

**BỘ CÔNG THƯƠNG**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP HCM**



**BÁO CÁO BÀI TẬP LỚN**  
**MÔN: XỬ LÝ NGÔN NGỮ TỰ NHIÊN**

**Chủ đề:** Tóm tắt văn bản tiếng việt  
(Text Summarization Vietnamese)

**GVHD:** Đặng Thị Phúc  
**Nhóm thực hiện:** DINOSAUR  
**Lớp:** DHKHD15A

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP. HỒ CHÍ MINH

**BÁO CÁO BÀI TẬP LỚN**  
**MÔN: XỬ LÝ NGÔN NGỮ TỰ NHIÊN**

**Tên nhóm: DINOSAUR**

**Lớp: DHKHD15A**

**Thành viên nhóm:**

Họ và tên	Mã số sinh viên
1. Lê Thanh Phong	19475611
2. Trần Tuấn Vũ	19474281
3. Đoàn Minh Trường	19519011

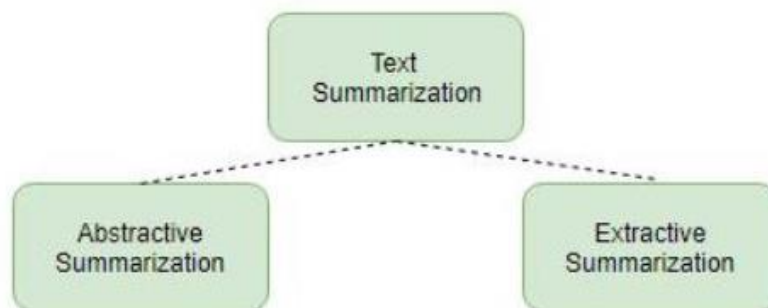
# BÁO CÁO BÀI TẬP LỚN THƯỜNG KỲ 1

## I. Giới thiệu đề tài

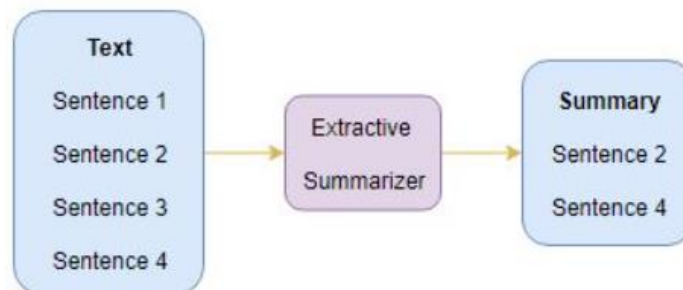
### 1. Tóm tắt văn bản là gì ?

Tóm tắt văn bản là quá trình rút trích những thông tin quan trọng nhất từ một văn bản để tạo ra phiên bản ngắn gọn, xúc tích mang lại đầy đủ lượng thông tin của văn bản gốc kèm theo đó là tính đúng đắn về ngữ pháp và chính tả. Bản tóm tắt phải giữ được những thông tin quan trọng của toàn bộ văn bản chính. Bên cạnh đó, bản tóm tắt cần phải có bố cục chặt chẽ có tính đến các thông số như độ dài câu, phong cách viết và cú pháp của văn bản.

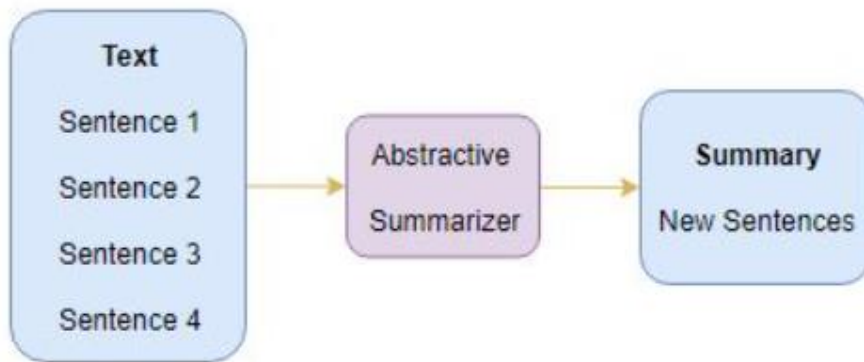
Hai cách tiếp cận nổi bật là tóm tắt trích xuất (**Extractive Summarization**) và tóm tắt trừu tượng (**Abstractive Summarization**).



**Extractive Summarization** tạo thành tóm tắt bằng cách sao chép các phần của văn bản nguồn thông qua một số biện pháp về mức độ quan trọng và sau đó kết hợp các phần/câu đó với nhau để tạo ra một bản tóm tắt. Tầm quan trọng của câu dựa trên các đặc điểm ngôn ngữ và thống kê của nó.



**Abstractive Summarization** tạo ra các cụm từ mới, có thể diễn đạt lại hoặc sử dụng các từ không có trong văn bản gốc. Các cách tiếp cận trừu tượng tự nhiên khó hơn. Để có một bản tóm tắt trừu tượng hoàn hảo, trước tiên người mẫu phải thực sự hiểu tài liệu và sau đó cố gắng diễn đạt sự hiểu biết đó dưới dạng ngắn gọn, có thể sử dụng các từ và cụm từ mới. Điều này khó hơn nhiều so với một bản tóm tắt chiết xuất, đòi hỏi các khả năng phức tạp như khái quát hóa, diễn giải và kết hợp kiến thức trong thế giới thực.



## 2. Lí do chọn đề tài

Trong một thời đại mà mỗi ngày, mỗi giờ, mỗi phút đều có một lượng thông tin khổng lồ được sinh ra nhưng giới hạn về thời gian, về khả năng đọc và tiếp thu của con người là có hạn, việc hiểu và nắm bắt thật nhiều thông tin một cách nhanh chóng không phải là vấn đề đơn giản với bất kỳ ai.

Thế bạn đã bao giờ dành hàng giờ đồng hồ để đọc các bài báo điện tử trên các trang mạng xã hội để nắm bắt tin tức một cách nhanh nhất. Để giải quyết vấn đề đó nhóm đã chọn đề tài tóm tắt văn bản tiếng việt giúp cho người đọc có thể giảm thiểu thời gian đọc các bài báo để nắm rõ nội dung, kiến thức mới để dành thời gian cho các công việc khác, mà vẫn có thể nắm bắt được gầy gọn những nội dung của nó.

## II. Các bước thực hiện

### 1. Chuẩn bị dữ liệu

Bộ dữ liệu bao gồm hơn 100.000 dòng dữ liệu được thu thập từ các trang báo điện tử trên tất cả các lĩnh vực: [vnexpress.net](http://vnexpress.net), [dantri.com.vn](http://dantri.com.vn), [thethao247.vn](http://thethao247.vn) ...

Ở đây nhóm sẽ minh họa một vài trang báo điện tử mà nhóm đi thu thập dữ liệu:

Trang báo [vnexpress.net](http://vnexpress.net)

VnExpress - Báo tiếng Việt nhiều

vnexpress.net

VNEXPRESS

Thứ hai, 16/5/2022

Mới nhất

International

Tìm kiếm

Đăng nhập

Thời sự

Góc nhìn

Thế giới

Video

Podcasts

Kinh doanh

Khoa học

Giải trí

Thể thao

Pháp luật

Giáo dục

Sức khỏe

Đời sống

Du lịch

Số hóa

Xe

Ý kiến

Tâm sự

Hài

Tất cả

Malaysia gặp Việt Nam ở bán kết SEA Games

NAM ĐỊNH-- Bị Campuchia cầm hòa 2-2 ở lượt cuối chiều 16/5, Malaysia mất đỉnh bảng B bóng đá nam SEA Games 31.

"Ghi bàn: Fayyadh 51' (pen), 68' - Chanchav 45' (pen), Rina 61'.Do đã ...

197

Việt Nam giành thêm 20 HC vàng ngày 16/5

Điền kinh, Pencak Silat, Taekwondo đóng góp lớn giúp Việt Nam tiếp tục dẫn đầu toàn đoàn tại SEA Games 31 với 88 HC vàng.

114

Bị huých ngã, chân chạy Việt Nam mất HC vàng

HÀ NỘI-- Đường kim vô địch Đinh Thị Bích ngả sấp sau cú đẩy tay của đối thủ Malaysia, lỡ cơ hội bảo vệ HC vàng 800m nữ chiều 16/5.

33

Góc nhìn

Tin vào tin giả

Những "mồi nhử nhấp chuột" của tin giả thách thức sự thông thái của bạn hàng ngày, hàng giờ.

Nguyễn Thị Hồng Chi

27

Sparkling QUYNHON

MUA VÉ

Các nhà khoa học tế tyu trước thêm lễ trao giải Sáng kiến Khoa học 2022

HÀ NỘI-- Chiều 16/5, các tác giả dự

Kinh doanh

Quốc tế

Doanh nghiệp

Chứng khoán

Bất động sản

Bảo hiểm

E-Commerce 4.0

Sắp có giải pháp ngân cổ

Thanh tra các công ty

## Trang báo dantri

Tin tức Việt Nam và quốc tế nổi

dantri.com.vn

D

International Version

Hà Nội

Thứ 2, 16/05/2022

24°C

Đăng nhập

VIDEO SỰ KIỆN XÃ HỘI THẾ GIỚI KINH DOANH BẤT ĐỘNG SẢN THỂ THAO VIỆC LÀM NHÂN ÁI SỨC KHỎE VĂN HÓA GIẢI TRÍ XE ++ SỨC MẠNH SỐ GIÁO DỤC AN SINH PHÁP LUẬT ...

Tổng Bí thư: Tổng thống Hy Lạp thăm Việt Nam là dấu mốc mới

Nga đồng ý cho thương binh Ukraine sơ tán khỏi "pháo đài" Azovstal

Cụ Chủ tịch Hạ Long đứng tên xe giá trị thấp nhất trong 4 ô tô bị tạm giữ

Trong 4 xe sang bị công an tạm giữ, ông Phạm Hồng Hà - nguyên Chủ tịch TP Hạ Long, nguyên Trưởng Ban quản lý vịnh Hạ Long (tỉnh Quảng Ninh) đứng tên chiếc xe có giá trị thấp nhất.

Nhiều người ngã trên cầu Sài Gòn vì đau nhót rơi vãi, CSGT ứng cứu kịp thời

Vụ Tịnh thất Bồng Lai: Công an đang truy tìm Võ Thị Diễm My

U23 Thái Lan đấu bảng, U23 Việt Nam đụng Malaysia ở bán kết SEA Games

Nhà đầu tư chứng khoán "ra đảo", Bộ Tài chính vào cuộc

DÂN TRÍ SPOTLIGHT

Ông Nông Quốc Tuấn được bổ nhiệm lại chức Phó Chủ nhiệm Ủy ban Dân tộc

THE GLOBAL CITY

Foster + Partners

Nhóm sử dụng thư viện scrapy để thu thập dữ liệu, và sau đây là code minh họa cho một trang báo điện tử

```
1 import scrapy
2 class post(scrapy.Item):
3     Title = scrapy.Field()
4     Summary = scrapy.Field()
5     Text = scrapy.Field()
6     Domain = scrapy.Field()
7 class crawling_data(scrapy.Spider):
8     name = "crawl_data"
9     def start_request(self):
10         path = "C:\\Users\\ASUS\\Documents\\Zalo Received Files\\dantri.txt"
11         f = open(path, 'r')
12         for link in f:
13             for i in range(1, 31):
14                 total_url = link + str(i) + '.htm'
15                 yield scrapy.Request(url=total_url, callback=self.parse)
16     def parse(self, response):
17         ''' get xpath href'''
18         list_xpath = response.xpath('//*[@id="bai-viet"]/div[1]/article/article/div[2]/h3/a/@href').extract()
19         for url in list_xpath:
20             string = "https://dantri.com.vn"
21             urls = string + str(url)
22             yield scrapy.Request(url=urls, callback=self.parse_post)
23     def parse_post(self, response):
24         items = post()
25         '''Summary'''
26         sum = response.xpath('//h2[@class="singular-sapo"]/text()').get()
27         items['Summary'] = sum
28         '''domains'''
29         do_main = response.xpath('//ul[@class="breadcrumbs"]').get()
30         items['Domain'] = do_main
31         '''text'''
32         text = response.xpath('//div[@class="singular-content"]').extract()
33         items['Text'] = text
34         yield items
```

Và chúng ta có dữ liệu nhưng còn các thẻ, nhóm phải xóa đi các thẻ đó

```
In [10]: data.iloc[23500]['Text']
Out[10]: '<p class="Normal">Theo Coinmarketcap, giá Bitcoin rạng sáng nay (22/8) đã lên hơn 49.600 USD - mức cao nhất từ ngày 16/5. Hiện tại, đồng tiền số lớn nhất thế giới giảm nhẹ khi giao dịch quanh mốc gần 49.000 USD. Một tuần qua, giá Bitcoin đã tăng gần 4%.</p><p class="Normal">"Mức kháng cự lớn tiếp theo là vùng 50.000 USD. Đợt tăng giá gần đây không phải là dấu hiệu của một bước nhảy vọt lớn. Tuy nhiên, nếu nhiều người mua lao vào để đẩy giá lên trên mức 50.000 USD, thì một đợt tăng giá diễn cường có thể xảy ra, mục tiêu giá trung hạn có thể là 55.000 USD", Konstantin Anissimov, CEO sàn giao dịch tiền điện tử CEX.IO nhận định.</p>><p class="Normal">Sức mua Bitcoin tăng mạnh sau thông tin ngân hàng Wells Fargo đã đăng ký một quỹ Bitcoin tư nhân với các cơ quan quản lý của Mỹ. Tương tự, quỹ Bitcoin của JPMorgan cũng đã được trình lên Ủy ban Chứng khoán Mỹ (SEC).</p><p class="Normal">Bitcoin đã phục hồi sau khi được giao dịch quanh mức 30.000 đến 40.000 USD trong suốt nhiều tuần, sau khi nó lao dốc từ mức kỷ lục gần 65.000 hồi giữa tháng 4.</p><p class="Normal">Đù vậy, đồng tiền này vẫn tăng đáng kể trong năm qua khi các tổ chức tên tuổi chấp nhận Bitcoin ngày càng nhiều. Theo Chainalysis, việc áp dụng tiền điện tử toàn cầu đã tăng khoảng 881% trong 12 tháng qua.</p><p class="Normal" style="text-align:right"><strong>Tú Anh</strong><em> (theo Bloomberg)</em></p>'
```

Một đoạn code để xóa các thẻ

```
def preprocessing_text(string):
    if '<p class="Normal">' in str(string):
        temp = []
        if str(string) != np.nan:
            soup = BeautifulSoup(str(string), 'html.parser')
            for element in soup.find_all('p', {'class': 'Normal'}):
                temp.append(element.text)
            text = ' '.join(str(j) for j in temp)
            return text
        else:
            return string
    elif '<div class="singular-content">' in str(string):
        list_string = []
        soup = BeautifulSoup(str(string), 'html.parser')
        for elements in soup:
            try:
                list_string.append(elements.text)
                return list_string
            except:
                list_string.append('0')
        return list_string
    else:
        return string

def convertListtostring(string):
    if type(string) == list:
        text = ' '.join(str(e) for e in string)
        return text
    else:
        return string

def preprocessing_summary(string):
    if '(Dân trí)' in str(string):
        if '-' in string:
            split_string = string.split('-')[1]
            return split_string
        else:
            return string
    else:
        return string
```



Sau khi thu thập xong dữ liệu nhóm sẽ lưu tất cả vào một file csv và đây là dữ liệu đã thu thập

