Tại bước thứ  $i$  của decoder chúng ta chỉ biết được các từ ở vị trí nhỏ hơn nên việc điều chỉnh đảm bảo attention chỉ áp dụng cho những từ nhỏ hơn vị trí thứ  $i$ . Cơ chế residual 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  $[0, 1]$  vector vị trí hoặc sử dụng hàm sin, cos.

Một trong những ưu điểm của transformer là mô hình này có khả năng xử lý song song cho các từ. Encoders của mô hình transformer là một dạng feedforward neural nets, bao gồm nhiều encoder layer khác, mỗi encoder layer này xử lý đồng thời các từ. Trong khi đó, với mô hình LSTM, thì các từ phải được xử lý tuần tự.

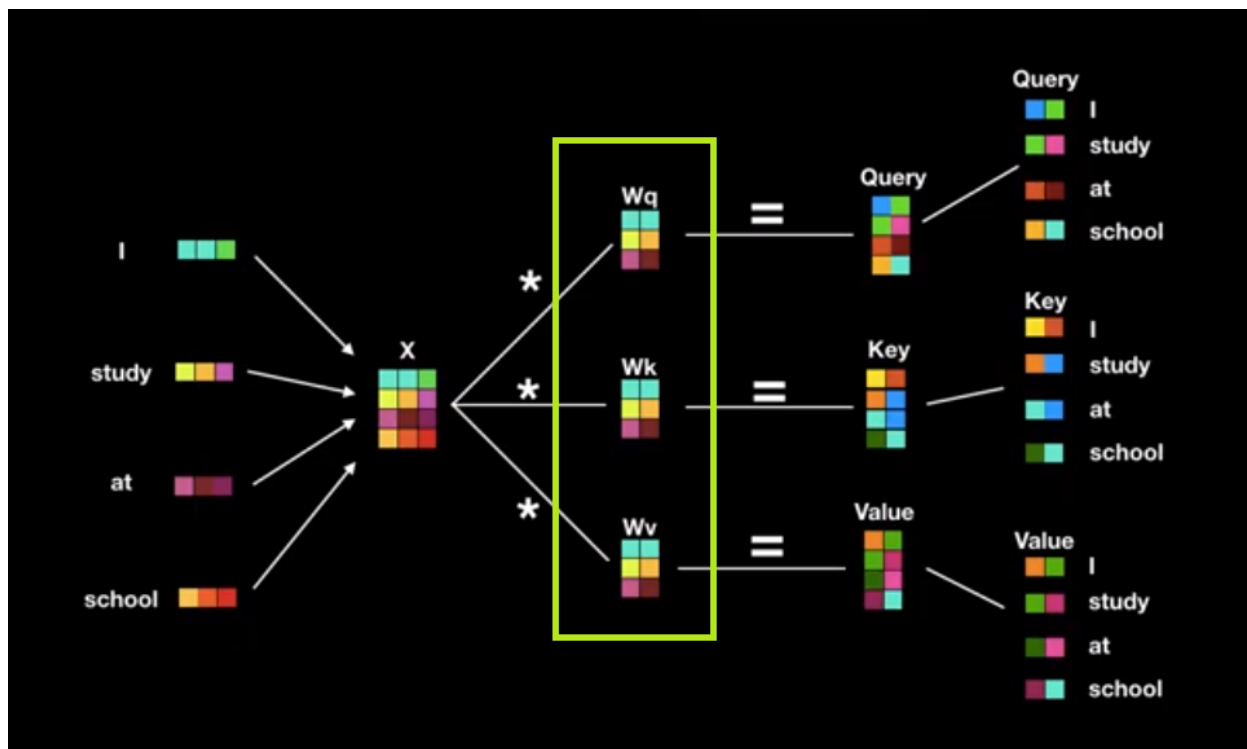


Mô hình transformer trong xử lý ngôn ngữ

#### b, Cơ chế Attention trong Transformers

Đây chính là một cơ chế self-attention khi mỗi từ có thể điều chỉnh trọng số của nó cho các từ khác trong câu sao cho từ ở vị trí càng gần nó nhất thì trọng số càng lớn và càng xa thì càng nhỏ dần. Sau bước nhúng từ (đi qua embedding layer) ta có đầu vào của encoder và decoder là

ma trận kích thước  $m \times n$ , với  $m, n$  lần lượt là độ dài câu và số chiều của một vector nhúng từ.



