ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN CƠ TIN HỌC

Hoàng Vũ Minh

TÓM TẮT NỘI DUNG TRUYỀN TẢI TỪ YOUTUBE VIDEO BẰNG PHƯƠNG PHÁP TÓM TẮT TRỪU TƯỢNG

Khóa luận tốt nghiệp đại học hệ chính quy
Ngành Khoa học Dữ liệu
Chương trình đào tạo chuẩn

Hà Nội - 2024

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN CƠ TIN HỌC

Hoàng Vũ Minh

TÓM TẮT NỘI DUNG TRUYỀN TẢI TỪ YOUTUBE VIDEO BẰNG PHƯƠNG PHÁP TÓM TẮT TRỪU TƯỢNG

Khóa luận tốt nghiệp đại học hệ chính quy
Ngành Khoa học dữ liệu
Chương trình đào tạo chuẩn

Cán bộ hướng dẫn: Th.S Nguyễn Tuấn Anh

Hà Nội - 2024

LỜI NÓI ĐẦU

Hiện nay, YouTube đã trở thành một trong những nền tảng chia sẻ video lớn nhất trên thế giới, với hàng tỷ video được tải lên mỗi ngày, điều này đặt ra một thách thức lớn với người sử dụng trong việc chọn lọc và tìm kiếm thông tin hữu ích.

Chính vì vấn đề này, em quyết định lựa chọn đề tài "Tóm tắt nội dung từ Video trên YouTube bằng phương pháp tóm tắt trừu tượng" làm nội dung thực hiện luận văn tốt nghiệp, với mong muốn tạo ra một giải pháp giúp người dùng giảm thời gian tìm kiếm và nắm bắt thông tin khi tiếp cận với những video trên nền tảng này.

Trong luận văn và đề tài này, em đã nghiên cứu và sử dụng mô hình học sâu hiện đang là SOTA (State-of-the-art) trong tác vụ tóm tắt trừu tượng văn bản tiếng Việt - ViT5 cùng bộ dữ liệu được gán nhãn mới với mục tiêu tạo ra những văn bản tóm tắt có chất lượng cao. Qua thử nghiệm và đánh giá ban đầu, mô hình đã đạt được kết quả khả quan, có thể ứng dụng trong xây dựng hệ thống tóm tắt, mang lại hiệu quả nhất định.

Bố cục của khóa luận gồm có 3 chương chính: Chương 1: Tổng quan về đề tài, Chương 2: Các kiến thức cơ sở, Chương 3: Phương pháp thực hiện nghiên cứu.

Do thời gian thực hiện luận văn cũng như kiến thức và tài nguyên còn hạn chế nên khi thực hiện khóa luận, em không thể tránh khỏi những sai sót. Em kính mong nhận được sự góp ý và ý kiến phản biện từ quý thầy cô nhằm có thêm ý tưởng và cơ sở để cải thiện đề tài trong tương lai. Em xin chân thành cảm ơn!

LỜI CẢM ƠN

Lời đầu tiên, em xin được bày tỏ lòng biết ơn sâu sắc tới Thầy hướng dẫn khoa học, **Th.S Nguyễn Tuấn Anh** thuộc đơn vị PIXTA Vietnam đã tận tình hướng dẫn em trong suốt thời gian thực hiện luận văn tốt nghiệp này.

Em xin chân thành cảm ơn thầy giáo **Đỗ Quốc Trường** và thầy giáo **Ngô Thế Quyền** đã có những ý kiến nhận xét và góp ý hết sức chi tiết, giúp em có thể hoàn thành luận văn một cách tốt nhất.

Em cũng xin gửi lời cảm ơn chân thành đến các thầy cô trong Khoa Toán - Cơ - Tin học, Trường Đại học Khoa học Tự nhiên, Đại học Quốc gia Hà Nội, đặc biệt là cô **Nguyễn Thị Minh Huyền**, giảng viên chủ nhiệm lớp K65A5 đã luôn tận tình giảng dạy, truyền đạt những kiến thức quý giá cho em trong suốt thời gian học tập tại nhà trường. Cảm ơn Khoa Toán - Cơ - Tin học đã tạo điều kiện, giúp em thực hiện đề tài này một cách tốt nhất.

Xin chân thành cảm ơn những tác giả của những công trình nghiên cứu trước đã để lại nguồn tư liệu tham khảo cùng những kinh nghiệm quý báu, giúp em có thêm kiến thức và cơ sở để hoàn thành luận văn.

Lời cuối cùng, em xin được gửi lời cảm ơn đến gia đình, người thân, bạn bè, đồng nghiệp, đã luôn ở bên động viên và tiếp thêm động lực cho em trong suốt quá trình nghiên cứu, thu thập tài liệu và hoàn thành luận văn.

Hà Nội, ngày 20 tháng 5 năm 2024

Sinh viên

Hoàng Vũ Minh

Mục lục

Danh sách từ viết tắt

IDF Inverse Document Frequency

LCS Longest Common Subsequence

LSTM Long-short Term Memory

RNN Recurrent Neural Network

SOTA State Of The Art

TF Text Frequency

Danh sách hình vẽ

Chương 1

Tổng quan về đề tài

1.1 Mục tiêu và ý nghĩa của đề tài

Đề tài được thực hiện nhằm áp dụng những kiến thức, kỹ thuật trong lĩnh vực xử lý ngôn ngữ tự nhiên, ứng dụng vào xây dựng một hệ thống tóm tắt nội dung truyền tải từ Video Youtube bằng phương pháp tóm tắt trừu tượng văn bản. Mục tiêu cốt yếu của đề tài là tạo ra một công cụ giúp người sử dụng có thể nhanh chóng nắm bắt được nội dung chính được đề cập đến trong một Youtube Video, và tùy vào mức độ sử dụng và hiểu thông tin, họ có thể quyết định xem để hiểu thêm về vấn đề đó. Qua đó giúp giảm thiểu thời gian và công sức người sử dụng cần bỏ ra để nắm bắt và tìm kiếm thông tin hữu ích.

Đề tài có ý nghĩa không chỉ về mặt nghiên cứu, mà hoàn toàn có thể ứng dụng trong các lĩnh vực như giáo dục, học trực tuyến, truyền thông, giải trí..., mở ra nhiều cơ hội giúp người sử dụng nâng cao hiệu suất công việc nhưng vẫn tiết kiệm được thời gian và công sức.

1.2 Hướng tiếp cận của đề tài

Trong những nghiên cứu trước đây, khi nhắc tới tóm tắt nội dung truyền tải từ Youtube Video, những nhà nghiên cứu thường hướng tới việc tóm tắt nội dung dựa trên hình ảnh từ các khung hình trong video, hay sử dụng các đa phương pháp (multi-method), đa mô hình (multi-model) kết hợp khả năng xử lý cả tóm tắt hình ảnh dựa vào khung hình, âm thanh và văn bản. Các phương pháp này được đề cập đến trong một số nghiên cứu:

- ASoVS: Abstractive Summarization of Video Sequences - được trình bày bởi Aniqa Dilawari, Muhammad Usman Ghani Khan [1].
- AudioVisual Video Summarization - được trình bày bởi Bin Zhao, Maoguo Gong, Xuelong Li [2].
- Progressive Video Summarization via Multimodal Self-supervised Learning - được trình bày bởi Li Haopeng, Ke Qiuhong, Gong Mingming, Tom Drummond [3].

Tuy nhiên, trong luận văn này, em đề cập đến một cách tiếp cận thuần về xử lý văn bản hơn, bằng cách lấy transcript bằng tiếng Anh của các video từ YouTube-Transcript-API kết hợp với việc sử dụng Google Translate API nhằm chuyển hóa đoạn văn bản về tiếng Việt, từ đó chuyển thành bài toán xây dựng mô hình tóm tắt trừu tượng cho văn bản tiếng Việt. Nguyên nhân cho cách tiếp cận này:

- **Mục đích sử dụng:** Khi theo dõi một video nhằm mục đích thu thập thông tin, người dùng thường theo dõi các video có trình bày ngôn ngữ tự nhiên, đặc biệt là âm thanh. Chính vì vậy, đôi

khi việc sử dụng cả yếu tố hình ảnh không phải lúc nào cũng là cần thiết với mục đích giúp người đọc nắm được nội dung chính được truyền tải.