	Summary	Text
0	Ngay khi bắt tay ai đó có thể chúng ta đã cảm giác đượ...	Tổng thống Mỹ Donald Trump và Tổng thống Pháp Em...
1	Rời khỏi Việt Nam vào 10/2016, ông Vũ Đình Duy tới số...	Vũ Đình Duy hồi tháng 5 xuất hiện trước tòa tại Berlin t...
2	Thủ tướng Iraq xác nhận với BBC rằng Syria không kích...	Ông Maliki xác nhận các vụ không kích của Syria nhằm v...
3	Người Việt Nam biết rất ít về ông Trump nhưng biết kh...	Tiến sỹ Vũ Cao Phan tin rằng nếu được tiến hành một c...
4	Bốn khoa học gia người Mỹ từng đoạt giải Nobel vật lý...	GS Jack Steinberger trao đổi với GS Lê Kim Ngọc, phu n...
5	Chính phủ Singapore cho hay cựu thủ tướng, người đư...	Lý Quang Diệu được cho là người đứng đằng sau "điều...
6	Các thỏa thuận trị giá nhiều tỷ USD giữa Hoa Kỳ và Saud...	Tổng thống Mỹ Donald Trump và phu nhân tới Saudi Ar...
7	Kết thúc phiên kiểm định định kỳ phổ quát về tình hình...	Ông Hà Kim Ngọc, Thứ trưởng Bộ Ngoại giao phát biểu...
8	Liên đoàn Bóng đá Việt Nam xin lỗi đội tuyển Indonesia...	Trận đấu lượt về ở sân Mỹ Đình Đêm 7/12, sau trận cầu...
9	Tòa án tỉnh Đồng Nai vừa tuyên án tù ba nhà hoạt động...	Bà Lê Thị Phương Anh (giữa) cùng chồng đã giúp đỡ nhi...
10	Hơn 1.200 cảnh sát và nhân viên công quyền Pháp vừa...	Chiến dịch giải tỏa trại Calais có thể kéo dài một tuần Tr...
11	Hoa Kỳ điều hai phi cơ ném bom loại B-1B Lancers tha...	Hai phi cơ ném bom loại B-1B Lancers Sau khi bay tập tr...
12	HLV Manchester City Manuel Pellegrini tin rằng việc thủ...	Hai bàn thắng của Luis Suarez ở hiệp một tạo lợi thế ch...
13	Vài ngày sau khi bị câu lưu, bà Cao Vĩnh Thịnh, thành vi...	Bà Cao Vĩnh Thịnh trong sự kiện công chiếu phim Đứng...
14	Joshua Wong tới Berlin, tham gia sự kiện "Bild100-Part...	Joshua Wong trò chuyện cùng Ngoại trưởng Đức Heiko...
15	Chủ tịch Đại học Fulbright ở Việt Nam Bob Kerrey nói ô...	Ông Kerrey nói sẵn sàng rút lui nếu việc tham gia gây bấ...
16	Giữa Bangkok có một cộng đồng nhỏ người Thượng nói...	Khu nhà những người Thượng sống ở Bangkok Cộng đồ...
17	Ngày 19/7 tới đây, U.22 Việt Nam sẽ bước vào vòng loạ...	Thế nhưng, không mấy ai đặt sự quan tâm thực sự vào...
18	Ít nhất đã xác định được 3.218 người thiệt mạng trong...	Rameshwor Dungal, người đứng đầu cơ quan theo dõi...
19	Một quan chức điều tra an toàn của Trung Quốc đang l...	Chưa biết tới khi nào người dân TQ mới biết được lời gi...

Cả nhóm tiến hành tạo cột New Summary bằng cách đọc từng bài báo để đưa ra bản tóm tắt khác với bảng tóm tắt của các trang báo. Do không có nhiều thời gian nên nhóm chỉ có thể làm được hơn 20.000 dữ liệu. Và đây là bộ dữ liệu mới

	Summary	Text	New Summary
0	Ngay khi bắt tay ai đó có thể chúng ta đã cảm giác đượ...	Tổng thống Mỹ Donald Trump và Tổng thống Pháp Em...	vi vậy, trong khi vẫn giữ cái ôm quá thân mật với macro...
1	Rời khỏi Việt Nam vào 10/2016, ông Vũ Đình Duy tới số...	Vũ Đình Duy hồi tháng 5 xuất hiện trước tòa tại Berlin t...	"ví dụ như quyết định của tôi khác, nhưng không phải l...
2	Thủ tướng Iraq xác nhận với BBC rằng Syria không kích...	Ông Maliki xác nhận các vụ không kích của Syria nhằm v...	vị thủ tướng nói iraq không đề nghị syria ném bom như...
3	Người Việt Nam biết rất ít về ông Trump nhưng biết kh...	Tiến sỹ Vũ Cao Phan tin rằng nếu được tiến hành một c...	nước mỹ đã có một tổng thống mới, thứ 45. nhưng nên...
4	Bốn khoa học gia người Mỹ từng đoạt giải Nobel vật lý...	GS Jack Steinberger trao đổi với GS Lê Kim Ngọc, phu n...	gs jack steinberger trao đổi với gs lê kim ngọc, phu nhâ...
5	Chính phủ Singapore cho hay cựu thủ tướng, người đư...	Lý Quang Diệu được cho là người đứng đằng sau "điều...	lý quang diệu được cho là người đứng đằng sau "điều k...
6	Các thỏa thuận trị giá nhiều tỷ USD giữa Hoa Kỳ và Saud...	Tổng thống Mỹ Donald Trump và phu nhân tới Saudi Ar...	trump: vụ sa thài 'gã điên' fbi 'làm giảm áp lực' trump:...
7	Kết thúc phiên kiểm định định kỳ phổ quát về tình hình...	Ông Hà Kim Ngọc, Thứ trưởng Bộ Ngoại giao phát biểu...	Ông Hà Kim Ngọc, Thứ trưởng Bộ Ngoại giao phát biểu...
8	Liên đoàn Bóng đá Việt Nam xin lỗi đội tuyển Indonesia...	Trận đấu lượt về ở sân Mỹ Đình Đêm 7/12, sau trận cầu...	trận đấu lượt về ở sân mỹ đình đêm 7/12, sau trận cầu...
9	Tòa án tỉnh Đồng Nai vừa tuyên án tù ba nhà hoạt động...	Bà Lê Thị Phương Anh (giữa) cùng chồng đã giúp đỡ nhi...	bà lê thị phương anh (giữa) cùng chồng đã giúp đỡ nhiề...
10	Hơn 1.200 cảnh sát và nhân viên công quyền Pháp vừa...	Chiến dịch giải tỏa trại Calais có thể kéo dài một tuần Tr...	sau đó hàng trăm người đổ ra từ trại và đứng dọc đườn...
11	Hoa Kỳ điều hai phi cơ ném bom loại B-1B Lancers tha...	Hai phi cơ ném bom loại B-1B Lancers Sau khi bay tập tr...	hai phi cơ ném bom loại b-1b lancers sau khi bay tập tr...
12	HLV Manchester City Manuel Pellegrini tin rằng việc thủ...	Hai bàn thắng của Luis Suarez ở hiệp một tạo lợi thế ch...	hai bàn thắng của luis suarez ở hiệp một tạo lợi thế cho...
13	Vài ngày sau khi bị câu lưu, bà Cao Vĩnh Thịnh, thành vi...	Bà Cao Vĩnh Thịnh trong sự kiện công chiếu phim Đứng...	hôm 16/3, nhóm green trees vừa tổ chức buổi công chi...
14	Joshua Wong tới Berlin, tham gia sự kiện "Bild100-Part...	Joshua Wong trò chuyện cùng Ngoại trưởng Đức Heiko...	nếu như sáng hôm nay vẫn còn chưa có thật nhiều tờ b...
15	Chủ tịch Đại học Fulbright ở Việt Nam Bob Kerrey nói ô...	Ông Kerrey nói sẵn sàng rút lui nếu việc tham gia gây bấ...	trong điện thư trả lời nguyên hùng của bbc tiếng việt h...
16	Giữa Bangkok có một cộng đồng nhỏ người Thượng nói...	Khu nhà những người Thượng sống ở Bangkok Cộng đồ...	những người thượng này nói rằng họ đã phải đổi mặt v...
17	Ngày 19/7 tới đây, U.22 Việt Nam sẽ bước vào vòng loạ...	Thế nhưng, không mấy ai đặt sự quan tâm thực sự vào...	tất cả đều giành cho sea games gần cả tuần nay, đội tuy...
18	Ít nhất đã xác định được 3.218 người thiệt mạng trong...	Rameshwor Dungal, người đứng đầu cơ quan theo dõi...	hàng chục ngàn người đã phải chịu cảnh màn trời chiếu...
19	Một quan chức điều tra an toàn của Trung Quốc đang l...	Chưa biết tới khi nào người dân TQ mới biết được lời gi...	40 người đã thiệt mạng trong vụ tai nạn ôn châu, sự cố...



Và để có cái nhìn tổng quan hơn về bộ dữ liệu mới của nhóm đây là minh họa sự khác biệt giữa tóm tắt cũ và tóm tắt mới. Cũng có thể hiểu một cách khoa học thì tóm tắt cũ là Abstractive Summarization và tóm tắt mới là Extractive Summarization đã được nói trên.

	Summary	New Summary
0	Ngay khi bắt tay ai đó có thể chúng ta đã cảm giác được là có cái gì đó sai sai, nhưng lại không thể biết đích xác là tại sao. Giữ tay quá lâu, nắm quá chặt, hay kéo tay khiến hai người gần sát nhau quá... những thứ đó đều có thể ảnh hưởng tới phần còn lại của cuộc gặp gỡ.	vì vậy, trong khi vẫn giữ cái ôm quá thân mật với macron thì trump đồng thời có thể hôn brigitte như thông lệ. nhưng cả hai đều không buông; cái bắt tay kỳ quái không theo quy tắc nào giữa macron và trump có lẽ là một bài học về những cử chỉ không nên làm khi chào hỏi người khác. chúng ta thường hay nắm tay của chính mình, nhưng sau một cái bắt tay, chúng ta thậm chí nắm tay liên tục hơn - nhất là với bàn tay phải, bàn tay mà chúng ta sử dụng để bắt tay khi chào hỏi. đừng dùng đáng lâu quá một cái bắt tay không chỉ truyền tải mức độ hồ hởi, phấn khởi và khả năng được tuyển dụng của bạn đến người phỏng vấn, mà còn giúp bạn tìm hiểu về người mà bạn đang gặp mặt.
1	Rời khỏi Việt Nam vào 10/2016, ông Vũ Đình Duy tới sống tại thủ đô của Ba Lan, và chủ yếu dành thời gian đi đi lại lại giữa Warsaw và Berlin.	"ví dụ như quyết định của tôi khác, nhưng không phải lúc nào cũng có cơ hội trình bày," ông nói. vợ ông trịnh xuân thanh tại tòa ở berlin khai vũ đình duy có quan hệ họ hàng thân thiết với chồng mình một số nguồn tin từ ba lan nói trước đó, vào tháng 5/2017, ông duy đã được giới chức ba lan cấp giấy cư trú dài hạn theo dạng di dân lao động. ông nói trong chuyến đi của ông tới prague hồi 7/2017, bạn gái ông có đi cùng, và người này cũng xuất hiện trong nhiều cuộc gặp gỡ của ông với những người khác tại đức và cộng hòa czech. hôm 31/5/2018, bộ công an việt nam khởi tố bổ sung tội 'nhận hối lộ' đối với ông vũ đình duy và phát thêm lệnh truy nã. chi tiết này cũng được ông duy xác nhận, và đó là lần cuối cùng ông duy gặp ông trịnh xuân thanh.
2	Thủ tướng Iraq xác nhận với BBC rằng Syria không kích quân nổi dậy Isis trên lãnh thổ Iraq.	vì thủ tướng nói iraq không đề nghị syria ném bom nhưng "hoan nghênh" bất kỳ cuộc tấn công nào nhắm vào nhóm hồi giáo isis. nhóm này và các đồng minh hồi giáo sunni đã chiếm được phần lớn lãnh thổ iraq trong tháng này bao gồm cả thành phố lớn thứ nhì mosul. họ cũng được sự ủng hộ của iran, nước có quan hệ gần gũi với phe đa số shia của iraq. ông maliki sẽ cố gắng lập ra chính phủ mới nhằm giữ hòa hợp dân tộc khi quốc hội nhóm họp trong tuần tới. vùng màu đỏ do isis kiểm soát, màu xanh đậm đang có tranh chấp, xanh nhạt do chính quyền kurd kiểm soát và vùng vàng là những nơi có hoạt động của isis
3	Người Việt Nam biết rất ít về ông Trump nhưng biết khá nhiều về bà Clinton. Họ biết vì bà đến đất nước này nhiều lần trên những cương vị khác nhau.	nước mỹ đã có một tổng thống mới, thứ 45. nhưng nền chính trị nước này sẽ còn lâu dài lật đi lật lại nhiều câu hỏi, và lại có nhiều người sẽ lấy được học vị tiến sĩ xung quanh cuộc bầu cử "khốc liệt và gây chia rẽ nhất" đất nước. ông trump đã không ngần ngại tuyên bố: "đây không phải là một cuộc vận động bầu cử mà là một phong trào". mong bà hãy giữ gìn sức khỏe và xin được gửi lại bà câu nói của tổng thống obama đêm trước cuộc bỏ phiếu: "bất luận điều gì xảy ra thì mặt trời vẫn mọc vào buổi sớm" (no matter what happens, the sun wil rise in the morning). các động thái đầu tiên cho thấy những dấu hiệu tích cực. ông sẽ có điều kiện để xem ý kiến đó của mình có đúng không.
4	Bốn khoa học gia người Mỹ từng đoạt giải Nobel vật lý đã cùng đến thành phố Quy Nhơn, tỉnh Bình Định chiều ngày 11/8 để tham dự một hội nghị khoa học quốc tế có tiêu đề: 'Các cửa sổ nhìn ra vũ trụ'.	gs jack steinberger trao đổi với gs lê kim ngọc, phu nhân gs trần thanh văn trước đó, một nhà vật lý đoạt giải nobel khác người đức cũng đã có mặt ở thành phố biển này để tham gia và hội nghị khoa học nói trên diễn ra từ ngày 12/8 đến ngày 17/8. các giáo sư người mỹ bao gồm: sheldon lee glashow đạt giải nobel năm 1979, jack steinberger năm 1988, david j. gross năm 2004 và george smoot năm 2006, còn vị giáo sư người đức là klaus von kltzing đạt giải năm 1985. tất cả các vị này là khách mời danh dự của chương trình 'gặp gỡ việt nam' lần thứ 9 mà hội thảo 'các cửa sổ nhìn ra vũ trụ' là một nội dung chính. ngoài ra còn có giám đốc trung tâm nghiên cứu hạt nhân châu âu (cern) rolf heuer, gs đàm thanh sơn và gs ngô bảo châu. theo tường thuật của báo chí trong nước thì trong phiên khai mạc hội nghị khoa học sáng 12/8, hai giáo sư sheldon lee glashow và klaus von kltzing đã có những bài giảng mở rộng cho công chúng khoa học về vai trò của khoa học cơ bản trong việc tạo ra những bước tiến đột phá trong công nghệ. sự kiện các nhân vật nổi trên có mặt ở việt nam đã được truyền thông trong nước quảng bá rầm rộ.