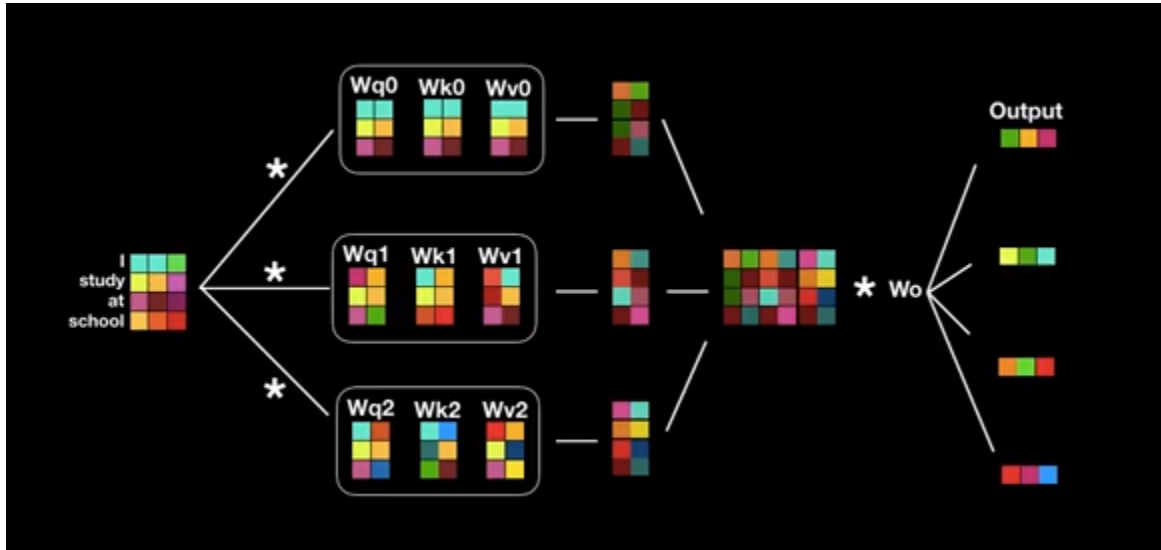
Đầu vào để tính attention sẽ bao gồm ma trận **Q** (mỗi dòng của nó là một vector query đại diện cho các từ input), ma trận **K** (tương tự như ma trận **Q**, mỗi dòng là vector key đại diện cho các từ input). Hai ma trận **Q, K** được sử dụng để tính attention mà các từ trong câu trả về cho 1 từ cụ thể trong câu. attention vector sẽ được tính dựa trên trung bình có trọng số của các vector value trong ma trận **V** với trọng số attention (được tính từ **Q, K**).

Trong thực hành chúng ta tính toán hàm attention trên toàn bộ tập các câu truy vấn (query) một cách đồng thời được đóng gói thông qua ma trận **Q**, keys và values cũng được đóng gói cùng nhau thông qua matrix **K, V**. Phương trình Attention như sau:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) V$$

Việc chia cho là số dimension của vector key nhằm mục đích tránh tràn luồng nếu số mũ là quá lớn. Như vậy sau quá trình Scale dot production chúng ta sẽ thu được 1 ma trận attention. Các tham số mà model cần tinh chỉnh chính là các ma trận **Wq, Wk, Wv**. Mỗi quá trình như vậy được gọi là 1 head của attention. Khi lặp lại quá trình này nhiều lần ta sẽ thu được quá trình Multi-head Attention

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{concatenate}(\text{head}_1, \text{head}_2, \dots, \text{head}_h) \mathbf{W}_0$$