- **Sự phát triển của YouTube Transcript API:** YouTube Transcript API hiện nay đã được cải tiến và nâng cấp rất nhiều so với trước đây, điều này cho phép công cụ này có thể tạo ra văn bản chuyển hóa tự động âm thanh thành văn bản tiếng Anh của hầu hết các video, dù cho nó có ở bất kỳ loại ngôn ngữ nào.
- **Kết hợp với việc sử dụng các công cụ dịch thuật**, ta có thể dễ dàng nhận được đoạn văn bản tiếng Việt chứa thông tin được trình bày trong video, qua đó có thể đưa vào các mô hình và các bước xử lý tiếp theo một cách dễ dàng.

1.3 Đối tượng tiếp cận của đề tài

Đối tượng mà đề tài hướng đến là những video truyền tải và diễn giải thông tin bằng âm thanh, bởi những lý do đã được trình bày ở mục 1.2.

Chương 2

Kiến thức cơ sở

2.1 Các phương pháp tóm tắt văn bản

Tóm tắt văn bản là quá trình rút gọn nội dung của một văn bản thành một văn bản khác ngắn hơn mà vẫn giữ được những ý chính và thông tin quan trọng. Các phương pháp tóm tắt văn bản có thể được chia thành hai loại chính: tóm tắt trích xuất (extractive summarization) và tóm tắt trừu tượng (abstractive summarization).

2.1.1 Tóm tắt trích xuất

Phương pháp tóm tắt trích xuất được thực hiện bằng việc xác định và chọn lọc các câu, cụm từ quan trọng từ văn bản ban đầu, sau đó hợp chúng lại với nhau để tạo thành một đoạn văn bản rút gọn mới. Điểm quan trọng của phương pháp này nằm ở việc xác định mức độ quan trọng của các câu, cụm từ. Một cách thường được sử dụng đó là xây dựng một thuật toán tính điểm cho các câu, sau đó xếp hạng chúng theo mức độ điểm cùng với tạo ra một ngưỡng quan trọng. Điều

này có thể được thực hiện bằng nhiều phương pháp khác nhau như TF-IDF, TextRank hay sử dụng các mô hình học máy. Bài luận này chủ yếu tập trung và phương pháp tóm tắt trừu tượng, chi tiết về phương pháp tóm tắt trích xuất có thể tìm thấy tại [4].

Hình 2.1 Minh họa về tóm tắt trích xuất

2.1.2 Tóm tắt trừu tượng

Khác với phương pháp tóm tắt trích xuất, tóm tắt trừu tượng tạo ra một văn bản tóm tắt mới sáng tạo hơn, vẫn tóm lược được ý chính của văn bản ban đầu tuy nhiên không nhất thiết phải xuất hiện các từ, các câu đã có trong văn bản gốc. Phương pháp này bao gồm các kỹ thuật hiểu và sinh văn bản trong lĩnh vực xử lý ngôn ngữ tự nhiên ngôn ngữ tự nhiên, đòi hỏi sự phức tạp trong xử lý nên các mô hình học sâu thường được sử dụng, một số mô hình phổ biến có thể kể đến như RNNs, LSTMs hay các mô hình transformers.

Tóm tắt trừu tượng cho phép linh hoạt và sáng tạo hơn so với các phương pháp trích lọc, vì nó có thể thay đổi và tái cấu trúc thông tin một cách phù hợp tùy theo mục đích, đổi tượng mà người làm mô hình hướng đến. Tuy nhiên, nó cũng đặt ra những thách thức về độ chính xác, mạch lạc cũng như dữ liệu sử dụng. Trong đề tài này, em

lựa chọn sử dụng mô hình ViT5 cho việc triển khai phương pháp tóm tắt trừu tượng, dựa vào những ưu điểm và kết quả mô hình này mang lại.

Hình 2.2 Minh họa về tóm tắt trừu tượng

2.2 Bộ chỉ số ROUGE

ROUGE là viết tắt của Recall Oriented Understudy for Gisting Evaluation, là một tập các chỉ số sử dụng để đánh giá chất lượng của bản tóm tắt bằng cách so sánh nó với các bản tóm tắt tham chiếu (thường do con người tạo ra). ROUGE đánh giá mức độ trùng nhau giữa bản tóm tắt được tạo tự động và các bản tóm tắt do con người tạo ra bằng cách đếm các đơn vị trùng khớp như n-gram (chuỗi từ ngắn), chuỗi từ, và cặp từ [5].

2.2.1 ROUGE-n

N-gram:

N-grams là các chuỗi liên tiếp gồm n phần tử trong một mẫu văn bản hoặc lời nói cụ thể. Các phần tử có thể là âm vị, âm tiết, chữ

cái, từ hoặc cặp cơ sở tùy thuộc vào ứng dụng cụ thể. N-grams được sử dụng trong nhiều lĩnh vực của ngôn ngữ học tính toán và phân tích văn bản. Chúng là một phương pháp đơn giản và hiệu quả cho các nhiệm vụ khai thác văn bản và xử lý ngôn ngữ tự nhiên (NLP), như dự đoán văn bản, sửa chữa chính tả, mô hình hóa ngôn ngữ và phân loại văn bản. Một vài n-gram phổ biến là 1-gram (uni-gram), 2-gram (bi-gram), 3-gram (tri-gram). Chi tiết hơn về n-grams có thể được tìm thấy trong [6].

ROUGE-n

ROUGE-n: đo đo số lượng n-gram phù hợp giữa văn bản do mô hình tạo ra và văn bản tham chiếu. Về mặt hình thức, ROUGE-n là phép đo sự tương đồng n-gram giữa một bản tóm tắt (do mô hình tạo ra) và một tập hợp các bản tóm tắt tham chiếu. ROUGE-n được tính như sau [5]:

$$\text{ROUGE-n} = \frac{\sum_{S \in \text{Reference Summaries}} \sum_{\text{gram}_n \in S} \text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{S \in \text{Reference Summaries}} \sum_{\text{gram}_n \in S} \text{Count}(\text{gram}_n)}$$

Trong đó:

- n đại diện cho độ dài của n-gram.
- $\text{Count}_{\text{match}}(\text{gram}_n)$ là số lượng lớn nhất n-gram xuất hiện ở cả văn bản tóm tắt được tạo và tập văn bản tóm tắt tham chiếu.

2.2.2 ROUGE-L: Longest Common Subsequence

ROUGE-L là một dạng khác của ROUGE, được sử dụng để đánh giá các văn bản tóm tắt dựa trên sự trùng khớp về chuỗi con dài nhất. Chúng ta sẽ đề cập đến 2 mức độ của ROUGE-L: mức độ câu (sentence-level), sẽ được đề cập đến với tên ROUGE-L và mức độ tóm tắt (summary-level), được đề cập đến với tên ROUGE-L Sum.

Chuỗi con chung dài nhất:

Một chuỗi Z là chuỗi con của chuỗi X nếu tồn tại một chuỗi chỉ số tăng dần của X sao cho các phần tử tương ứng của Z và X bằng nhau [5]. Ta ký hiệu X là câu tóm tắt tham chiếu, Y là câu tóm tắt được tạo. Chuỗi con chung dài nhất (LCS) của X và Y phản ánh tỷ lệ các từ trong X cũng xuất hiện trong Y. Chuỗi từ chung dài nhất không nhất thiết phải liên tiếp nhưng vẫn phải theo đúng thứ tự, được chia sẻ giữa cả hai chuỗi.

Ví dụ:
Chuỗi 1: `the cat is on the table`
Chuỗi 2: `the dog bites the table`
LCS: `the the table`

ROUGE-L

Một chuỗi Z là chuỗi con của chuỗi X nếu tồn tại một chuỗi chỉ số tăng dần của X sao cho các phần tử tương ứng của Z và X bằng nhau [5]. Ta ký hiệu X là câu tóm tắt tham chiếu, Y là câu tóm tắt được tạo. Chuỗi con chung dài nhất (LCS) của X và Y phản ánh tỷ lệ các từ trong X cũng xuất hiện trong Y. Chuỗi từ chung dài nhất không nhất

thiết phải liên tiếp nhưng vẫn phải theo đúng thứ tự, được chia sẻ giữa cả hai chuỗi.