## 2. Tiền xử lí dữ liệu

- Xóa các kí tự đặc biệt
- Xóa link
- Xóa các khoảng trống
- Chuyển các chữ hoa thành chữ thường
- Xóa các từ stopwords tiếng việt và tiếng anh ( Vì đây là các bài báo đôi khi họ có thể viết một vài từ tiếng anh vào trong đấy)

```
def cleanWord(s):
    miss = ['!', '@', '#', '$', '%', '^', '&', '*', '(', ')', '-', '_', '+', '=', '{', '[', '}', ']', '|', ':', ';', '?', '/', '<', '>', '~', '`', '´',
    for i in miss:
        s = s.replace(i, '')
    return re.sub(r'\s+', ' ', s)

def remove_link(string):
    return re.sub(r'\w+:\/\/{2}[\d\w-]+(\.[\d\w-]+)*(?:\/(?:\^\/[^\s/]*)*)', '', string)

# Xử lí Stopwords_Vietnamese and Stopwords_English

def remove_extra_whitespace(string):
    text = re.sub(r'\s+', ' ', string).strip()
    return text

def load_stopword(path):
    with open(path, 'r', encoding="utf-8") as f:
        stopwords = f.readlines()
        stop_set = set(m.strip() for m in stopwords)
        return stop_set

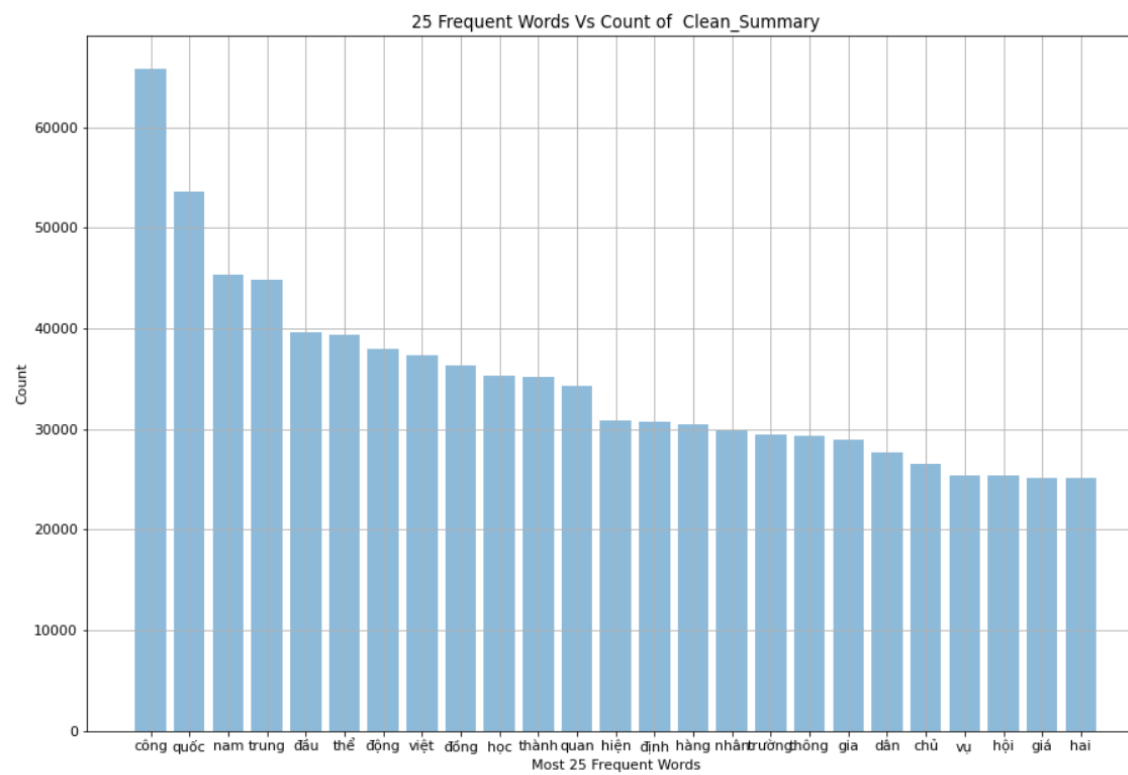
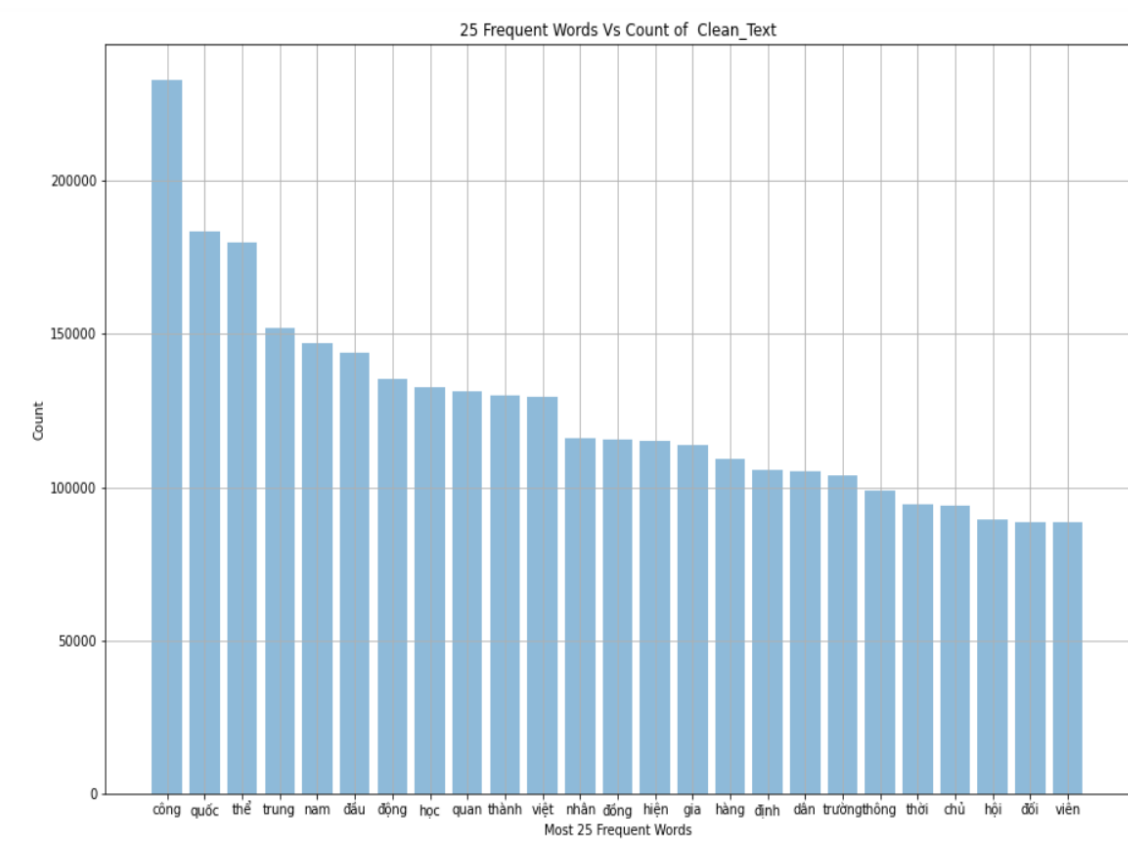
stopword = load_stopword(r"D:\Năm 3 - Đại học\Xử lí ngôn ngữ tự nhiên\Đồ án NLP\Data\vietnamese-stopwords.txt")

def remove_stopwords(line):
    words = []
    for word in line.strip().split():
        if word not in stopword:
            words.append(word)
    return ' '.join(words)

def remove_stopword_english(data):
    text = data.split()
    stops = set(stopwords.words("english"))
    text = [w for w in text if w not in stops]
    text = " ".join(text)
    return text
```

## 3. Trục quan hóa dữ liệu

Đây là đồ thị minh họa 25 từ có tần suất xuất hiện nhiều nhất trong 2 cột dữ liệu đã qua xử lí là Text và Summary



## 4. Huấn luyện mô hình

### 4.1 Build Model

Model: "model"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 500)]	0	[]
input_2 (InputLayer)	[(None, None)]	0	[]
Encoder_Embedding_layer (Embedding)	(None, 500, 100)	7873400	['input_1[0][0]']
Decoder_Embedding_layer (Embedding)	(None, None, 100)	3528000	['input_2[0][0]']
Encoder_LSTM1 (LSTM)	[(None, 500, 300), (None, 300), (None, 300)]	481200	['Encoder_Embedding_layer[0][0]']
Decoder_LSTM1 (LSTM)	[(None, None, 300), (None, 300), (None, 300)]	481200	['Decoder_Embedding_layer[0][0]', 'Encoder_LSTM1[0][1]', 'Encoder_LSTM1[0][2]']
time_distributed (TimeDistributed)	(None, None, 35280)	10619280	['Decoder_LSTM1[0][0]']
Total params: 22,983,080			
Trainable params: 22,983,080			
Non-trainable params: 0			

### 4.2 Tiến hành huấn luyện mô hình

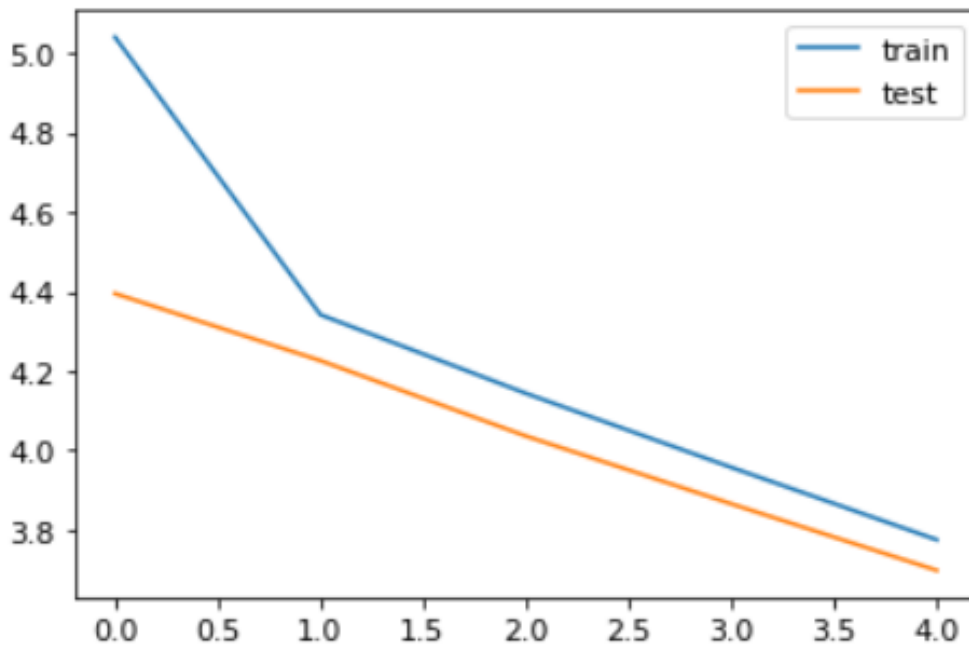
- Sử dụng mô hình Seq2Seq để tiến hành train dữ liệu
- Thời gian train: 1 giờ 30 phút / epoch
- Tiến hành train 5 epoch

```
Epoch 1/5
266/266 [=====] - 3982s 15s/step - loss: 5.0402 - acc: 0.3970 - val_loss: 4.3948 - val_acc: 0.4059
Epoch 2/5
266/266 [=====] - 4118s 15s/step - loss: 4.3416 - acc: 0.4096 - val_loss: 4.2261 - val_acc: 0.4198
Epoch 3/5
266/266 [=====] - 4280s 16s/step - loss: 4.1437 - acc: 0.4208 - val_loss: 4.0363 - val_acc: 0.4348
Epoch 4/5
266/266 [=====] - 4085s 15s/step - loss: 3.9575 - acc: 0.4456 - val_loss: 3.8656 - val_acc: 0.4631
Epoch 5/5
266/266 [=====] - 4099s 15s/step - loss: 3.7753 - acc: 0.4718 - val_loss: 3.6983 - val_acc: 0.4855
```

## 5. Kết quả

```
94/94 [=====] - 216s 2s/step - loss: 3.6983 - acc: 0.4855  
Loss of the model is - 3.6983304023742676  
94/94 [=====] - 228s 2s/step - loss: 3.6983 - acc: 0.4855  
Accuracy of the model is - 48.55259656906128 %
```

Đồ thị biểu diễn độ lỗi của mô hình



### Nhận xét:

- Độ lỗi của mô hình vẫn còn cao
- Độ chính xác vẫn còn thấp
- Cần chuẩn bị dữ liệu thêm nữa và tăng số lần chạy epoch lên thêm nữa
- Và tiến hành thử nhiều model khác nhau để so sánh với nhau

# BÁO CÁO BÀI TẬP LỚN THƯỜNG KỲ 2

## 1. Mô tả

Cả nhóm tóm tắt bằng tay thêm cho bộ dữ liệu và bộ dữ liệu mới được 60.000 dòng và tiến hành train thử trên mô hình cũ nhưng không được do độ dài của hai cột **Text** và **New Summary** quá lớn với **max\_text\_len = 2158** và **max\_newsummary\_len = 258** không thể train được mô hình vì bộ nhớ không đủ trên cả kaggle và google colab và quá thời gian chạy 12 tiếng trên cả hai.

Nên cả nhóm quyết định không làm theo hướng **Extractive Summarization** mà tiến hành làm theo hướng **Abstractive Summarization** với bộ dữ liệu ban đầu là hơn 100.000 dòng dữ liệu. Với hai cột **Text** và **New Summary** với **max\_text\_len = 2158** và **max\_summary\_len = 70** tiến hành lọc data với với **max\_text\_len = 300** và **max\_summary\_len = 60** thì bộ dữ liệu chỉ còn 14.664 dòng

Có thể tăng **max\_text\_len = 500** để lọc data để có nhiều dòng dữ liệu hơn nhưng nhóm sẽ thử trước với bộ dữ liệu với 14.664 dòng

Nếu có thời gian thêm nữa thì nhóm sẽ tiến hành thu thập dữ liệu thêm cho bộ dữ liệu để đem đi huấn luyện mô hình tốt hơn

## 2. Tiền xử lí dữ liệu

```
def cleanWord(s):
    miss = ['!', '@', '#', '$', '%', '^', '&', '*', '(', ')', '-', '_', '+', '=', '{', '[', '}', ']', '|', ':', ';', '?', '/', '<', '>', '~', "'", '<img alt="Python code for text preprocessing functions" data-bbox="121 573 875 875"/>
    for i in miss:
        s = s.replace(i, '')
    return re.sub(r'\', ', s)

def remove_link(string):
    return re.sub(r'\w+:\/\/[2][\d\w-]+\.[\d\w-]+(?:\/(?:\/[\^\/s/]*)*)', '', string)

def remove_extra_whitespace(string):
    text = re.sub(r'\s+', ' ', string).strip()
    return text

def lower_word(data):
    return data.lower()

# Remove numbers from text
def rm_number_from_text(text):
    text = re.sub('[0-9]+', '', text)
    return ' '.join(text.split()) # to rm `extra` white space

# Remove punctuation from word
def rm_punc_from_word(word):
    clean_alphabet_list = [alphabet for alphabet in word if alphabet not in string.punctuation]
    return ''.join(clean_alphabet_list)

# Remove punctuation from text
def rm_punc_from_text(text):
    clean_word_list = [rm_punc_from_word(word) for word in text]
    return ''.join(clean_word_list)
```



```

# Cleaning text
def clean_text(text):
    text = text.lower()
    text = rm_number_from_text(text)
    text = rm_punc_from_text(text)

    # there are hyphen(-) in many titles, so replacing it with empty str
    # this hyphen(-) is different from normal hyphen(-)
    text = re.sub('-', '', text)
    text = ' '.join(text.split()) # removing `extra` white spaces

    # Removing unnecessary characters from text
    text = re.sub("(\t)", ' ', str(text)).lower()
    text = re.sub("(\r)", ' ', str(text)).lower()
    text = re.sub("(\n)", ' ', str(text)).lower()

    text = re.sub("(_+)", ' ', str(text)).lower()
    text = re.sub("--+", ' ', str(text)).lower()
    text = re.sub("~+", ' ', str(text)).lower()
    text = re.sub("(\\+\\+)", ' ', str(text)).lower()
    text = re.sub("(\\.\\.\\.+)", ' ', str(text)).lower()

    text = re.sub(r"<>()|&@#\\[\\]\\'\";?~*!]", ' ', str(text)).lower()

    text = re.sub("(mailto:)", ' ', str(text)).lower()
    text = re.sub(r"(\x9\d)", ' ', str(text)).lower()
    text = re.sub("([iI][nN][cC]\d+)", 'INC_NUM', str(text)).lower()
    text = re.sub("([cC][mM]\d+)|([cC][hH][gG]\d+)", 'CM_NUM', str(text)).lower()

    text = re.sub("(\\.s+)", ' ', str(text)).lower()
    text = re.sub("(\\-s+)", ' ', str(text)).lower()
    text = re.sub("(\\:s+)", ' ', str(text)).lower()
    text = re.sub("(\\s+.s+)", ' ', str(text)).lower()

    try:
        url = re.search(r'((https?:\\/*)([^\s/]+))\\.([^\s]+)', str(text))
        repl_url = url.group(3)
        text = re.sub(r'((https?:\\/*)([^\s/]+))\\.([^\s]+)', repl_url, str(text))
    except Exception as e:
        pass

    text = re.sub("(s+)", ' ', str(text)).lower()
    text = re.sub("(s+.s+)", ' ', str(text)).lower()

    return text

def solve(string):
    func = [lower_word, remove_link, remove_extra_whitespace, cleanWord, clean_text]
    for i in func:
        string = i(string)
    return string