Hình ảnh mô tả Multi-head Attention

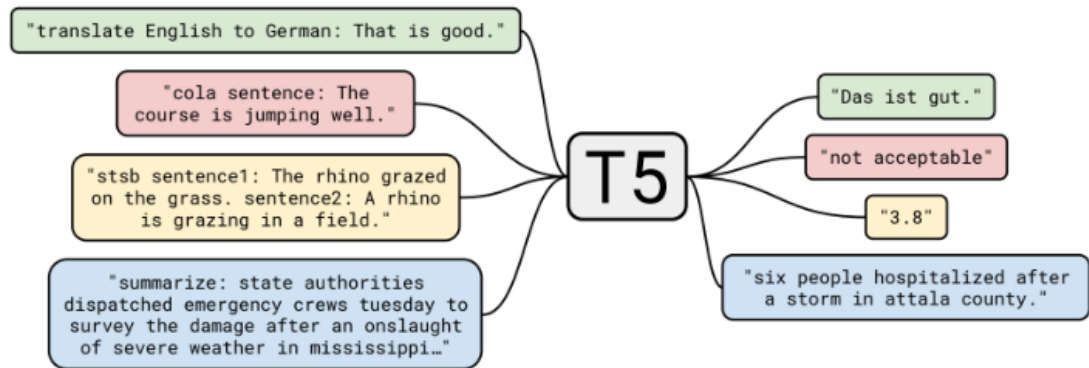
### 3.2.4. Text-to-text Transfer Transformers

#### a, Giới thiệu

Mô hình T5 (Text-to-Text Transfer Transformers) là một mô hình học sâu dựa trên kiến trúc Transformers, được giới thiệu bởi công ty Google vào năm 2019. T5 là một mô hình đa nhiệm (multi-task) và có khả năng thực hiện nhiều tác vụ liên quan đến xử lý ngôn ngữ tự nhiên (NLP). Mô hình này được giới thiệu lần đầu tiên qua bài báo [Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer](#) bởi Colin Raffel.

T5 được xây dựng dựa trên kiến trúc Transformers, kiến trúc này đã đạt được thành công đáng kể trong nhiều nhiệm vụ ngôn ngữ, như dịch máy, phân loại văn bản, và sinh văn bản. T5 mở rộng kiến trúc Transformer để có thể thực hiện nhiều loại tác vụ chỉ bằng cách thay đổi dữ liệu đầu vào và đầu ra của mô hình.

Thay vì chỉ dựa trên dữ liệu đầu vào là văn bản và đầu ra là dự đoán hoặc sinh văn bản (sequence-to-sequence), T5 chuyển đổi mọi tác vụ ngôn ngữ thành một định dạng chung là "text-to-text", trong đó cả dữ liệu đầu vào và đầu ra đều là văn bản. Điều này cho phép T5 thực hiện các tác vụ khác nhau, bao gồm dịch máy, phân loại, tạo câu hỏi, trả lời câu hỏi, và nhiều nhiệm vụ khác chỉ bằng cách định dạng dữ liệu đầu vào và đầu ra tương ứng.