Để dễ dàng áp dụng LCS trong đánh giá tóm tắt, một câu tóm tắt có thể được coi là một chuỗi các từ. Như vậy, có thể nói LCS của hai câu tóm tắt càng dài thì hai bản tóm tắt càng giống nhau. Chỉ số F dựa trên LCS được sử dụng để ước lượng sự tương đồng giữa hai bản tóm tắt X có độ dài m và Y có độ dài n, giả sử X là câu tóm tắt tham chiếu và Y là câu tóm tắt ứng viên, như sau [5]:

$$R = \frac{LCS(X,Y)}{m}$$
$$P = \frac{LCS(X,Y)}{n}$$
$$F = \frac{(1+\beta^2) \cdot R \cdot P}{R + \beta^2 \cdot P}$$

Trong đó:

- LCS(X, Y) là độ dài của chuỗi con chung dài nhất giữa X và Y.
- β (beta): Là một hằng số, trong DUC (Document Understanding Conference), β được đặt rất cao (khoảng 8), điều này khiến recall được xem xét chú trọng hơn so với precision. β^2 được đặt ở dưới mẫu, nghĩa là khi β càng lớn, $\beta^2 \cdot P$ sẽ tiến dần về 0, khi đó chỉ còn phụ thuộc vào Recall.
- F-measure: Một thước đo kết hợp cả recall và precision, được sử dụng rộng rãi để đo lường độ chính xác trong nhiều lĩnh vực. Ta gọi F-measure trong trường hợp này là ROUGE-L.

Dựa vào các công thức và lý thuyết, ta có thể nói rằng nếu ROUGE-L càng cao thì câu tóm tắt càng tốt, càng sát với bản tóm tắt

tham chiếu. ROUGE-L bằng 1 khi 2 bản trùng khớp hoàn toàn và bằng 0 khi không có gì trùng khớp.

ROUGE-L-SUM

Tiếp theo, phần này sẽ trình bày cách tính ROUGE-L ở summary-level, hay còn được đề cập tới với tên ROUGE-L Sum. Trong mức độ tóm tắt, ta áp dụng LCS giữa mỗi câu tóm tắt tham chiếu r_i và mỗi câu tóm tắt được tạo c_j. Cho một tóm tắt tham chiếu gồm u câu chứa tổng cộng m từ và một tóm tắt ứng viên gồm v câu chứa tổng cộng n từ.

Công thức tính ROUGE-L-SUM [5]:

$$R = \frac{\sum_{i=1}^{u} \text{LCS}(r_i, c_j)}{m}$$
$$P = \frac{\sum_{j=1}^{v} \text{LCS}(r_i, c_j)}{n}$$
$$F = \frac{(1 + \beta^2) \cdot R \cdot P}{R + \beta^2 \cdot P}$$

Trong đó:

- $\text{LCS}(r_i, c_j)$ là độ dài của chuỗi con chung dài nhất giữa câu tóm tắt tham chiếu r_i và câu tóm tắt được tạo c_j.
- u là số câu trong tóm tắt tham chiếu.
- v là số câu trong tóm tắt được tạo.
- m là tổng số từ trong tất cả các câu trong tóm tắt tham chiếu.
- n là tổng số từ trong tất cả các câu trong tóm tắt được tạo.

β (beta): trong trường hợp này cũng được đặt cho một số cao (thường là 8) để ảnh hưởng của Recall được tốt hơn (vì recall được

tính dựa theo m là số từ trong văn bản tham chiếu).

2.3 Mô hình T5

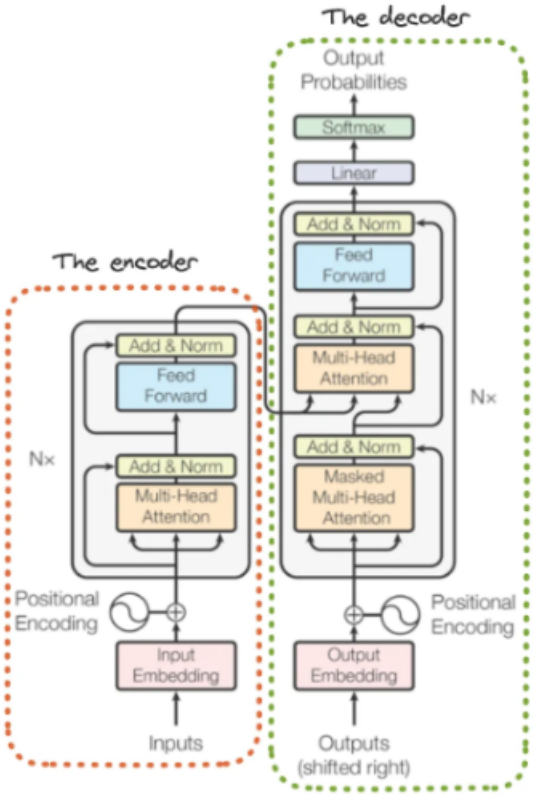
ViT5 là một mô hình encoder-decoder, được thiết kế dựa trên kiến trúc Encoder-Decoder [9] và T5 [7] , được tiền huấn luyện trên một tập dữ liệu tiếng việt có dung lượng lớn và chất lượng cao. Kiến trúc này kết hợp những tiến bộ của Transformers trong việc xử lý ngôn ngữ tự nhiên và mô hình hóa ngôn ngữ, giúp tăng cường khả năng của ViT5 trong việc thực hiện các tác vụ. Nhóm tác giả đánh giá hiệu suất mô hình trên hai tác vụ sinh văn bản: Tóm tắt văn bản trừu tượng (Abstractive Text Summarization) và Nhận dạng thực thể (Named Entity Recognition) [8].

Tuy nhiên, thực tế cả mô hình và nghiên cứu về ViT5, T5 không đóng góp một kiến trúc hoặc phương pháp huấn luyện mới, thay vào đó, họ hướng đến việc dựa trên các kỹ thuật hiện có, xem xét nhiều khía cạnh trong việc học chuyển giao (transfer learning), các phương pháp tiền huấn luyện, dữ liệu, tinh chỉnh...nhằm xác định phương pháp hiệu quả nhất. Điểm đặc biệt là các khía cạnh này đều được nghiên cứu thông qua một định dạng văn bản thành văn bản (text - to - text format) từ đầu đến cuối, trong các tác vụ khác nhau, điều này sẽ được trình bày ở trong mục 2.3.2.

2.3.1 Kiến trúc Transformer Encoder-Decoder

Kiến trúc Encoder-Decoder được trình bày bởi Vaswani và các cộng sự trong "Attention is All you need" [9]. Những thông tin được trình bày dưới đây cũng được tuân theo những kiến thức và khái niệm

được đề cập đến trong bài báo này.



Hình 2.3 Kiến trúc encoder-decoder (Attention is All you need)

Encoder: nằm ở khối bên trái, nhận đầu vào và xây dựng một biểu diễn của nó (các feature). Điều này có nghĩa là mô hình được tối ưu hóa để đạt được sự hiểu biết từ đầu vào. Khối Encoder bao gồm $N = 6$ lớp con giống nhau với mỗi lớp có 2 thành phần:

- Lớp Multi-head self-attention: Đây là một cơ chế quan trọng trong Transformer, giúp mô hình dành sự chú ý (attention) vào các phần quan trọng của đầu vào. Bằng cách này, mô hình có thể hiểu mối quan hệ giữa các từ hoặc các phần của đầu vào một cách hiệu quả.

- Mạng nơ-ron truyền thẳng kết nối đầy đủ (Fully Connected Feed Forward Network): Đây là một lớp mạng nơ-ron thông thường, mỗi đơn vị trong lớp này được kết nối với tất cả các đơn vị của lớp trước và lớp sau nó. Mục tiêu của lớp này là tạo ra một biểu diễn mới của đầu vào, giúp mô hình hiểu sâu hơn về thông tin đầu vào.