```

## Tách từ tiếng việt bằng thư viện VnCoreNLP

```
%%capture
# Install the vncorenlp python wrapper
!pip install vncorenlp

# Download VnCoreNLP-1.1.1.jar & its word segmentation component (i.e. RDRSegmenter)
!mkdir -p vncorenlp/models/wordsegmenter
!wget https://raw.githubusercontent.com/vncorenlp/VnCoreNLP/master/VnCoreNLP-1.1.1.jar
!wget https://raw.githubusercontent.com/vncorenlp/VnCoreNLP/master/models/wordsegmenter/vi-vocab
!wget https://raw.githubusercontent.com/vncorenlp/VnCoreNLP/master/models/wordsegmenter/wordsegmenter.rdr
!mv VnCoreNLP-1.1.1.jar vncorenlp/
!mv vi-vocab vncorenlp/models/wordsegmenter/
!mv wordsegmenter.rdr vncorenlp/models/wordsegmenter/
```

Và sau khi tách từ xong

	Text	Summary
0	ông maliki xác_nhận các vụ không_kích của syri...	thủ_tướng iraq xác_nhận với bbc rằng syria khô...
1	lý quang diệu được cho là người đứng đầu sau...	chính_phủ singapore cho_hay cựu thủ_tướng ngườ...
2	ông hà kim ngọc thứ_trưởng bộ ngoại_giao phát_...	kết_thúc phiên kiểm_định định_kỳ phổ_quát về t...
3	hai bàn thắng của luis suarez hiệp một tạo lợi...	hlv manchester city manuel pellegrini tin rằng...
4	thủ_hiện tiểu_bang nam úc ông jay weatherill l...	một chính_khách gốc việt từng là dân tỵ nạn ...

### 3. Training build\_vocab with 2 method of Skip-gram and CBOW

```
from gensim.models import KeyedVectors
model_ug_cbow = KeyedVectors.load('w2v_model_ug_cbow.word2vec')
model_ug_sg = KeyedVectors.load('w2v_model_ug_sg.word2vec')
```

Appending cbow and sg for better result

```
embeddings_index = {}
for w in model_ug_cbow.wv.index_to_key:
    embeddings_index[w] = np.append(model_ug_cbow.wv[w], model_ug_sg.wv[w])
print('Found %s word vectors.' % len(embeddings_index))
```

Found 31125 word vectors.

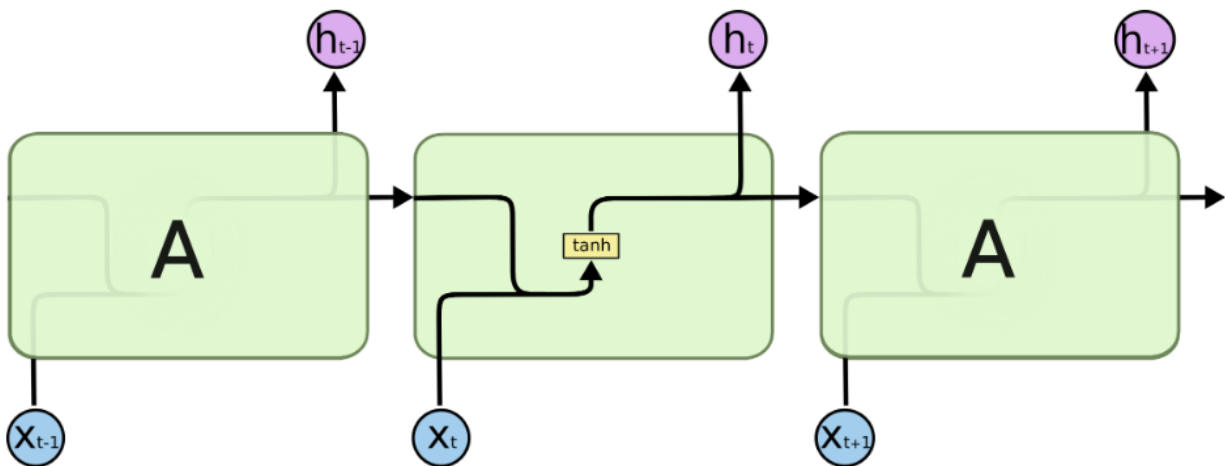
## 4. Build Model

### Mạng LSTM

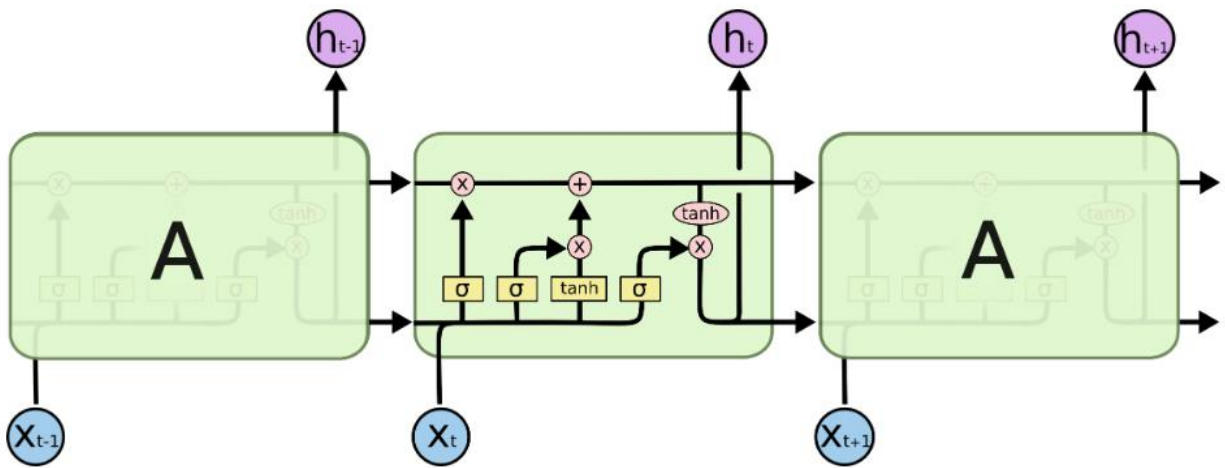
Mạng bộ nhớ dài-ngắn (Long Short-Term Memory networks), thường được gọi là LSTM - là một dạng đặc biệt của RNN, nó có khả năng học được các phụ thuộc xa. LSTM được giới thiệu bởi Hochreiter & Schmidhuber (1997), và sau đó đã được cải tiến và phổ biến bởi rất nhiều người trong ngành. Chúng hoạt động cực kì hiệu quả trên nhiều bài toán khác nhau nên dần đã trở nên phổ biến như hiện nay.

LSTM được thiết kế để tránh được vấn đề phụ thuộc xa (long-term dependency). Việc nhớ thông tin trong suốt thời gian dài là đặc tính mặc định của chúng, chứ ta không cần phải huấn luyện nó để có thể nhớ được. Tức là ngay nội tại của nó đã có thể ghi nhớ được mà không cần bất kì can thiệp nào.

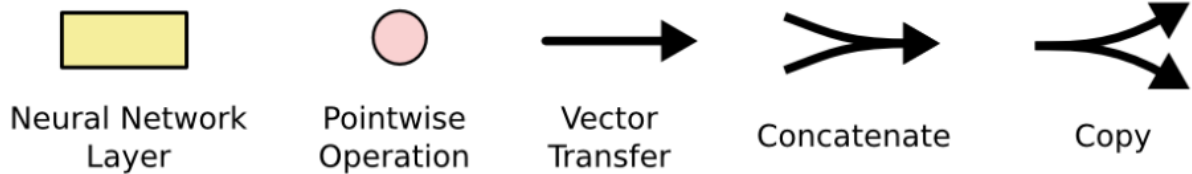
Mọi mạng hồi quy đều có dạng là một chuỗi các mô-đun lặp đi lặp lại của mạng nơ-ron. Với mạng RNN chuẩn, các mô-dun này có cấu trúc rất đơn giản, thường là một tầng tanhtanh.



LSTM cũng có kiến trúc dạng chuỗi như vậy, nhưng các mô-đun trong nó có cấu trúc khác với mạng RNN chuẩn. Thay vì chỉ có một tầng mạng nơ-ron, chúng có tới 4 tầng tương tác với nhau một cách rất đặc biệt.



Các kí hiệu

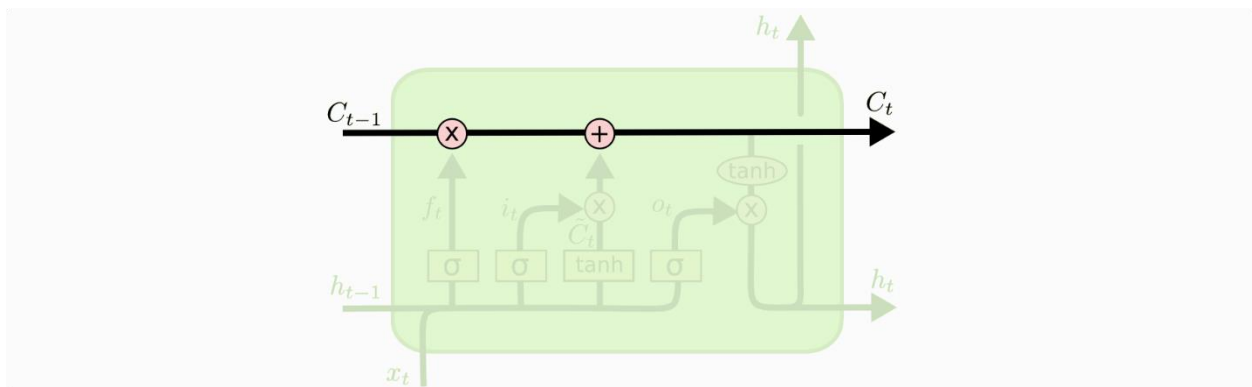


Ở sơ đồ trên, mỗi một đường mang một véc-tơ từ đầu ra của một nút tới đầu vào của một nút khác. Các hình trong màu hồng biểu diễn các phép toán như phép cộng véc-tơ chẳng hạn, còn các ô màu vàng được sử dụng để học trong các từng mạng nơ-ron. Các đường hợp nhau kí hiệu việc kết hợp, còn các đường rẽ nhánh ám chỉ nội dung của nó được sao chép và chuyển tới các nơi khác nhau.

### Ý tưởng cốt lõi của LSTM

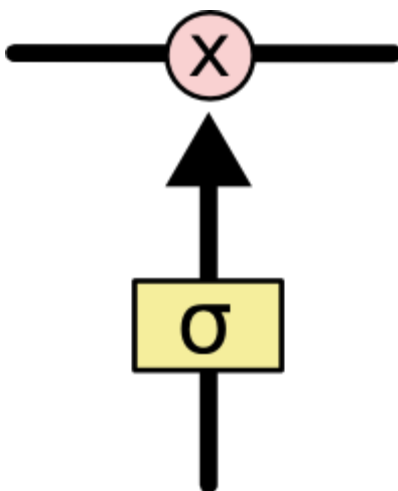
Chìa khóa của LSTM là trạng thái tế bào (cell state) - chính đường chạy thông ngang phía trên của sơ đồ hình vẽ.

Trạng thái tế bào là một dạng giống như băng truyền. Nó chạy xuyên suốt tất cả các mắt xích (các nút mạng) và chỉ tương tác tuyến tính đôi chút. Vì vậy mà các thông tin có thể dễ dàng truyền đi thông suốt mà không sợ bị thay đổi.



LSTM có khả năng bỏ đi hoặc thêm vào các thông tin cần thiết cho trạng thái tế bào, chúng được điều chỉnh cẩn thận bởi các nhóm được gọi là cổng (gate).

Các cổng là nơi sàng lọc thông tin đi qua nó, chúng được kết hợp bởi một tầng mạng sigmoid và một phép nhân.



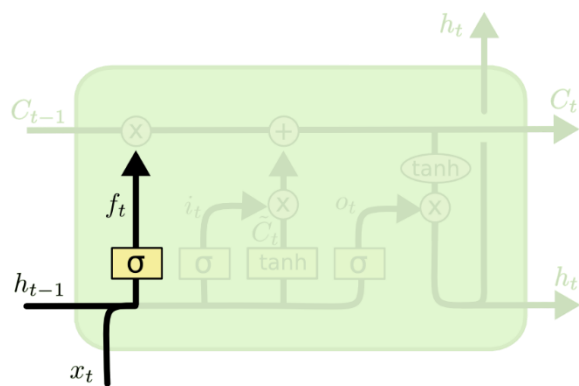
Tầng sigmoid sẽ cho đầu ra là một số trong khoản  $[0, 1]$ , mô tả có bao nhiêu thông tin có thể được thông qua. Khi đầu ra là 00 thì có nghĩa là không cho thông tin nào qua cả, còn khi là 11 thì có nghĩa là cho tất cả các thông tin đi qua nó.

Một LSTM gồm có 3 cổng như vậy để duy trì và điều hành trạng thái của tế bào.

### **Bên trong LSTM**

Bước đầu tiên của LSTM là quyết định xem thông tin nào cần bỏ đi từ trạng thái tế bào. Quyết định này được đưa ra bởi tầng sigmoid - gọi là “tầng cổng quên” (forget gate layer). Nó sẽ lấy đầu vào là  $h_{t-1}$  và  $x_t$  rồi đưa ra kết quả là một số trong khoảng  $[0, 1]$  cho mỗi số trong trạng thái tế bào  $C_{t-1}$ . Đầu ra là 11 thể hiện rằng nó giữ toàn bộ thông tin lại, còn 00 chỉ rằng toàn bộ thông tin sẽ bị bỏ đi.