Hình ảnh mô hình T5 với tác vụ text-to-text

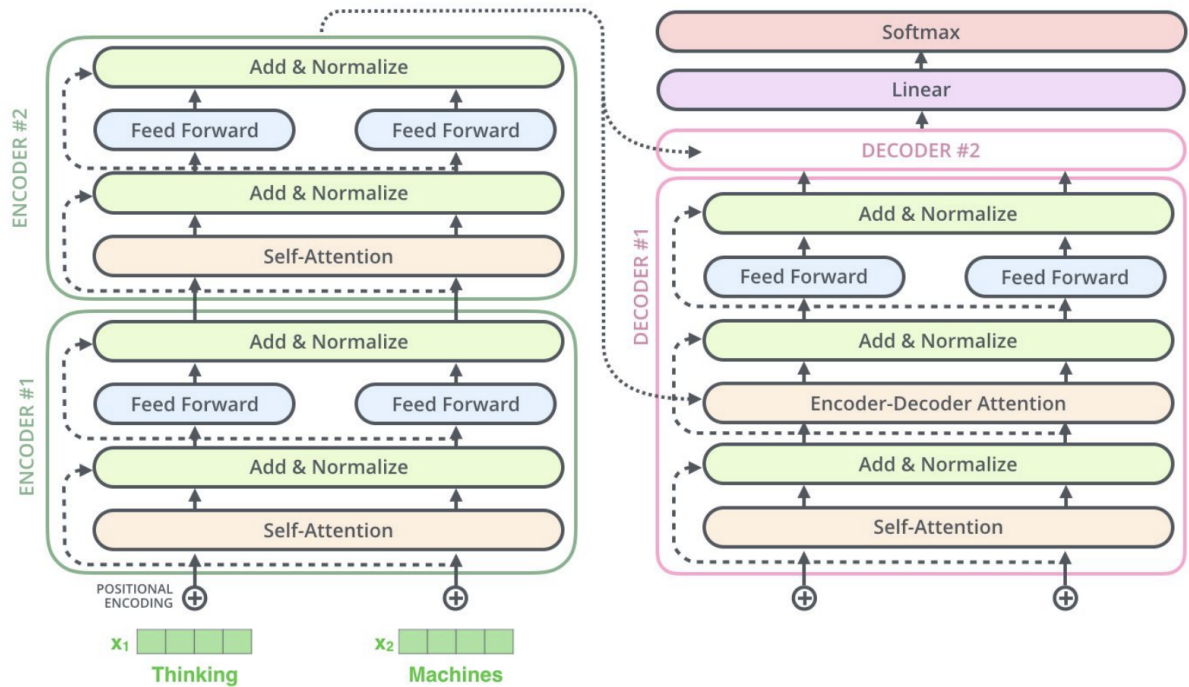
Mô hình T5 đã đạt được kết quả ấn tượng trong nhiều bài toán ngôn ngữ và đã trở thành một trong những mô hình quan trọng trong lĩnh vực xử lý ngôn ngữ tự nhiên.

## b, Kiến trúc mô hình T5

- Mã hóa đầu vào: Đầu vào của mô hình T5 là một chuỗi các token đại diện cho văn bản nguồn. Các token này thường là các đơn vị phụ từ (subword units) được tạo ra bởi một tokenizer như Byte Pair Encoding (BPE) hoặc SentencePiece.
- Mã hóa chung (Shared Encoder): T5 sử dụng một mã hóa chung cho tất cả các tác vụ. Mã hóa chung bao gồm các lớp self-attention được xếp chồng lên nhau, mỗi lớp có khả năng tập trung vào toàn bộ chuỗi đầu vào. Điều này cho phép mô hình nắm bắt các mối quan hệ ngữ cảnh giữa các token trong văn bản nguồn.
- Bộ chuyển đổi riêng cho từng tác vụ (Task-Specific Adapter): Sau quá trình mã hóa chung, T5 sử dụng các bộ chuyển đổi riêng cho từng tác vụ. Các bộ chuyển đổi này là các mô-đun nhỏ được thêm vào mô hình để tùy chỉnh mã hóa chung cho các tác vụ khác nhau mà không cần thay đổi cấu trúc lớn của mô hình.
- Giải mã đầu ra: T5 sử dụng một bộ giải mã cho các tác vụ yêu cầu tạo ra một chuỗi kết quả. Bộ giải mã tương tự như bộ mã hóa, nhưng còn bao gồm các lớp cross-attention để tập trung vào các biểu diễn mã hóa đầu vào. Trong quá trình huấn luyện, bộ giải mã dự đoán chuỗi kết quả theo kiểu tự sinh, từng token một.



- Tiền huấn luyện và Tinh chỉnh (Pretraining and Fine-Tuning): T5 thực hiện hai giai đoạn: tiền huấn luyện và tinh chỉnh. Trong giai đoạn tiền huấn luyện, mô hình được đào tạo trên một tập dữ liệu lớn bằng cách giải quyết bài toán dự đoán từ ngữ thuật ngữ trước, trong đó một số token được che đi (mask) và mô hình phải dự đoán chúng. Trong giai đoạn tinh chỉnh, mô hình đã được tiền huấn luyện được huấn luyện thêm trên các tác vụ cụ thể với các tập dữ liệu và mục tiêu riêng cho từng tác vụ.