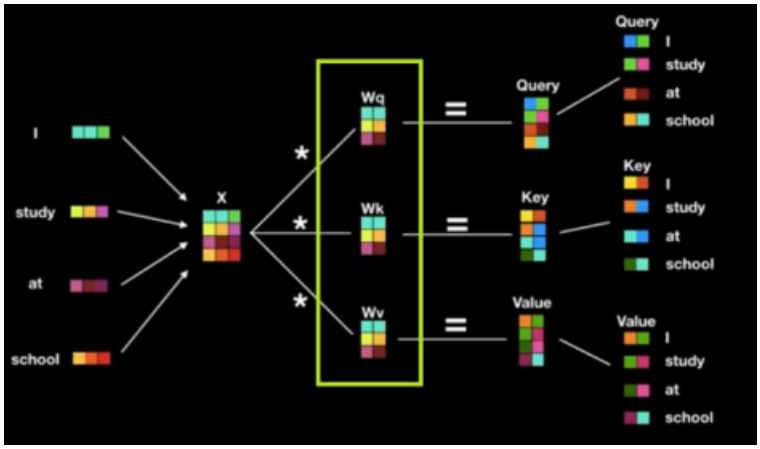
- Residual connection (phần add & norm) được sử dụng xung quanh mỗi lớp con, điều này có nghĩa là đầu ra của mỗi lớp con được cộng với đầu vào của nó trước khi đi qua chuẩn hóa lớp, giúp mô hình dễ dàng học được biến thể hoặc thông tin mới từ đầu vào ban đầu mà không mất mát thông tin.

- Các lớp con trong mô hình, cũng như các lớp nhúng, tạo ra đầu ra có kích thước $d_{\text{model}} = 512$, giúp đảm bảo rằng các đầu vào và đầu ra của các lớp con có cùng kích thước.

Decoder: nằm ở khối bên phải, sử dụng biểu diễn (feature) từ bộ mã hóa cùng với các đầu vào khác để tạo ra một chuỗi mục tiêu. Khối này được tối ưu hóa với nhiệm vụ tạo ra các đầu ra. Khối Decoder cũng bao gồm N = 6 lớp con giống nhau với nhưng mỗi lớp có 3 thành phần, 2 phần tương tự với Encoder và thêm một phần thực hiện Multi-head self-attention qua đầu ra của Encoder.

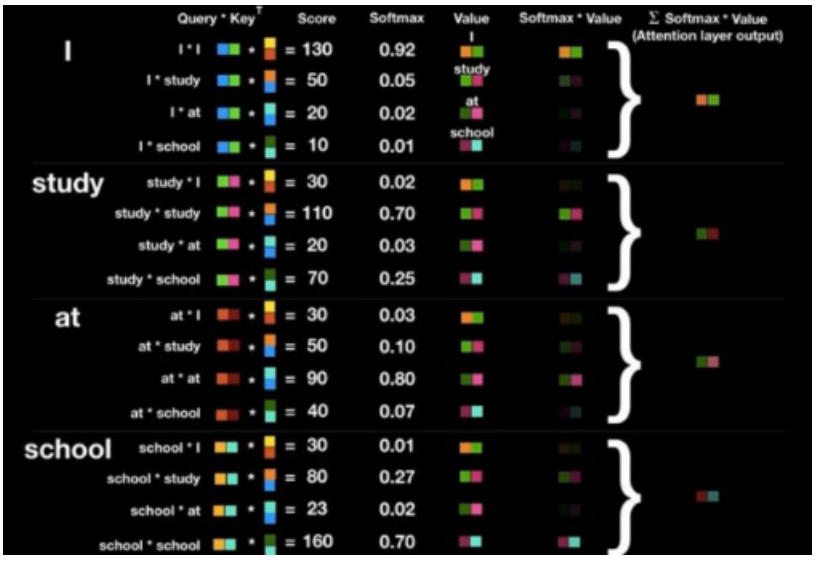
Self-Attention (Scaled Dot-Product Attention)

Cơ chế self-attention được thể hiện qua hình ảnh minh họa dưới đây:



Hình 2.4 Cơ chế self-attention

Trong một hàm attention, chúng ta sử dụng các biến $\mathbf{W}_q$, $\mathbf{W}_k$, $\mathbf{W}_v$ là các tham số có thể học được. Các biến này được sử dụng để biến đổi các vector query, key và value tương ứng để tạo ra các biểu diễn cuối cùng của token. Tính điểm cho tất cả các tokens trong câu ta có:

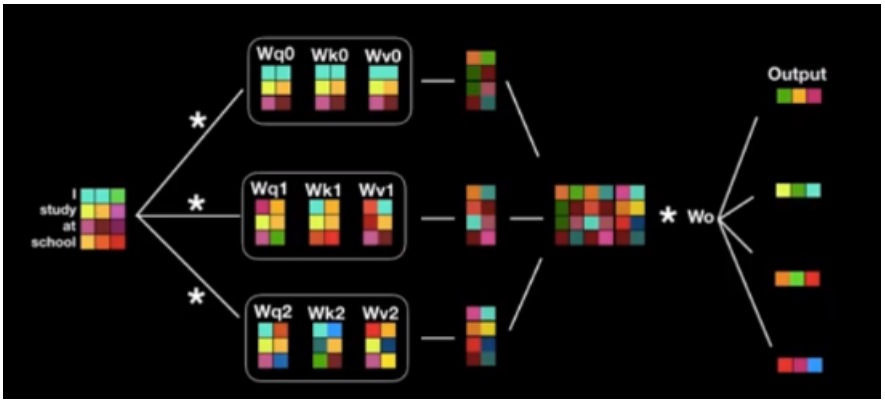


Hình 2.5 Tính toán self-attention cho tất cả các tokens

Công thức tính Attention [9]:

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

Sau mỗi quá trình Scale dot production chúng ta sẽ thu được 1 ma trận attention (Wq, Wk, Wv), gọi là một head. Lặp lại quá trình Self-Attention nhiều lần với các tập (Wq, Wk, Wv) khác nhau, ta được Multi-head Attention. Điều đó được minh họa trong hình ảnh dưới đây.



Hình 2.6 Minh họa Multi-head Attention

Công thức tính Multi-head Attention [9]:

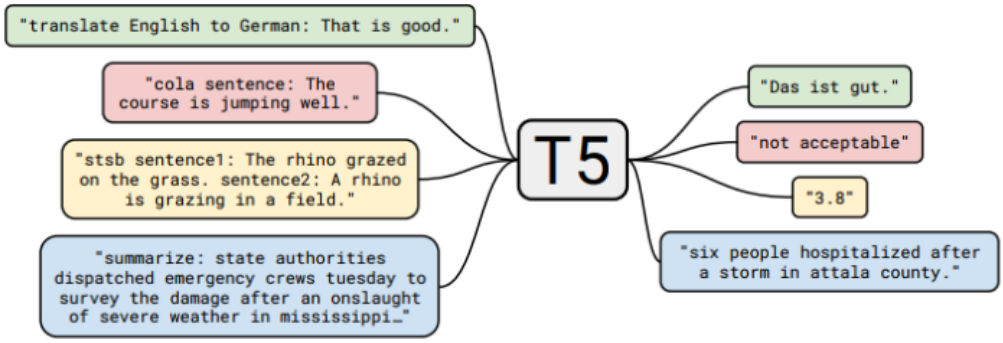
$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}\left(\text{head}_1, ..., \text{head}_h\right)\mathbf{W}^O$$

Trong đó:

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V), \quad \text{khi} \quad i = 1, 2, ..., h$$

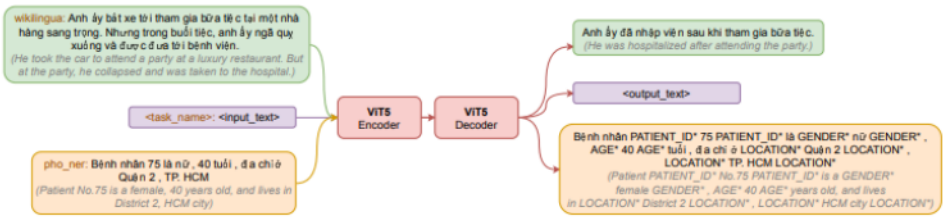
2.3.2 Text-to-Text Format

Điểm đặc biệt ở T5 cũng như ViT5 là các mô hình này sử dụng đầu vào là văn bản và đầu ra là văn bản cho tất cả các tác vụ NLP (được thể hiện trong 2 sơ đồ minh họa bên dưới), thậm chí là cả tác vụ phân loại văn bản, điều này khác với các model BERT (Bidirectional Encoder Representation from Transformer). Điều này có nghĩa là chúng ta có thể dùng chung mô hình, mục tiêu, quy trình huấn luyện và giải mã cho tất cả các tác vụ phổ biến trong xử lý ngôn ngữ tự nhiên. Ví dụ minh họa được thể hiện ở sơ đồ minh họa trích trong [7]:



Hình 2.7 Minh họa cho text-to-text format trong T5

Sơ đồ minh họa định dạng text-to-text và kiến trúc encoder-decoder với ViT5, được trích trong [8]:



Hình 2.8 Minh họa cho text-to-text format trong ViT5

Cụ thể hơn, điều này có thể được thực hiện bằng việc thêm tiền

tố, hay thuật ngữ gọi là “prefix” ở chuỗi đầu vào. Điều này giúp T5 biết được tác vụ nó thực hiện. Trong 2 sơ đồ minh họa, các tiền tố đó là “summarize”, "cola text" hay “translate English to German”.

Tuy nhiên, có một số vấn đề có thể xảy ra với định dạng text-to-text, cụ thể là trong bài toán phân loại văn bản. Thông thường, những mô hình Transformer thuộc dạng Encoder-Only như BERT được sử dụng cho bài toán này vì nó tạo ra đầu ra là các biểu diễn (feature), từ đó ta có thể dễ dàng tính toán xác suất rơi vào các nhãn của văn bản đầu vào. Với định dạng văn bản mà các mô hình T5 tạo ra, có thể văn bản sinh ra sẽ không tương ứng với bất kỳ một nhãn nào. Trong trường hợp này, các nhà phát triển sẽ coi nó là sai, tuy nhiên theo các nhà phát triển T5, họ chưa từng gặp vấn đề nào như vậy với các mô hình của họ. Chi tiết thêm về vấn đề này có thể tìm thấy tại [7].

2.3.3 Bộ từ vựng và bộ dữ liệu pre-training ViT5

Bộ từ vựng của ViT5 được nhóm tác giả xử lý trên một tập dữ liệu 5GB là tập con của dữ liệu tiền huấn luyện, được điều chỉnh và xử lý bởi nhiều kỹ thuật xử lý ngôn ngữ tự nhiên như chuẩn hóa dấu câu, từ viết hoa. Kích thước bộ từ vựng được cố định ở mức 36K sub-words, nhóm tác giả cũng huấn luyện lại tokenizer dựa trên thuật toán SentencePiece trên tập dữ liệu trên nhằm tạo ra một bộ từ vựng hiệu quả [8].

Về dữ liệu tiền huấn luyện, model ViT5 được tiền huấn luyện trên bộ dữ liệu CC100, là một bộ dữ liệu đơn ngữ (mono-language) cho hơn 100 ngôn ngữ trong đó có Tiếng Việt, được thu thập từ các website. Kích thước của phần dữ liệu tiền huấn luyện sau khi được lọc và xử lý là 69GB đoạn văn ngắn cho mô hình độ dài đầu vào là 256 và 71GB đoạn văn dài cho mô hình độ dài đầu vào 1024. Chi tiết hơn

về dữ liệu được trình bày trong [8]. Trong đề tài này, em sử dụng mô hình có độ dài 1024 vì phù hợp với nhu cầu sử dụng của hệ thống tóm tắt.

2.3.4 Kết quả mô hình ViT5 trên tác vụ tóm tắt trừu tượng văn bản

Hiệu quả của việc tinh chỉnh mô hình ViT5 trên tác vụ tóm tắt trừu tượng văn bản được đo lường qua 2 bộ dữ liệu nổi tiếng là Wikilingua và Vietnews. Wikilingua là một tập dữ liệu bao gồm 18 ngôn ngữ, được sử dụng cho nhiệm vụ tóm tắt trừu tượng. Các cặp nội dung và tóm tắt được trích xuất từ WikiHow. Các phần dữ liệu tiếng Việt được dịch từ các bài viết gốc tiếng Anh và đã được đội ngũ dịch thuật quốc tế của WikiHow xem xét lại nhằm đảm bảo độ chính xác cũng như chất lượng cho bộ dữ liệu này. Bộ dữ liệu được giới thiệu bởi Ladhak và các cộng sự trong [10].

Vietnews là bộ dữ liệu tiếng Việt được tạo bằng cách thu thập dữ liệu tin tức từ 3 trang web tin tức nổi tiếng của Việt Nam: *tuoitre.vn, vnexpress.net và nguoiduatin.vn*. Các bài viết được thu thập từ năm 2016 đến năm 2019. Sau đó, các tác giả loại bỏ tất cả các bài viết liên quan đến bảng câu hỏi, bình luận phân tích và dự báo thời tiết (do tính phù hợp cho nhiệm vụ tóm tắt văn bản), tập dữ liệu cuối cùng chỉ chứa các sự kiện và tin tức. Bộ dữ liệu được giới thiệu bởi Kiet Van Nguyen và các cộng sự trong [11].

Kết quả huấn luyện mô hình ViT5 trên hai bộ dữ liệu cho tác vụ tóm tắt trừu tượng văn bản được trình bày ở bảng bên dưới [8], với các metric ROUGE-1, ROUGE-2, ROUGE-L đã được trình bày ở mục 2.2.

Models	WikiLingua			Vietnews		
	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
Transformer (RND2RND)	46.25	16.57	29.82	57.56	24.25	35.53
PhoBERT2PhoBERT	50.4	19.88	32.49	60.37	29.12	39.44
mBERT2mBERT	52.82	20.57	31.55	59.67	27.36	36.73
mBART	55.21	25.69	37.33	59.81	28.28	38.71
mT5	55.27	27.63	38.30	58.05	26.76	37.38
BARTpho	57.16	31.18	40.89	61.14	30.31	40.15
$ViT5_{base\ 256\text{-}length}$	57.86	29.98	40.23	61.85	31.70	41.70
$ViT5_{base\ 1024\text{-}length}$	58.61	31.46	41.45	62.77	33.16	42.75
$ViT5_{large\ 1024\text{-}length}$	**60.22**	**33.12**	**43.08**	**63.37**	**34.24**	**43.55**

Hình 2.9 Kết quả ViT5 trên tác vụ tóm tắt trừu tượng văn bản

Dựa vào kết quả thể hiện bảng trên, có thể thấy mô hình ViT5 cho kết quả tốt hơn nhiều so với các mô hình khác, kể cả ở bản base hay bản large. Điều đó chứng tỏ được hiệu quả mà mô hình mang lại cho tác vụ tóm tắt văn bản trừu tượng này. Chính vì vậy, em quyết định lựa chọn mô hình ViT5 để tinh chỉnh cho tác vụ tóm tắt trừu tượng văn bản do kết quả mô hình mang lại cũng như tài nguyên yêu cầu (bản base chỉ có khoảng 310M tham số).