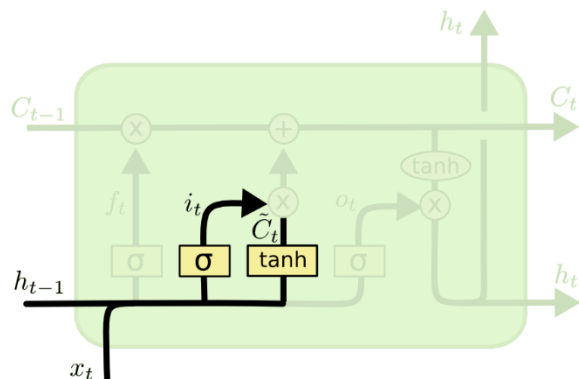
Quay trở lại với ví dụ mô hình ngôn ngữ dự đoán từ tiếp theo dựa trên tất cả các từ trước đó, với những bài toán như vậy, thì trạng thái tế bào có thể sẽ mang thông tin về giới tính của một nhân vật nào đó giúp ta sử dụng được đại từ nhân xưng chuẩn xác. Tuy nhiên, khi đề cập tới một người khác thì ta sẽ không muốn nhớ tới giới tính của nhân vật nữa, vì nó không còn tác dụng gì với chủ thể mới này.



$$f_t = \sigma (W_f \cdot [h_{t-1}, x_t] + b_f)$$

Bước tiếp theo là quyết định xem thông tin mới nào ta sẽ lưu vào trạng thái tế bào. Việc này gồm 2 phần. Đầu tiên là sử dụng một tầng sigmoid được gọi là “tầng cổng vào” (input gate layer) để quyết định giá trị nào ta sẽ cập nhập. Tiếp theo là một tầng *tanh* tạo ra một véc-tơ cho giá trị mới  $\tilde{C}_t$  nhằm thêm vào cho trạng thái. Trong bước tiếp theo, ta sẽ kết hợp 2 giá trị đó lại để tạo ra một cập nhập cho trạng thái.

Chẳng hạn với ví dụ mô hình ngôn ngữ của ta, ta sẽ muốn thêm giới tính của nhân vật mới này vào trạng thái tế bào và thay thế giới tính của nhân vật trước đó.



$$i_t = \sigma (W_i \cdot [h_{t-1}, x_t] + b_i)$$

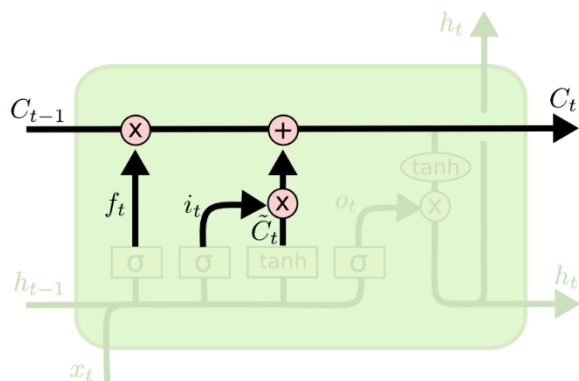
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

Giờ là lúc cập nhập trạng thái tế bào cũ  $C_{t-1}$  thành trạng thái mới  $C_t$ . Ở các bước trước đó đã quyết định những việc cần làm, nên giờ ta chỉ cần thực hiện là xong.

Ta sẽ nhân trạng thái cũ với  $f_t$  để bỏ đi những thông tin ta quyết định quên lúc trước. Sau đó cộng thêm  $i_t * \tilde{C}_t$ . Trạng thái mới thu được này phụ thuộc vào việc ta quyết định cập nhập mỗi giá trị trạng thái ra sao.

Với bài toán mô hình ngôn ngữ, chính là việc ta bỏ đi thông tin về giới tính của nhân vật cũ, và thêm thông tin về giới tính của nhân vật mới như ta đã quyết định ở các bước trước đó.

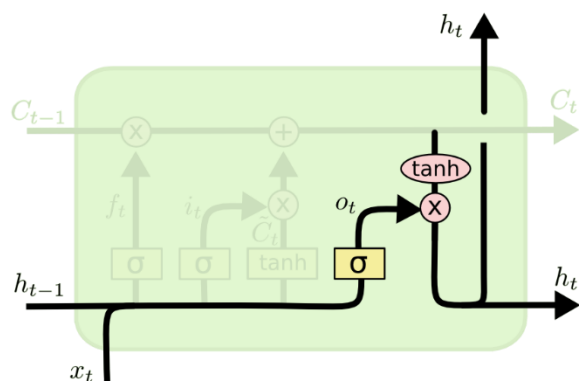




$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Cuối cùng, ta cần quyết định xem ta muốn đầu ra là gì. Giá trị đầu ra sẽ dựa vào trạng thái tế bào, nhưng sẽ được tiếp tục sàng lọc. Đầu tiên, ta chạy một tầng sigmoid để quyết định phần nào của trạng thái tế bào ta muốn xuất ra. Sau đó, ta đưa nó trạng thái tế bào qua một hàm *tanh* để co giá trị nó về khoảng  $[-1, 1]$ , và nhân nó với đầu ra của cổng sigmoid để được giá trị đầu ra ta mong muốn.

Với ví dụ về mô hình ngôn ngữ, chỉ cần xem chủ thể mà ta có thể đưa ra thông tin về một trạng từ đi sau đó. Ví dụ, nếu đầu ra của chủ thể là số ít hoặc số nhiều thì ta có thể biết được dạng của trạng từ đi theo sau nó phải như thế nào.



$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

## 4.1. Text Summary model with just LSTM

LSTM (Long Sort-Term Memory) Bộ nhớ dài ngắn

Cả hai encoder và decoder chỉ có LSTM

Model: "model"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 300)]	0	
embedding (Embedding)	(None, 300, 300)	9069300	input_1[0][0]
input_2 (InputLayer)	[(None, None)]	0	
lstm (LSTM)	[(None, 300, 240), (	519360	embedding[0][0]
embedding_1 (Embedding)	(None, None, 300)	2929200	input_2[0][0]
lstm_1 (LSTM)	[(None, 300, 240), (	461760	lstm[0][0]
lstm_2 (LSTM)	[(None, None, 240),	519360	embedding_1[0][0] lstm_1[0][1] lstm_1[0][2]
time_distributed (TimeDistribut	(None, None, 9764)	2353124	lstm_2[0][0]

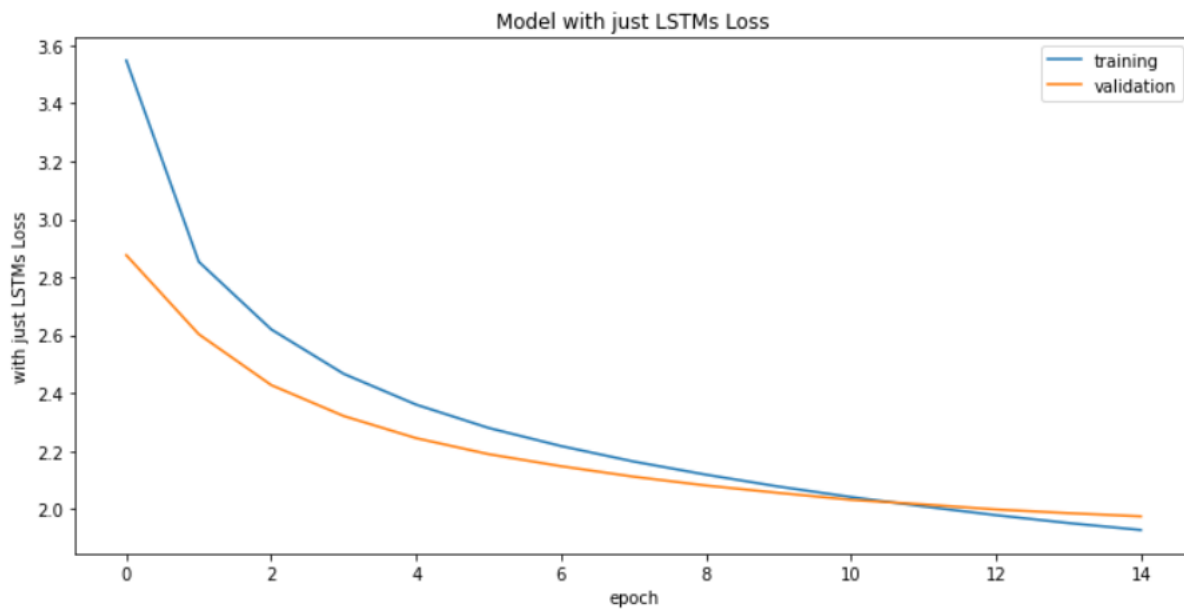
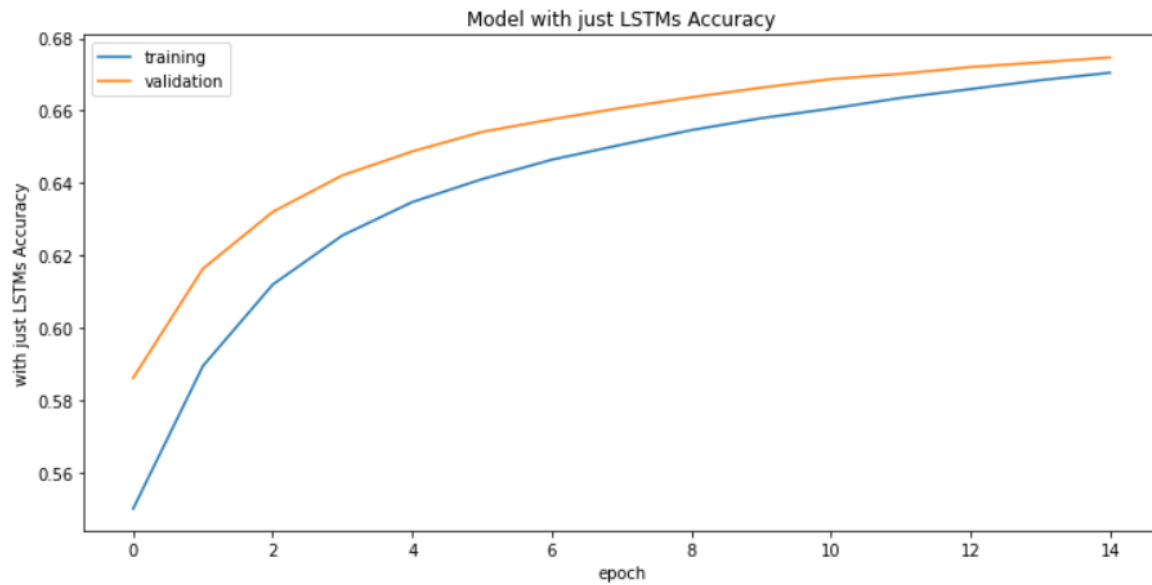
Total params: 15,852,104

Trainable params: 6,782,804

Non-trainable params: 9,069,300

## Kết quả:

- Thời gian train: 15 phút / epoch
- Tiến hành train 15 epoch



## 4.2 Text Summary model with Bidirectional LSTM

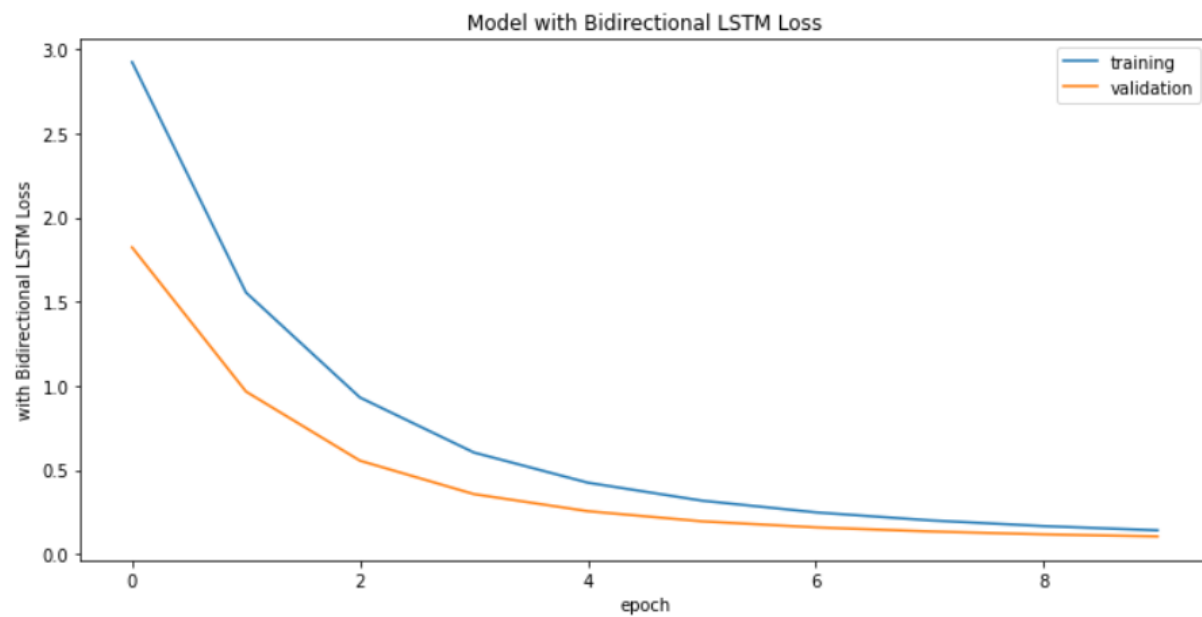
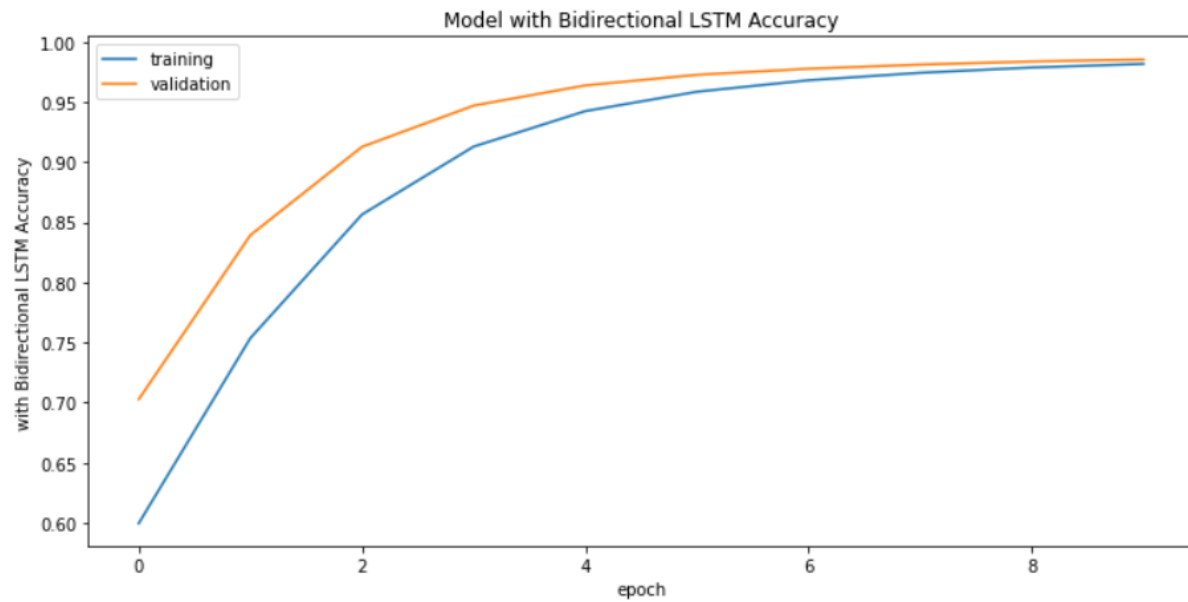
Bidirectional LSTM (BiLSTM) mạng nơ ron bộ nhớ dài ngắn song song