Kiến trúc tổng quan T5 model

## 4. Thực nghiệm mô hình

### 4.1. Môi trường thực nghiệm và mô hình cài đặt

- Môi trường thực nghiệm sử dụng Google Collab, Python version 3.10
- Mô hình cài đặt:
  - + Sử dụng model pretrained **t5-base** của Google, sử dụng thư viện Huggingface, Transformers,.. Mô hình này đã được pretrained với rất nhiều tác vụ supervised và unsupervised learning. Đặc biệt mô hình này đã được pretrained tác vụ Sentiment Analysis với bộ dữ liệu SST-2 [Socher et al., 2013](#)
  - + Các tham số cụ thể như sau:

```
"d_ff": 3072,  
"d_kv": 64,  
"d_model": 768,  
"decoder_start_token_id": 0,  
"dense_act_fn": "relu",  
"dropout_rate": 0.1,  
"eos_token_id": 1,  
"feed_forward_proj": "relu",  
"initializer_factor": 1.0,  
"is_encoder_decoder": true,  
"is_gated_act": false,  
"layer_norm_epsilon": 1e-06,  
"max_length": 128,  
"model_type": "t5",  
"n_positions": 512,  
"num_decoder_layers": 12,  
"num_heads": 12,  
"num_layers": 12,  
"output_past": true,  
"pad_token_id": 0,
```

"relative\_attention\_max\_distance": 128,  
"relative\_attention\_num\_buckets": 32,  
"vocab\_size": 32128.

## 4.2 Kết quả thực nghiệm

Kết quả training trên các tập Laptop14, Restaurant14, Restaurant15, Restaurant16 được ghi lại như sau:

Laptop14

Epoch	Training Loss	Validation Loss
1	No log	0.756559
2	No log	0.483671
3	No log	0.404070
4	No log	0.371508
5	No log	0.354967
6	No log	0.345324

Restaurant14

Epoch	Training Loss	Validation Loss
1	No log	0.297349
2	No log	0.235211
3	No log	0.207909
4	No log	0.206483
5	No log	0.205758
6	No log	0.204038

Restaurant15

Epoch	Training Loss	Validation Loss
1	No log	0.367091
2	No log	0.316285
3	No log	0.292715
4	No log	0.282691
5	No log	0.291322
6	No log	0.290165

Restaurant16

**Epoch    Training Loss    Validation Loss**

1	No log	2.731566
2	No log	0.788149
3	No log	0.544894
4	No log	0.482268
5	No log	0.461622
6	No log	0.461416

	accuracy	precision	recall	f1
Laptop14	0.6273	0.3525	0.3601	0.3521
Restaurant14	0.7642	0.5887	0.5896	0.5884
Restaurant15	0.6884	0.5117	0.5223	0.5173
Restaurant16	0.5536	0.2995	0.3126	0.3023

## 5. Kết luận và hướng phát triển

Qua đề tài này, em đã tìm hiểu được các lý thuyết liên quan đến, Transfer Learning, Instruction Learning giúp xử lý các tác vụ xử lý ngôn ngữ tự nhiên (NLP), thực hiện triển khai cài đặt được mô hình Aspect-based Sentiment Analysis sử dụng một model T5 đã được pretrained, và benchmark trên 4 tập dataset có sẵn.

Tuy vậy, trong quá trình thực nghiệm, em gặp khá nhiều vấn đề về phần cứng, và cài đặt các độ đo mô hình sao cho phù hợp. Kết quả thực nghiệm cũng cho thấy độ hiệu quả mô hình với từng tác vụ là khác nhau, khi mà mô hình thể hiện khá tốt ở 2 tác vụ ATE và APC, nhưng không đủ tốt trong tác vụ dự đoán opinion và category.

Trong thời gian sắp tới, em sẽ thực hiện benchmark với những mô hình pretrained lớn với nhiều tham số hơn, thay đổi một số cấu hình mô hình và benchmark lại đối với từng tác vụ.

## 6. Tài liệu tham khảo

[1] Is Prompt All You Need? No. A Comprehensive and Broader View of Instruction Learning

*Renze Lou, Kai Zhang, Wenpeng Yin*

[2] Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer

*Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J. Liu*

[3] InstructABSA: Instruction Learning for Aspect Based Sentiment Analysis

*Kevin Scaria, Himanshu Gupta, Siddharth Goyal, Saurabh Arjun Sawant, Swaroop Mishra, Chitta Baral*

[4] PyABSA: A Modularized Framework for Reproducible Aspect-based Sentiment Analysis

*Heng Yang, Ke Li*