Chương 3

Phương pháp nghiên cứu

3.1 Xây dựng luồng hệ thống tóm tắt

3.1.1 Sơ đồ luồng hệ thống

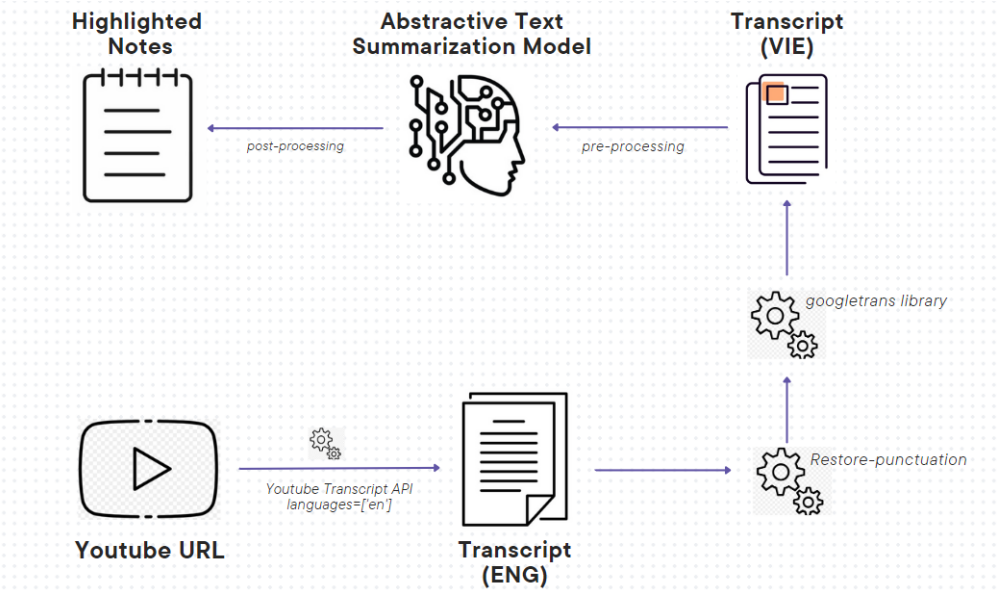
Sau quá trình thiết kế và chỉnh sửa hệ thống, luồng thực hiện của hệ thống tóm tắt cuối cùng bao gồm một số giai đoạn và các bước xử lý chính:

- Đầu vào: Youtube URL, đầu ra là các gạch đầu dòng tóm tắt ý chính bằng tiếng việt.
- Các transcript từ video tương ứng được lấy bằng Youtube Transcript API. Hiện nay, Youtube Transcript API đã hỗ trợ chuyển hóa hầu hết các ngôn ngữ trở thành phụ đề tiếng anh tự động với độ chính xác cao.
- Xử lý văn bản: Khôi phục lại các dấu và các quy tắc ngữ pháp. Điều này quan trọng bởi nó có thể gây ảnh hưởng đến hiệu suất của công cụ dịch thuật, và đầu ra của công cụ dịch thuật lại là

đầu vào của mô hình tóm tắt.

- Sử dụng công cụ dịch thuật để chuyển văn bản sang tiếng Việt. Để đảm bảo sự hoạt động của công cụ dịch thuật, một số bước xử lý văn bản cũng được thêm vào giai đoạn này.
- Sử dụng mô hình tóm tắt trừu tượng để tạo ra các đoạn văn bản rút gọn.
- Hậu xử lý: Khôi phục chữ viết hoa, tách văn bản thành gạch đầu dòng và các hậu xử lý khác.

Dưới đây là sơ đồ minh họa cho quá trình hoạt động của hệ thống tóm tắt nội dung truyền tải từ Youtube Video:



Hình 3.1 Luồng thực hiện hệ thống tóm tắt

Trong sơ đồ trên, phần thực hiện chủ yếu của đề tài là **xây dựng mô hình tóm tắt văn bản tiếng việt bằng việc tinh chỉnh mô hình ViT5**, sẽ được trình bày chi tiết ở mục 3.3. Các thành phần còn lại chủ

yếu là sử dụng API và kết hợp các kỹ thuật văn bản nhằm tạo ra đầu ra tốt nhất cho hệ thống.

3.1.2 Các thành phần hệ thống và công cụ sử dụng

API lấy transcript từ Youtube

Youtube Transcript API: API Python cho phép bạn lấy bản chép lời, phụ đề cho một video YouTube cụ thể. Hoạt động với cả phụ đề được tạo tự động và hỗ trợ dịch phụ đề. Cách hoạt động của API này đơn giản chỉ là truy cập vào kho dữ liệu của Youtube và lấy những transcript đã được tạo sẵn (từ các công cụ chuyển đổi giọng nói của Youtube) hoặc bản chép lời tự động do người dùng tải lên. Tuy nhiên trường hợp xuất hiện của bản chép lời tự động là không nhiều.

Thành phần xử lý transcript tiếng anh

Restore-function model: Sử dụng mô hình được đề cập trong repo "Deep Multilingual Punctuation Prediction" [13] để khôi phục lại những dấu câu đã bị mất trong quá trình lấy transcript từ Youtube Transcript API.

Googletrans API: Thư viện Python miễn phí và không giới hạn, triển khai dựa trên API Google Translate. Hạn chế về thiết kế chỉ cho phép tối đa 5000 ký tự cho đầu vào.

Mô hình tóm tắt văn bản tiếng việt

Cụ thể việc xây dựng mô hình sẽ được đề cập đến trong mục 3.3. Tuy nhiên, có một điểm hạn chế của mô hình này là nó được tạo ra

với đầu vào hữu hạn, chỉ khoảng 1024 tokens tương ứng với khoảng 700 từ. Chính vì vậy, em đề xuất một kỹ thuật để xử lý có thể giúp tóm tắt các văn bản dài: kỹ thuật chia đoạn có chồng lấn.

Kỹ thuật này được sử dụng nhằm chia đoạn văn bản đầu vào thành các đoạn nhỏ hơn tuy nhiên mỗi đoạn có một phần trùng nhau, qua đó giúp làm giảm sự mất mát thông tin trong quá trình chia đoạn. Sau đó ta áp dụng mô hình tóm tắt cho từng đoạn nhỏ và đưa ra kết quả.

Hậu xử lý

Phần hậu xử lý có 2 mục tiêu quan trọng: tạo ra văn bản đầu ra đủ tốt, có tính thẩm mỹ và loại bỏ các phần thông tin trùng lặp bị dư thừa. Những phần thông tin này có thể đến từ khả năng sinh văn bản của mô hình. Tuy nhiên hiện tại, phần hậu xử lý mới chỉ sử dụng nguyên lý đơn giản để loại bỏ trùng lặp: thực hiện loại bỏ các ý giống nhau hoàn toàn và loại bỏ các ý trùng khớp lớn về mặt văn bản (lớn hơn 15 ký tự). Những ý trùng khớp về mặt nội dung nhưng có phần văn bản khác nhau vẫn chưa được loại bỏ.

3.2 Xây dựng bộ dữ liệu

3.2.1 Gãn nhãn lại bộ dữ liệu Wikilingua

Sau khi tiến hành đánh giá mức độ phù hợp của các bộ dữ liệu, em đã lựa chọn bộ dữ liệu Wikilingua để tiến hành gán nhãn lại dữ liệu nhằm phục vụ cho đào tạo mô hình tóm tắt trừu tượng.

Nguyên nhân lựa chọn gán nhãn lại bộ dữ liệu Wikilingua:

nhằm tạo ra bộ dữ liệu mới có phần tóm tắt đầy đủ hơn so với bộ dữ liệu ban đầu. Bộ dữ liệu Wikilingua được chuyển từ tiếng anh sang tiếng Việt, chính vì vậy nó cũng phù hợp với đầu vào của mô hình tóm tắt trong hệ thống được trình bày ở mục. Bộ dữ liệu này cũng có kích thước vừa phải, phù hợp với tài nguyên hiện có.

Quá trình gán nhãn lại được thực hiện bán tự động với việc sử dụng Prompt kết hợp cùng Gemini AI [12]. Bằng cách này, những bản tóm tắt với chất lượng tốt, tương tự con người sẽ được tạo ra bởi mô hình ngôn ngữ lớn một cách hoàn toàn tự động, giúp giảm thiểu công sức và thời gian của con người.

Tuy nhiên cách làm này cũng tồn tại một nhược điểm. các mô hình ngôn ngữ lớn thường được huấn luyện trên các tập dữ liệu rất lớn, cùng với kích thước lớn, do đó chúng có khả năng tổng quát rất tốt, dẫn đến việc trong quá trình gán nhãn dữ liệu mới, đôi khi kết quả không giống như yêu cầu được mô tả trong prompt. Một số văn bản cũng bị loại bỏ theo tiêu chuẩn của Gemini, dẫn đến thất thoát dữ liệu.

Prompt được sử dụng để gán nhãn lại bộ dữ liệu:

> prompt = 'Được cung cấp một đoạn văn bản bên dưới, tóm tắt đoạn văn bản, giảm các từ dẫn và các từ nối, tóm tắt trong ít hơn 100 từ, viết trên một dòng. Loại bỏ các ký tự đặc biệt '

3.2.2 Tăng cường dữ liệu

Hai phương pháp chính được sử dụng để tăng cường dữ liệu cho việc huấn luyện mô hình là viết lại câu và dịch ngược.