Cả hai encoder và decoder có Bidirectional LSTM

```
Model: "seq2seq_model_with_bidirectional_lstm"
-----
Layer (type)                 Output Shape          Param #   Connected to
-----
input_1 (InputLayer)         [(None, 300)]         0
encoder_embedding (Embedding) (None, 300, 300)      9069300   input_1[0][0]
encoder_bidirectional_lstm_1 (B [(None, 300, 480), ( 1038720   encoder_embedding[0][0]
input_2 (InputLayer)         [(None, None)]        0
encoder_bidirectional_lstm_2 (B [(None, 300, 480), ( 1384320   encoder_bidirectional_lstm_
1[0][0]
decoder_embedding (Embedding) (None, None, 300)     2929200   input_2[0][0]
encoder_bidirectional_lstm_3 (B [(None, 300, 480), ( 1384320   encoder_bidirectional_lstm_
2[0][0]
decoder_bidirectional_lstm_1 (B [(None, None, 480), 1038720   decoder_embedding[0][0]
3[0][1] encoder_bidirectional_lstm_
3[0][2] encoder_bidirectional_lstm_
3[0][3] encoder_bidirectional_lstm_
3[0][4] encoder_bidirectional_lstm_
time_distributed (TimeDistribut (None, None, 9764) 4696484   decoder_bidirectional_lstm_
1[0][0]
Total params: 21,541,064
Trainable params: 9,542,564
Non-trainable params: 11,998,500
```

## Kết quả:

- Thời gian train: 45 phút / epoch
- Tiến hành train 10 epoch



### 4.3 Text Summary model with Hybrid Architecture

encoder có Bidirectional LSTMs trong khi đó decoder chỉ có LSTM

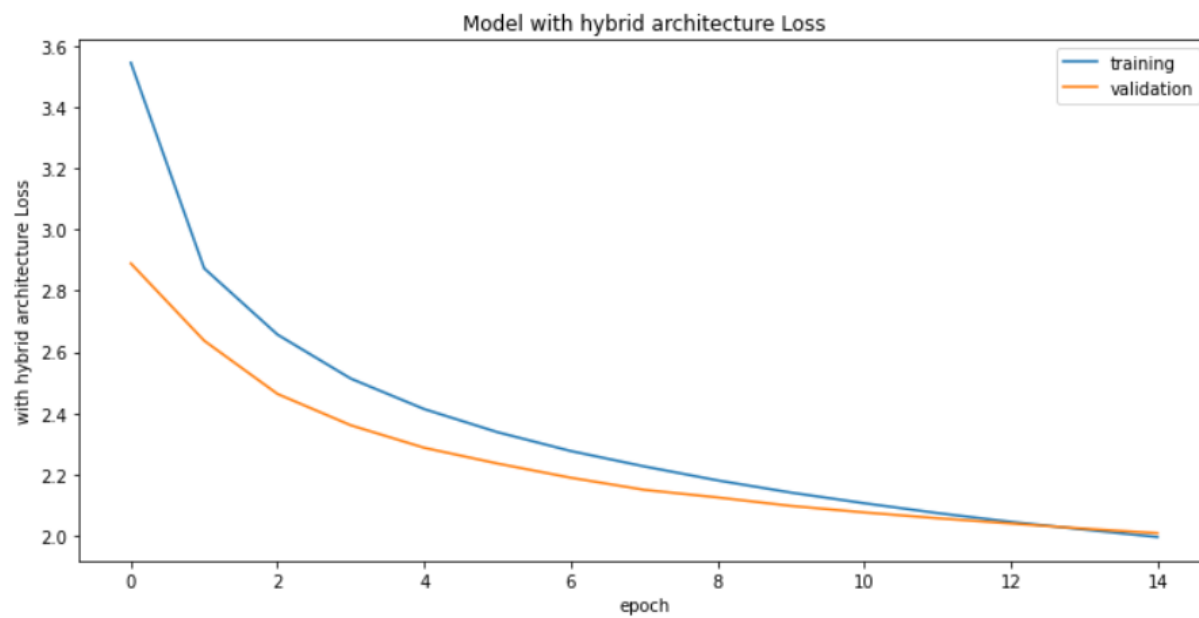
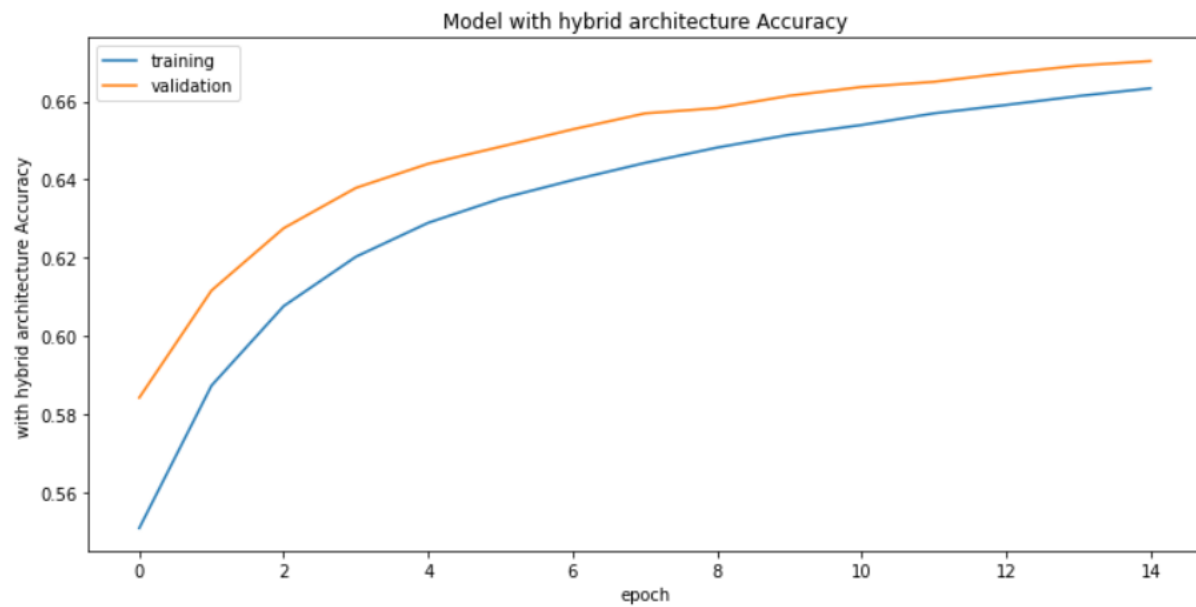
Model: "seq2seq\_model\_with\_bidirectional\_lstm"

```
-----
Layer (type)                 Output Shape          Param #   Connected to
-----
input_1 (InputLayer)         [(None, 300)]         0
encoder_embedding (Embedding) (None, 300, 300)      9069300   input_1[0][0]
encoder_bidirectional_lstm_1 (B [(None, 300, 480), ( 1038720   encoder_embedding[0][0]
input_2 (InputLayer)         [(None, None)]        0
encoder_bidirectional_lstm_2 (B [(None, 300, 480), ( 1384320   encoder_bidirectional_lstm_
1[0][0]
decoder_embedding (Embedding) (None, None, 300)     2929200   input_2[0][0]
encoder_bidirectional_lstm_3 (B [(None, 300, 480), ( 1384320   encoder_bidirectional_lstm_
2[0][0]
decoder_lstm_1 (LSTM)         [(None, None, 240),  519360     decoder_embedding[0][0]
3[0][1]                               encoder_bidirectional_lstm_
3[0][2]
time_distributed (TimeDistribut (None, None, 9764)   2353124   decoder_lstm_1[0][0]
-----
Total params: 18,678,344
Trainable params: 6,679,844
Non-trainable params: 11,998,500
```



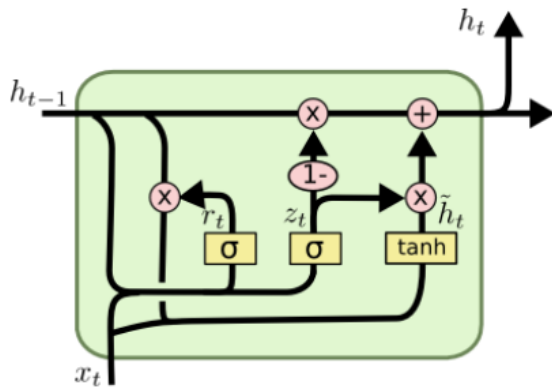
## Kết quả:

- Thời gian train: 38 phút / epoch
- Tiến hành train 15 epoch



#### 4.4 Text Summary GRU have AttentionLayer

Một biến thể khá thú vị khác của LSTM là Gated Recurrent Unit, hay **GRU** được giới thiệu bởi Cho, et al. (2014). Nó kết hợp các cổng loại trừ và đầu vào thành một cổng “cổng cập nhật” (update gate). Nó cũng hợp trạng thái tế bào và trạng thái ẩn với nhau tạo ra một thay đổi khác. Kết quả là mô hình của ta sẽ đơn giản hơn mô hình LSTM chuẩn và ngày càng trở nên phổ biến.



$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

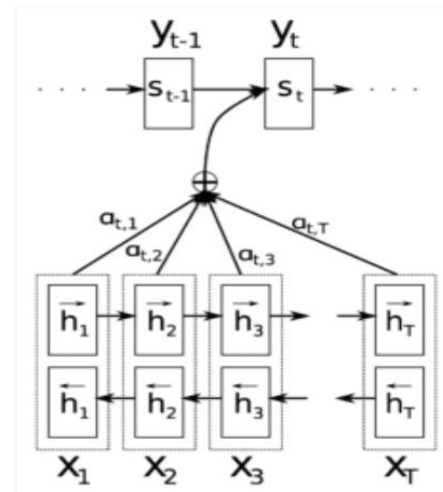
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

#### Kiến trúc của Bahdanau

Các thành phần chính được sử dụng bởi kiến trúc bộ mã hóa-giải mã Bahdanau như sau:

- $s_{t-1}$  là trạng thái bộ giải mã ẩn ở bước thời gian trước đó,  $t - 1$ .
- $c_t$  là vector ngữ cảnh tại bước thời gian,  $t$ . Nó được tạo duy nhất ở mỗi bước bộ giải mã để tạo ra một từ đích,  $y_t$ .
- $h_i$  là một chú thích ghi lại thông tin có trong các từ tạo thành toàn bộ câu đầu vào,  $\{x_1, x_2, \dots, x_T\}$ , với sự tập trung mạnh mẽ xung quanh từ thứ  $i$  trong số  $T$  tổng số từ.
- $\alpha_{t,i}$  là giá trị trọng số được chỉ định cho mỗi chú thích,  $h_i$ , ở bước thời gian hiện tại,  $t$ .
- $e_{t,i}$  là điểm chú ý do mô hình căn chỉnh tạo ra,  $a(\cdot)$ , điểm đó tốt như thế nào so với  $s_{t-1}$  và  $h_i$  trên đầu.

Các thành phần này được sử dụng ở các giai đoạn khác nhau của kiến trúc Bahdanau, sử dụng RNN hai chiều làm bộ mã hóa và bộ giải mã RNN, với cơ chế chú ý ở giữa:



Vai trò của decoder (bộ giải mã) là tạo ra các từ đích bằng cách tập trung vào thông tin phù hợp nhất có trong câu nguồn. Với mục đích này, nó sử dụng cơ chế chú ý.

Bộ giải mã lấy từng chú thích và đưa nó vào mô hình căn chỉnh,  $a(\cdot)$ , cùng với trạng thái bộ giả mã ẩn trước đó,  $S_{t-1}$ . Điều này tạo ra một điểm số chú ý:

$$e_{t,i} = a(s_{t-1}, h_i)$$

Chức năng được thực hiện bởi mô hình liên kết, ở đây, kết hợp  $S_{t-1}$  và  $h_i$  bằng một phép toán cộng. Vì lý do này, cơ chế chú ý được thực hiện bởi Bahdanau et al. được gọi là sự chú ý phụ gia.

Điều này có thể được thực hiện theo hai cách, hoặc (1) bằng cách áp dụng ma trận trọng số,  $W$ , trên các vector được nối,  $S_{t-1}$  và  $h_i$ , hoặc (2) bằng cách áp dụng các ma trận trọng số,  $W_1$  và  $W_2$ , đến  $S_{t-1}$  và  $h_i$  riêng biệt:

1. 
$$a(s_{t-1}, h_i) = v^T \tanh(W [h_i ; s_{t-1}])$$

2. 
$$a(s_{t-1}, h_i) = v^T \tanh(W_1 h_i + W_2 s_{t-1})$$

Đây,  $v$ , là một vector trọng lượng.

Mô hình liên kết được tham số hóa như một mạng nơ-ron truyền thẳng và được huấn luyện chung với các thành phần hệ thống còn lại.

Sau đó, một hàm softmax được áp dụng cho mỗi điểm chú ý để thu được giá trị trọng số tương ứng:

$$\alpha_{t,i} = \text{softmax}(e_{t,i})$$

Ứng dụng của hàm softmax về cơ bản chuẩn hóa các giá trị chú thích thành một phạm vi từ 0 đến 1 và do đó, trọng số kết quả có thể được coi là giá trị xác suất. Mỗi giá trị xác suất (hoặc trọng số) phản ánh mức độ quan trọng  $h_i$  và  $S_{t-1}$  đang tạo trạng thái tiếp theo,  $S_t$  và đầu ra tiếp theo,  $y_t$ .

Cuối cùng, điều này được theo sau bởi việc tính toán vector ngữ cảnh dưới dạng tổng trọng số của các chú thích:

$$c_t = \sum_{i=1}^T \alpha_{t,i} h_i$$

**Text Summary GRU have AttentionLayer** : Cả hai encoder và decoder chỉ có LSTM và Bahdanau Attention được thêm vào trong decoder

```
class BahdanauAttention(tf.keras.layers.Layer):
    def __init__(self, units):
        super(BahdanauAttention, self).__init__()
        self.W1 = tf.keras.layers.Dense(units)
        self.W2 = tf.keras.layers.Dense(units)
        self.V = tf.keras.layers.Dense(1)

    def call(self, query, values):

        query_with_time_axis = tf.expand_dims(query, 1)

        score = self.V(tf.nn.tanh(
            self.W1(query_with_time_axis) + self.W2(values)))

        attention_weights = tf.nn.softmax(score, axis=1)

        context_vector = attention_weights * values
        context_vector = tf.reduce_sum(context_vector, axis=1)

        return context_vector, attention_weights
```

## Kết quả:

Epoch 1 Loss 3.0315  
Time taken for 1 epoch 126.69849348068237 sec

Epoch 2 Loss 2.8673  
Time taken for 1 epoch 56.49686408042908 sec

Epoch 3 Loss 2.7239  
Time taken for 1 epoch 56.35508942604065 sec

Epoch 4 Loss 2.5748  
Time taken for 1 epoch 56.56177496910095 sec

Epoch 5 Loss 2.4367  
Time taken for 1 epoch 56.099013328552246 sec

Epoch 6 Loss 2.3213  
Time taken for 1 epoch 56.57333993911743 sec

Epoch 7 Loss 2.2280  
Time taken for 1 epoch 56.108662366867065 sec

Epoch 8 Loss 2.1463  
Time taken for 1 epoch 56.81694769859314 sec

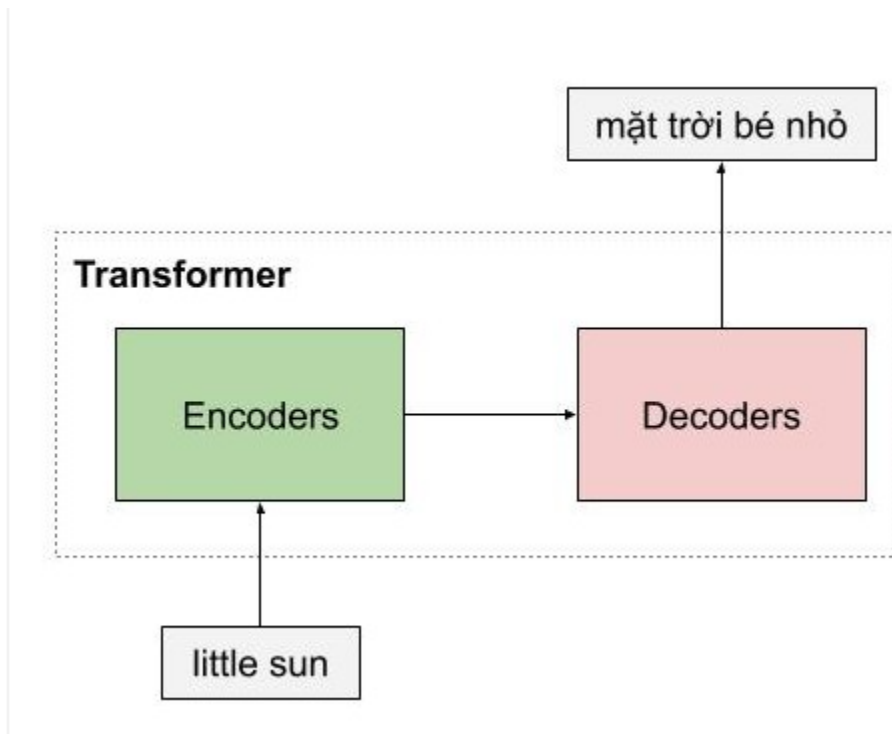
Epoch 9 Loss 2.0731  
Time taken for 1 epoch 56.21662187576294 sec