Viết lại câu (Rephase)

Viết lại câu (Rephase) là một phương pháp tăng cường dữ liệu trong lĩnh vực xử lý ngôn ngữ tự nhiên, được thực hiện bằng cách diễn đạt lại đoạn văn bản gốc bằng các cách sử dụng từ ngữ, câu khác nhưng vẫn giữ nguyên ý nghĩa ban đầu. Phương pháp tăng cường dữ liệu này có thể được thực hiện bằng nhiều cách khác nhau như viết lại thủ công hay sử dụng các mô hình sinh văn bản. Trong đề tài này, chúng tôi sử dụng Gemini API kết hợp với Prompting để tạo ra văn bản mới với nội dung tương tự văn bản gốc. Chi tiết hơn về Gemini API và Prompting có thể tìm thấy tại [12].

Prompt được sử dụng để viết lại câu:

> prompt2 = 'Được cung cấp một đoạn văn bản dưới đây, hãy viết lại đoạn văn, nội dung giữ nguyên, thay đổi các từ và độ dài tương đương.'

Dịch ngược (Backtranslation)

Dịch ngược (Backtranslation) là một phương pháp tăng cường dữ liệu phổ biến trong lĩnh vực xử lý ngôn ngữ tự nhiên, được thực hiện bằng cách chuyển một văn bản đã được dịch sang ngôn ngữ đích (ngôn ngữ thứ hai) quay ngược trở lại về ngôn ngữ ban đầu. Trong mỗi lần dịch, một số từ ngữ sẽ bị thay đổi nhưng nhìn chung về mặt ý nghĩa sẽ không có thay đổi quá nhiều, điều này tạo ra sự phong phú về cách dùng trong dữ liệu, giúp tăng tính đa dạng của dữ liệu huấn luyện. Trong đề tài này, chúng tôi sử dụng ngôn ngữ đích là tiếng Anh nhằm phù hợp với đầu vào của mô hình tóm tắt trong hệ thống được trình bày ở 3.1.

Thông tin về bộ dữ liệu mới được trình bảy trong bảng dưới đây:

- Tổng số câu được gán nhãn lại: 9898 câu.
- Tổng số câu được tăng cường theo viết lại câu: 8217 câu.
- Tổng số câu được tăng cường theo dịch ngược: 5000 câu.

	Wikilingua (gốc)	New-Wikilingua (Nhãn mới)	New-Wikilingua-Aug (tăng cường)
train	13707	7918	18492
test	3916	1980	4653
độ dài trung bình nội dung	521	505	415
độ dài trung bình tóm tắt	44	45	45

Hình 3.2 Thông tin về dữ liệu mới

Mặc dù trong mục đích trình bày ở phần đầu mục này, mục tiêu gán nhãn lại dữ liệu là để tăng thêm thông tin ở phần tóm tắt. Tuy nhiên, theo như thông tin về 2 bộ dữ liệu mới được trình bày ở bảng trên, ta thấy độ dài trung bình phần tóm tắt hầu như không thay đổi, điều này vì một số nguyên nhân:

- Khi xây dựng bộ dữ liệu mới, em đã lược bỏ một số cặp văn bản có đầu vào quá dài. Vì mô hình được xây dựng ở độ dài 1024 tokens, nên những văn bản quá dài trên 1200 từ có thể được lược bỏ nhằm giảm thiểu trường hợp khó cho mô hình.

- Điều này dẫn đến những đoạn tóm tắt dài cũng bị lược bỏ, qua đó mặc dù đã tăng độ dài ở mỗi mẫu tóm tắt nhưng độ dài trung bình vẫn bằng so với ban đầu.

- Khả năng của mô hình Gemini rất tốt, dẫn đến việc khi viết lại câu theo cách khác, nó đã tự tổng hợp một phần văn bản thành

dạng ngắn hơn nên ở mục dữ liệu New-Wikilingua bản tăng cường, ta thấy chiều dài ở câu inputs trung bình bị giảm xuống rõ rệt.

3.3 Tinh chỉnh mô hình ViT5

3.3.1 Xử lý dự liệu đầu vào

Dữ liệu ban đầu được xử lý trở thành dạng Dataset của Huggingface [14], trở thành một mảng chứa các đối tượng từ điển, mỗi đối tượng từ điển sẽ có hai khóa là “inputs” và “labels”. Phần “inputs” chứa các đoạn văn bản gốc, còn phần “labels” chứa các đoạn tóm tắt tương ứng. Tiếp theo, áp dụng quá trình tiền xử lý cho dữ liệu trước khi đưa vào mô hình, bao gồm các bước:

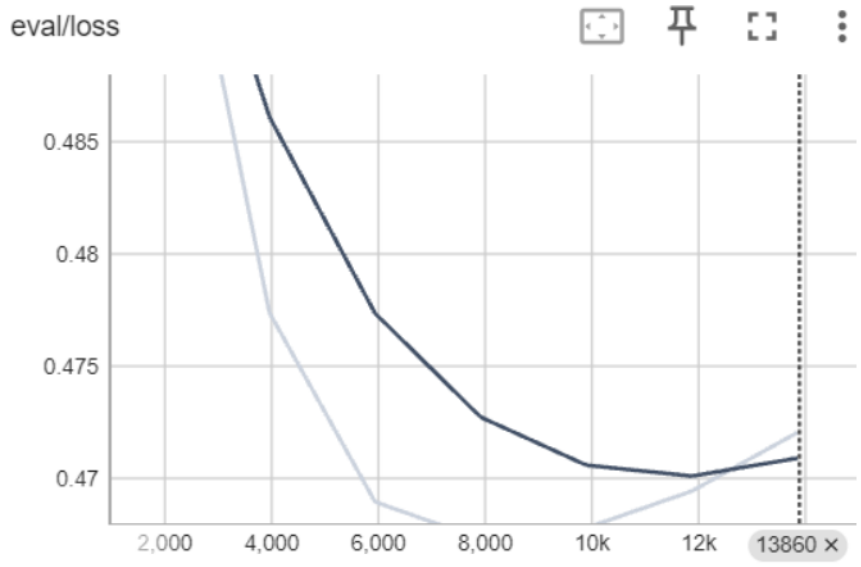
- Tiền xử lý loại bỏ ký tự: Loại bỏ các ký tự đặc biệt, ký tự thừa trong quá trình xử lý utf-8, chỉ giữ lại các dấu câu có tác dụng phân tách các câu như “.”, “!”, “?”, ";".
- Chuyển dữ liệu về chữ viết thường, nhằm giảm thiểu kích thước dữ liệu đầu vào.
- Tách từ và số hóa dữ liệu văn bản: Quá trình được thực hiện bằng Tokenizer đã huấn luyện được đội ngũ ViT5 công bố, bao gồm việc tách từ theo thuật toán SentencePiece [15] và số hóa các từ tương ứng với giá trị của nó trong bộ từ vựng của ViT5.
- Cắt hoặc chèn thêm (padding) dữ liệu để đảm bảo độ dài không vượt quá 1024 tokens cho mỗi mẫu dữ liệu đầu vào.