Epoch 10 Loss 2.0032  
Time taken for 1 epoch 56.66279196739197 sec

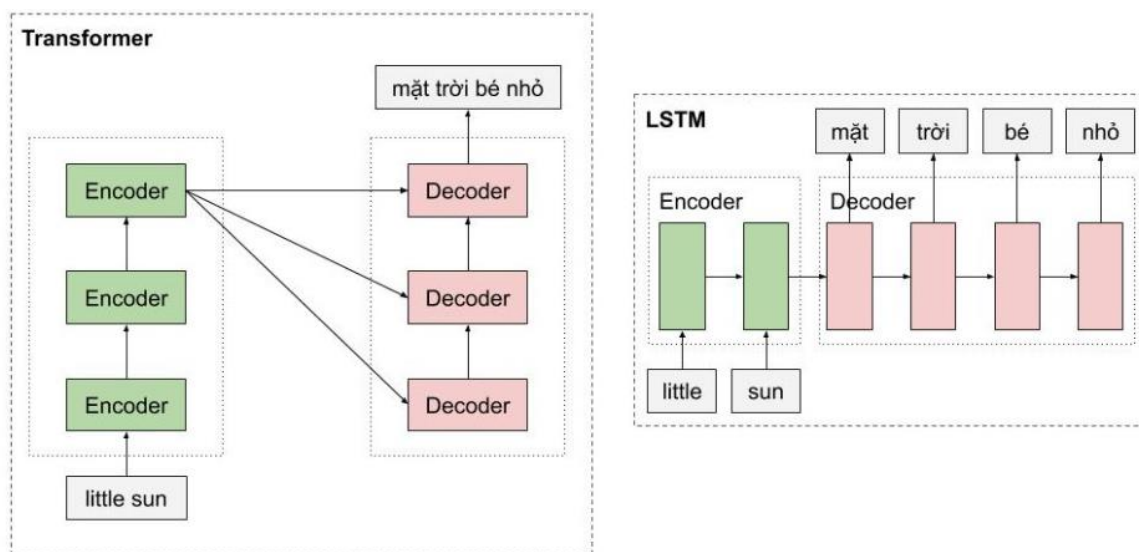
## Transformers – Giới thiệu vài nét cơ bản Transformers

Giống như những mô hình khác, 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.

Trong ví dụ ở dưới, encoder của mô hình transformer nhận một câu tiếng anh, và encode thành một vector biểu diễn ngữ nghĩa của câu *little sun*, sau đó mô hình decoder nhận vector biểu diễn này, và dịch nó thành câu tiếng việt *mặt trời bé nhỏ*

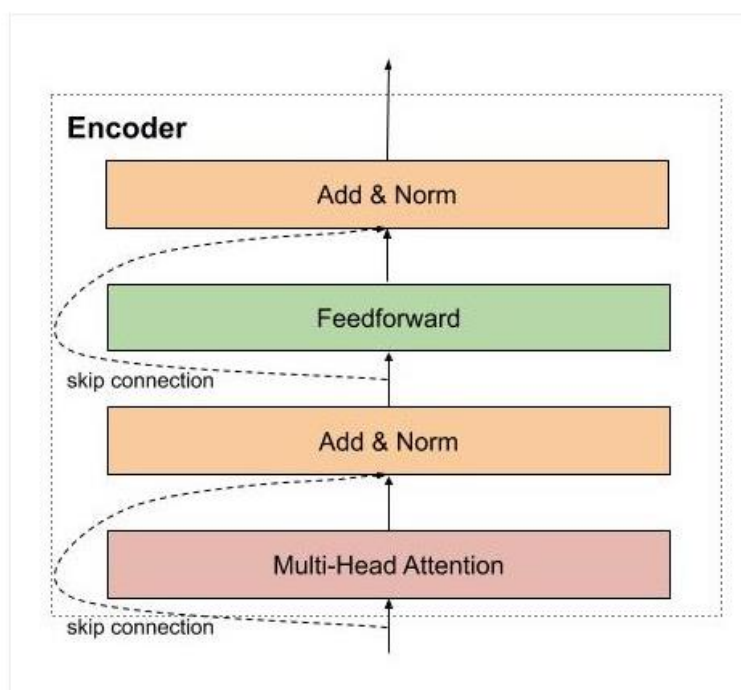


Một trong những ưu điểm của transformer là mô hình này có khả năng xử lý song song cho các từ. Encoders của mô hình transformer là một dạng feedforward neural nets, bao gồm nhiều encoder layer khác, mỗi encoder layer này xử lý đồng thời các từ. Trong khi đó, với mô hình LSTM, thì các từ phải được xử lý tuần tự. Ngoài ra, mô hình Transformer còn xử lý câu đầu vào theo 2 hướng mà không cần phải stack thêm một hình LSTM nữa như trong kiến trúc Bidirectional LSTM.



+ **Encoder:** Encoder của mô hình transformer có thể bao gồm nhiều encoder layer tương tự nhau. Mỗi encoder layer của transformer lại bao gồm 2 thành phần chính là multi head attention và feedforward network, ngoài ra còn có cả skip connection và normalization layer.

Trong 2 thành phần chính này, các bạn sẽ hứng thú nhiều hơn về multi-head attention vì đó là một layer mới được giới thiệu trong bài báo này, và chính nó tạo nên sự khác biệt giữa mô hình LSTM và mô hình Transformer mà chúng ta đang tìm hiểu.





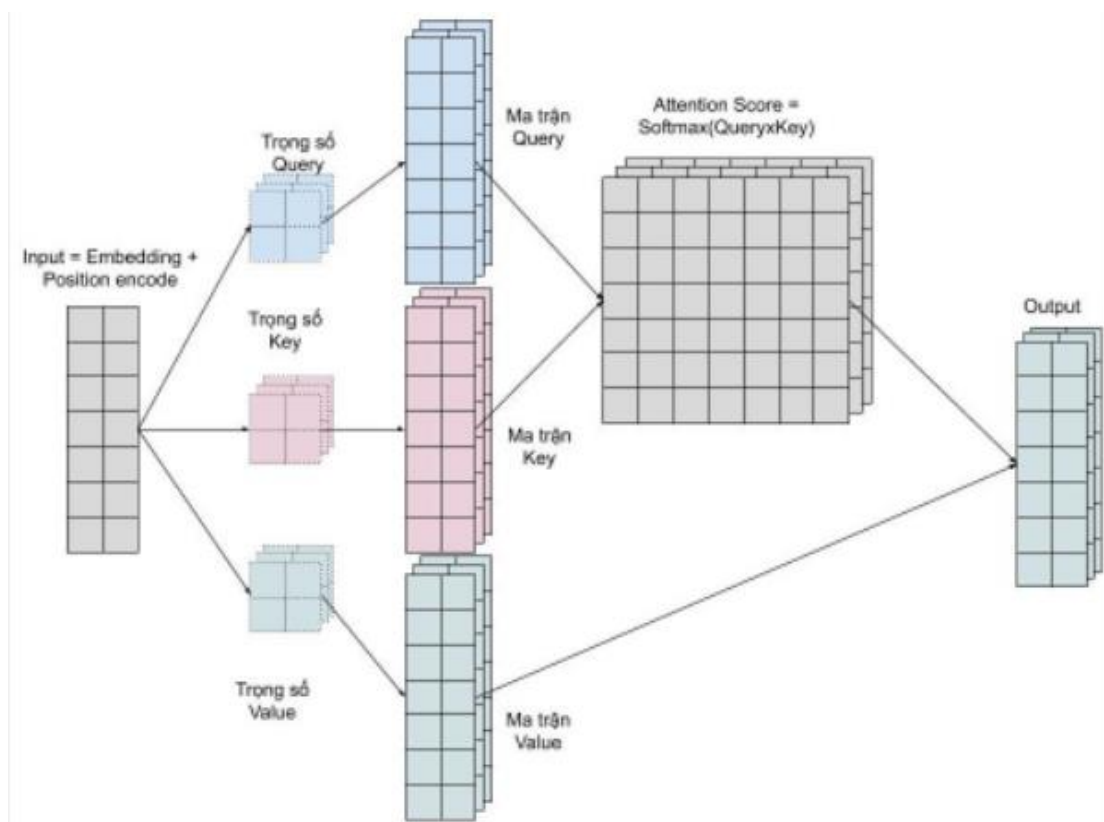
Encoder đầu tiên sẽ nhận ma trận biểu diễn của các từ đã được cộng với thông tin vị trí thông qua positional encoding. Sau đó, ma trận này sẽ được xử lý bởi Multi Head Attention. Multi Head Attention thật chất là self-attention, nhưng mà để mô hình có thể có chú ý nhiều pattern khác nhau, tác giả đơn giản là sử dụng nhiều self-attention.

+ **Encoder Layer:** Transformer EncoderLayer được tạo thành từ mạng tự điều chỉnh và chuyển tiếp.

### + Mutli Head Attention Layer

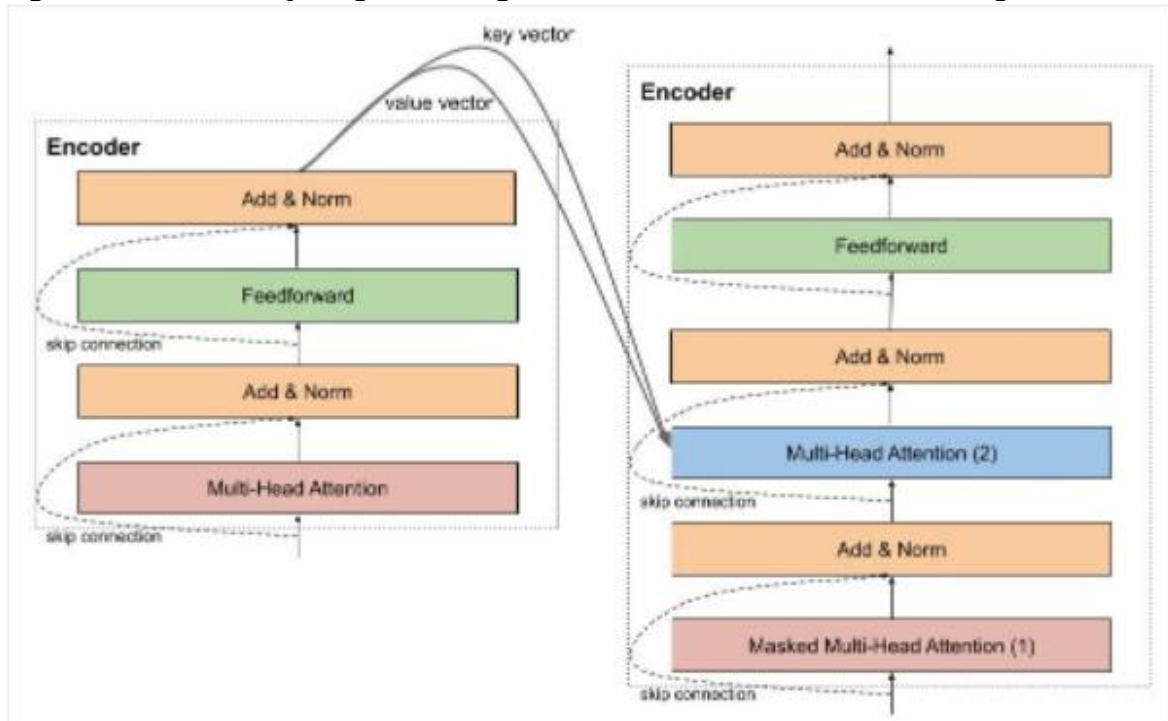
Chúng ta muốn mô hình có thể học nhiều kiểu mối quan hệ giữa các từ với nhau. Với mỗi self-attention, chúng ta học được một kiểu pattern, do đó để có thể mở rộng khả năng này, chúng ta đơn giản là thêm nhiều self-attention. Tức là chúng ta cần nhiều ma trận query, key, value mà thôi. Giờ đây ma trận trọng số key, query, value sẽ có thêm 1 chiều depth nữa.

Multi head attention cho phép mô hình chú ý đến đồng thời những pattern để quan sát được như sau.



- Chú ý đến từ kế trước của một từ
- Chú ý đến từ kế sau của một từ
- Chú ý đến những từ liên quan của một từ

+ **Decoder:** Decoder thực hiện chức năng giải mã vector của câu nguồn thành câu đích, do đó decoder sẽ nhận thông tin từ encoder là 2 vector key và value. Kiến trúc của decoder rất giống với encoder, ngoại trừ có thêm một multi head attention nằm ở giữa dùng để học mối liên quan giữ từ đang được dịch với các từ được ở câu nguồn.



+ **Decoder layer:** TransformerDecoderLayer được tạo thành từ mạng self-attn, multi-head-attn và nguồn cấp dữ liệu.

+ **seq2seq:** Mô hình Sequence to Sequence (thường được viết tắt là seq2seq) là một lớp đặc biệt của kiến trúc Mạng thần kinh lặp lại mà chúng tôi thường sử dụng (nhưng không bị hạn chế) để giải quyết các vấn đề ngôn ngữ phức tạp như Dịch máy, Trả lời câu hỏi, tạo Chatbots, Tóm tắt văn bản, v.v.

#### 4.5 Text Summary with Transformers

Mô hình của nhóm sẽ được định dạng như phần nêu trên nhưng có phần thay đổi khác ở các thông số

```
class CustomSchedule(tf.keras.optimizers.schedules.LearningRateSchedule):
    def __init__(self, d_model, warmup_steps=4000):
        super(CustomSchedule, self).__init__()

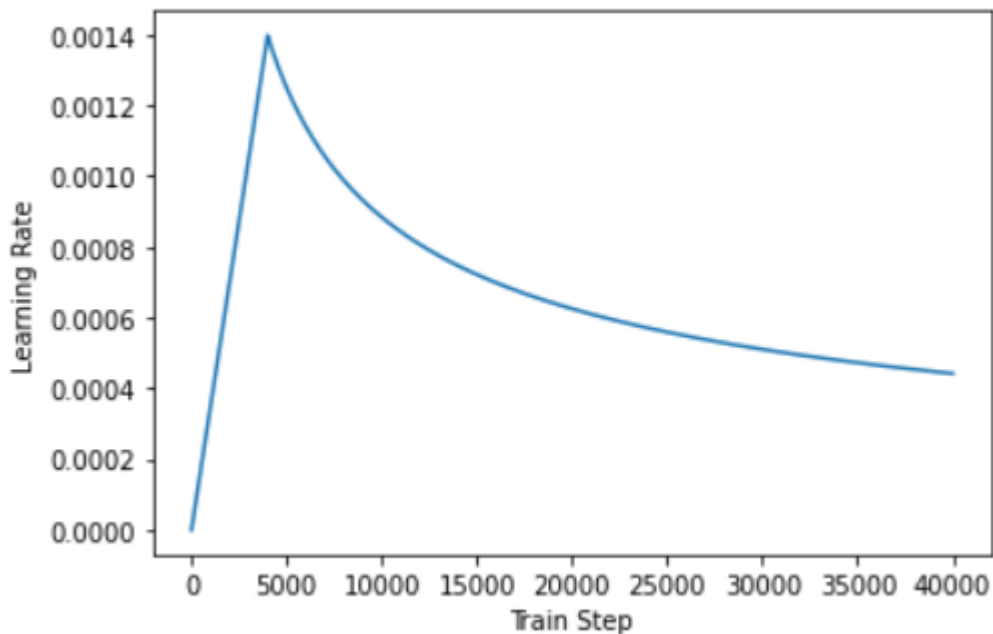
        self.d_model = d_model
        self.d_model = tf.cast(self.d_model, tf.float32)

        self.warmup_steps = warmup_steps

    def __call__(self, step):
        arg1 = tf.math.rsqrt(step)
        arg2 = step * (self.warmup_steps ** -1.5)

        return tf.math.rsqrt(self.d_model) * tf.math.minimum(arg1, arg2)
```

**Đồ thị thể hiện Learning Rate và Train Step**



## Kết quả:

Nhóm tiến hành train 30 epoch

Và tốc độ train của mô hình Transformers nhanh hơn nhiều so với các mô hình khác đã được đề cập ở phần trên

```
Epoch 30 Batch 0 Loss 2.7461 Accuracy 0.2531
Epoch 30 Batch 100 Loss 2.1954 Accuracy 0.2568
Epoch 30 Batch 200 Loss 2.3842 Accuracy 0.2595
Saving checkpoint for epoch 30 at checkpoints/ckpt-6
Epoch 30 Loss 2.3708 Accuracy 0.2605
Time taken for 1 epoch: 31.90864634513855 secs
```

## 4.6 Text Summarization With Transformers Pretrained

RoBERT là một mô hình dựa trên sự biến đổi (transformer), cho phép biểu diễn ngữ cảnh của một từ bằng cách dựa trên mối quan hệ của từ đó với các từ xung quanh. RoBERT khác biệt với các mô hình một chiều (unidirectional) khi chỉ học các biểu diễn từ trái qua phải hoặc từ phải qua trái. RoBERT được sử dụng để huấn luyện mô hình ngôn ngữ mặt nạ (masked language model) bằng cách học hai bài toán cùng một lúc là bài toán dự đoán từ và bài toán dự đoán câu. Với bài toán dự đoán từ, các từ trong một câu sẽ được che giấu (masked). Quá trình huấn luyện sẽ dự đoán từ bị che dấu bằng cách dựa vào các từ xung quanh.

Nhóm tiến hành Pretrained model của RoBERT để xem kết quả có được khả quan hơn không.

Và đây là một đoạn mã của mô hình cài đặt các thông số

```

# set special tokens
roberta_shared.config.decoder_start_token_id = tokenizer.bos_token_id
roberta_shared.config.eos_token_id = tokenizer.eos_token_id

# sensible parameters for beam search
# set decoding params
roberta_shared.config.max_length = 64
roberta_shared.config.early_stopping = True
roberta_shared.config.no_repeat_ngram_size = 3
roberta_shared.config.length_penalty = 2.0
roberta_shared.config.num_beams = 4
roberta_shared.config.vocab_size = roberta_shared.config.encoder.vocab_size

```

```

# set training arguments - these params are not really tuned, feel free to change
training_args = Seq2SeqTrainingArguments(
    output_dir= './small-datasets-checkpoints/',
    per_device_train_batch_size=batch_size,
    per_device_eval_batch_size=batch_size,
    predict_with_generate=True,
    # evaluate_during_training=True,
    do_train=True,
    do_eval=True,
    logging_steps=200, # set to 2000 for full training
    save_steps=5000, # set to 500 for full training
    eval_steps=7500, # set to 7500 for full training
    warmup_steps=3000, # set to 3000 for full training
    num_train_epochs=5, #uncomment for full training
    overwrite_output_dir=True,
    save_total_limit=50,
    fp16=True,
)

# instantiate trainer
trainer = Seq2SeqTrainer(
    model=roberta_shared,
    args=training_args,
    compute_metrics=compute_metrics,
    train_dataset=train_data_batch,
    eval_dataset=val_data_batch,
)
trainer.train()

```

## 5. Đánh giá

Để đánh giá độ chính xác của các mô hình, nhóm tiến hành chạy các mô hình với bộ dữ liệu test, và sử dụng phương pháp ROUGE. ROUGE viết tắt của Recall Oriented Understudy for Gist Evaluation, đây là phương pháp được coi là chuẩn mực và được sử dụng rộng rãi trong các nghiên cứu về tóm tắt văn bản. Điểm ROUGE-N được xác định như sau:

$$\text{ROUGE} - N = \frac{\sum_{S \in \{\text{ReferenceSummary}\}} \sum_{gram_n \in S} \text{Count}_{match}(gram_n)}{\sum_{S \in \{\text{ReferenceSummary}\}} \sum_{gram_n \in S} \text{Count}(gram_n)}$$

Trong đó  $\text{Count}_{match}(gram_n)$  là số lượng n-grams lớn nhất có trong văn bản tóm tắt sinh ra và văn bản tóm tắt tham chiếu

$\text{Count}(gram_n)$  là số lượng n-grams có trong văn bản tóm tắt tham chiếu.

Về cơ bản, nó là một tập hợp các thước đo để đánh giá tính năng tổng hợp tự động các văn bản cũng như dịch máy. Nó hoạt động bằng cách so sánh một bản tóm tắt hoặc bản dịch được tạo tự động với một tập hợp các bản tóm tắt tham chiếu (thường là do con người tạo ra). Nó được chia thành nhiều biện pháp khác nhau tùy thuộc vào độ chi tiết.

- ROUGE-1 đề cập đến sự chồng chéo của các đơn vị giữa bản tóm tắt hệ thống và bản tóm tắt tham chiếu. Được đánh giá dựa trên số 1-gram cùng có trong văn bản tóm tắt do mô hình sinh ra và văn bản tóm tắt tham chiếu.
- ROUGE-2 đề cập đến sự chồng chéo của bigrams giữa hệ thống và tóm tắt tham chiếu. Được đánh giá dựa trên số 2-gram cùng có trong văn bản tóm tắt do mô hình sinh ra và văn bản tóm tắt tham chiếu.
- ROUGE-L - đo chuỗi từ phù hợp dài nhất bằng cách sử dụng LCS, được đánh giá dựa trên chuỗi chung dài nhất có trong văn bản tóm tắt sinh ra và văn bản tóm tắt tham chiếu, đây là tham số quan trọng để đánh giá chất lượng của mô hình sinh tóm tắt.

## BẢNG ĐÁNH GIÁ ĐỘ CHÍNH XÁC THEO ĐỘ ĐO ROUGE CỦA CÁC MÔ HÌNH

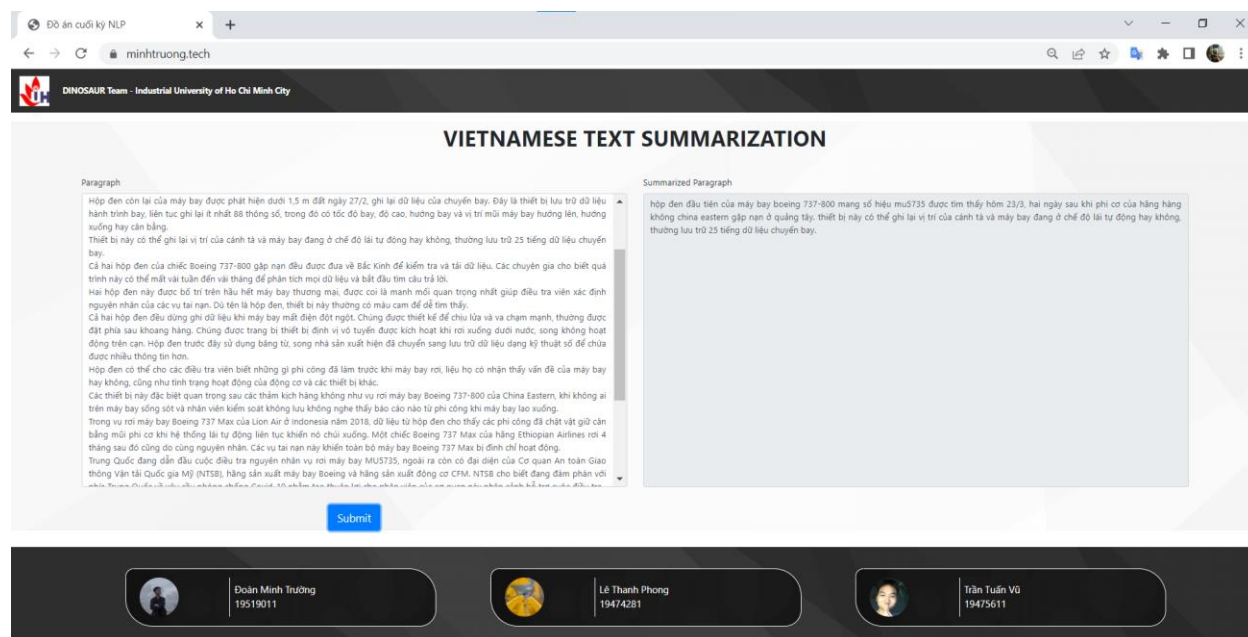
			METHODS					
			LSTMs	Bidirectional LSTM	Hybrid Architecture	GRU have AttentionLayer	Transformers	Transformers Pretrained RoBERT
ROUGE	Rouge-1	precision	0.18	0.06	0.09	0.06	0.1	0.58
		recall	0.31	0.52	0.17	0.29	0.13	0.52
		f-measure	0.2	0.09	0.11	0.1	0.11	0.51
	Rouge-2	precision	0.03	0	0	0.01	0.009	0.19
		recall	0.05	0	0	0.03	0.01	0.2
		f-measure	0.04	0	0	0.01	0.009	0.2
	Rouge-L	precision	0.18	0.06	0.07	0.06	0.083	0.42
		recall	0.3	0.5	0.12	0.03	0.107	0.4
		f-measure	0.2	0.09	0.08	0.09	0.09	0.41

## 6. Deploy

Giao diện của deploy

The screenshot shows a web browser window with the URL `minhtruong.tech`. The page title is "VIETNAMESE TEXT SUMMARIZATION". The interface includes a text input area labeled "Paragraph" with the placeholder "Enter your paragraph", a "Submit" button, and an output area labeled "Summarized Paragraph" with the placeholder "Output here". At the bottom, there is a footer with three team members: Đoàn Minh Trường (19519011), Lê Thanh Phong (19474281), and Trần Tuấn Vũ (19475611).

## Khi sử dụng



Dò án cuối kỳ NLP

minhtruong.tech

DINOSAUR Team - Industrial University of Ho Chi Minh City

### VIETNAMESE TEXT SUMMARIZATION

Paragraph

Hộp đen còn lại của máy bay được phát hiện dưới 1.5 m đất ngày 27/2, ghi lại dữ liệu của chuyến bay. Đây là thiết bị lưu trữ dữ liệu hành trình bay, liên tục ghi lại ít nhất 88 thông số, trong đó có tốc độ bay, độ cao, hướng bay và vị trí của máy bay hướng lên, hướng xuống hay cân bằng. Thiết bị này có thể ghi lại vị trí của cánh tà và máy bay đang ở chế độ lái tự động hay không, thường lưu trữ 25 tiếng dữ liệu chuyến bay.

Cả hai hộp đen của chiếc Boeing 737-800 gặp nạn đều được đưa về Bắc Kinh để kiểm tra và tái dữ liệu. Các chuyên gia cho biết quá trình này có thể mất vài tuần đến vài tháng để phân tích mọi dữ liệu và bắt đầu tìm câu trả lời.

Vai trò của hộp đen này được bổ trợ trên hầu hết máy bay thương mại, được coi là mảnh mồi quan trọng nhất giúp điều tra viên xác định nguyên nhân của các vụ tai nạn. Dù tên là hộp đen, thiết bị này thường có màu cam để dễ tìm thấy.

Cả hai hộp đen đều dùng ghi dữ liệu khi máy bay mất điện đột ngột. Chúng được thiết kế để chịu lửa và va chạm mạnh, thường được đặt phía sau khoang hàng. Chúng được trang bị thiết bị định vị vô tuyến được kích hoạt khi rơi xuống dưới nước, song không hoạt động trên cạn. Hộp đen trước đây sử dụng băng từ, song nhà sản xuất hiện đã chuyển sang lưu trữ dữ liệu dạng kỹ thuật số để chưa được nhiều thông tin hơn.

Hộp đen có thể cho các điều tra viên biết những gì phi công đã làm trước khi máy bay rơi, liệu họ có nhận thấy vấn đề của máy bay hay không, cũng như tình trạng hoạt động của động cơ và các thiết bị khác.

Các thiết bị này đặc biệt quan trọng sau các thảm kịch hàng không như vụ rơi máy bay Boeing 737-800 của China Eastern, khi không ai trên máy bay sống sót và nhân viên kiểm soát không lưu không nghe thấy báo cáo nào từ phi công khi máy bay lao xuống.

Trong vụ rơi máy bay Boeing 737 Max của Lion Air ở Indonesia năm 2018, dữ liệu từ hộp đen cho thấy các phi công đã chặt vật giữ cân bằng mũi phi cơ khi hệ thống lái tự động liên tục khiến nó chúi xuống. Một chiếc Boeing 737 Max của hãng Ethiopian Airlines rơi 4 tháng sau đó cũng do cùng nguyên nhân. Các vụ tai nạn này khiến toàn bộ máy bay Boeing 737 Max bị đình chỉ hoạt động.

Trung Quốc đang dẫn đầu cuộc điều tra nguyên nhân vụ rơi máy bay MU5735, ngoài ra còn có đại diện của Cơ quan An toàn Giao thông Vận tải Quốc gia Mỹ (NTSB), hãng sản xuất máy bay Boeing và hãng sản xuất động cơ CFM. NTSB cho biết đang đàm phán với nhiều hãng sản xuất thiết bị hàng không để có thể truy cập dữ liệu của các thiết bị này.

Summarized Paragraph

Hộp đen đầu tiên của máy bay Boeing 737-800 mang số hiệu MU5735 được tìm thấy hôm 23/3, hai ngày sau khi phi cơ của hãng hàng không China Eastern gặp nạn ở Quảng Tây. Thiết bị này có thể ghi lại vị trí của cánh tà và máy bay đang ở chế độ lái tự động hay không, thường lưu trữ 25 tiếng dữ liệu chuyến bay.

Submit

Đoàn Minh Trường  
19519011

Lê Thanh Phong  
19474281

Trần Tuấn Vũ  
19475611

## Định hướng nghiên cứu trong tương lai

Để tăng độ chính xác cho mô hình, một điều kiện quan trọng là xây dựng tập dữ liệu đầu vào word2vec chất lượng hơn, thể hiện chính xác hơn sự tương quan, mối liên hệ giữa các từ, các token. Do đó, việc xây dựng tập dữ liệu lớn và phong phú về chủ đề, đa dạng về mặt từ vựng là rất cần thiết cho mô hình tóm tắt văn bản tiếng Việt.

Hoặc nhóm sẽ tập chung vào một chủ đề nào đó ví dụ như: giáo dục, thể thao, giải trí... Để làm tăng độ chính xác cho mô hình hơn