3.3.2 Mô hình với bộ dữ liệu New-Wikilingua

Bộ dữ liệu New-Wikiingua bao gồm 7918 cặp văn bản trong tập huấn luyện và 1980 văn bản trong tập kiểm chứng, sau khi tiến hành tiền xử lý dữ liệu theo các bước được trình bày trong 3.3.1, ta tiến hành huấn luyện mô hình. Sau 7 epoch đầu, kết quả thu được (để tăng tính trực quan, các giá trị ROUGE được nhân lên với 100):

- Loss: 0.4720
- ROUGE-1: 48.6293
- ROUGE-2: 25.6053
- ROUGE-L: 35.2967
- ROUGE-L-SUM: 37.4842

Đồ thị giá trị loss trên tập kiểm chứng:



Hình 3.3 Giá trị loss trên tập kiểm chứng của model chưa tăng cường

Nhận thấy giá trị loss đã có dấu hiệu đi ngang, cùng với đó, các giá trị ROUGE khác đều không có dấu hiệu tăng, em tiến hành

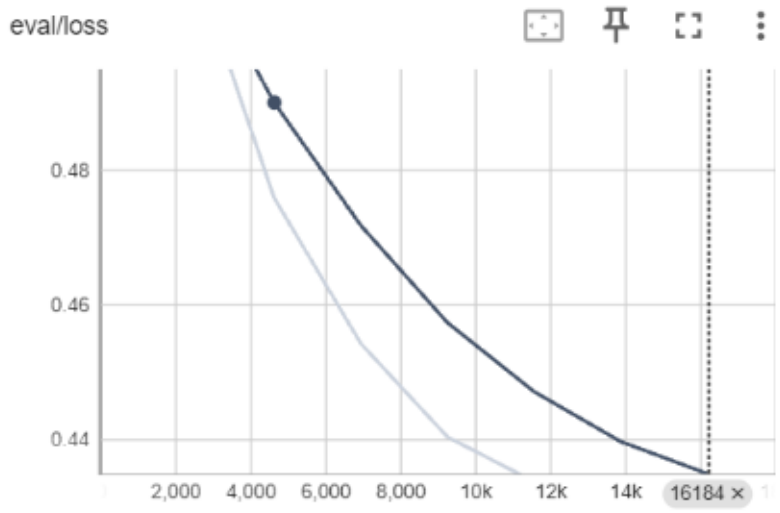
dừng sớm (early stopping) để tránh việc mô hình bị quá khít với dữ liệu huấn luyện (overfitting). Tiến hành thay đổi các thông số và tiếp tục tinh chỉnh mô hình. Nhưng kết quả không có cải tiến. Điều này dẫn đến việc cần sử dụng một số giải pháp nhằm tránh tình trạng quá khít với dữ liệu huấn luyện, một trong số đó là tăng cường dữ liệu, đây cũng là phương án được sử dụng trong việc huấn luyện mô hình trong luận văn này.

3.3.3 Mô hình với bộ dữ liệu New-Wikilingua tăng cường

Bộ dữ liệu NewWikilingua bản tăng cường bao gồm 18492 cặp văn bản trong tập huấn luyện và 4653 cặp văn bản trong tập kiểm chứng. Tiến hành huấn luyện mô hình, sau 7 epoch đầu, kết quả đạt được (để tăng tính trực quan, các giá trị ROUGE được nhân lên với 100):

- Loss: 0.4282
- ROUGE-1: 48.7749
- ROUGE-2: 26.3665
- ROUGE-L: 35.7765
- ROUGE-L-SUM: 38.0111

Đồ thị giá trị loss trên tập kiểm chứng:

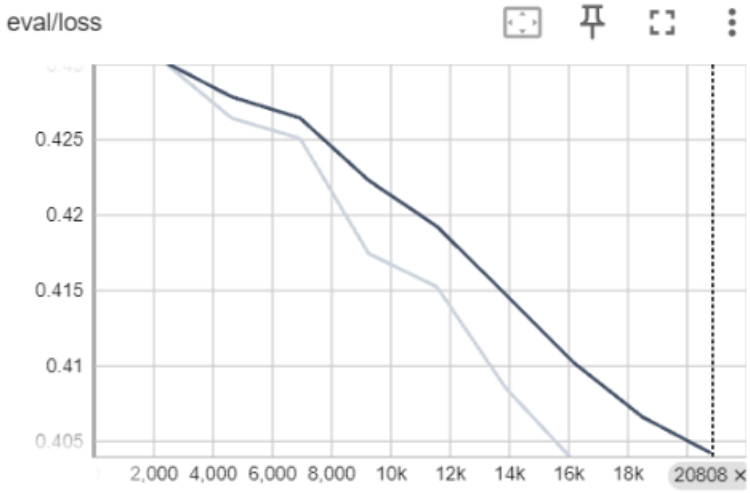


Hình 3.4 Giá trị loss trên tập kiểm chứng của model tăng cường đầu tiên

Kết quả đã cải thiện so với mô hình ở mục 3.3.2, giá trị loss đã giảm, các thông số cũng tăng lên, đặc biệt là ROUGE-L và ROUGE-L-SUM, điều này cho thấy chất lượng của các bản tóm tắt do mô hình tạo ra đã được cải thiện. Do giá trị loss vẫn đang có xu hướng giảm, tiếp tục tiến hành tinh chỉnh mô hình với tốc độ học được giảm đi, thêm 9 epoch, ta thu được kết quả (để tăng tính trực quan, các giá trị ROUGE được nhân lên với 100):

- Loss: 0.401387
- ROUGE-1: 49.352919
- ROUGE-2: 27.310599
- ROUGE-L: 36.524861
- ROUGE-L-SUM: 38.649183

Đồ thị giá trị loss trên tập kiểm chứng:



Hình 3.5 Giá trị loss trên tập kiểm chứng của model tăng cường đầu tiên

Dựa vào đồ thị, có thể thấy loss vẫn đang tiếp tục giảm, điều này chứng tỏ nếu tiếp tục huấn luyện mô hình tiếp, kết quả đạt được có thể được cải thiện, Tuy nhiên do hạn chế về mặt thời gian và tài nguyên, tạm thời nghiên cứu của em dừng lại ở kết quả này, và sẽ tiếp tục được cải thiện dần trong tương lai.

3.3.4 Đánh giá và nhận xét

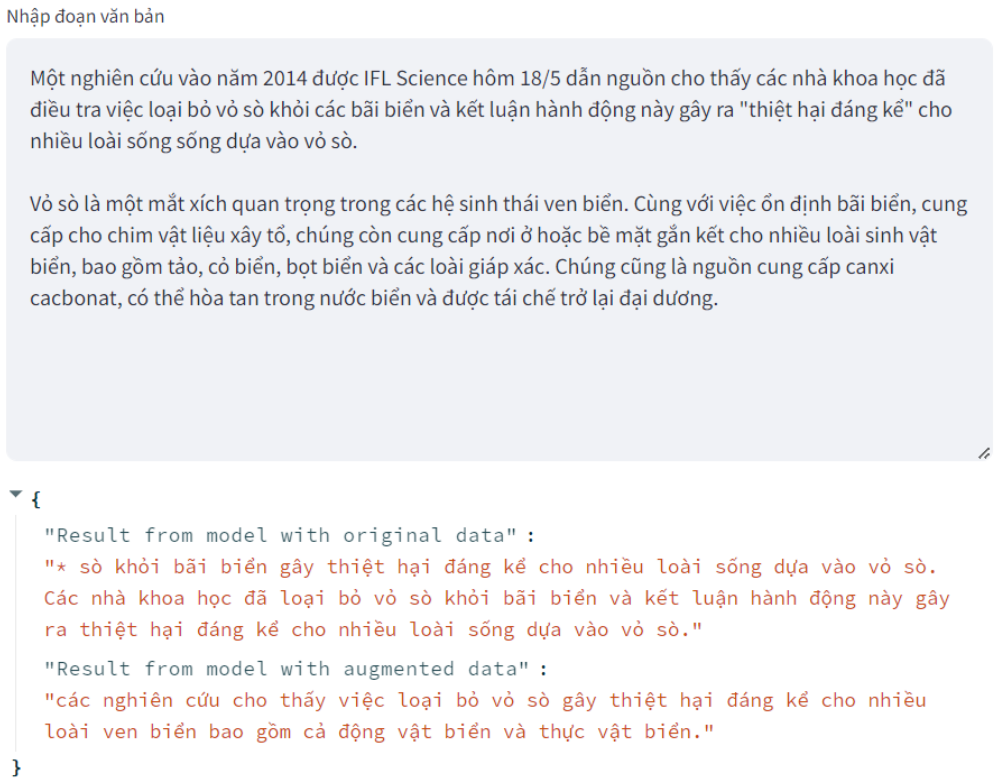
Sau khi tiến hành thử nghiệm và đánh giá, em nhận rút ra một số kinh nghiệm về 2 mô hình được trình bày bên trên:

- Cả hai mô hình đều đã có thể tạo ra những bản tóm tắt tiếng việt hoàn chỉnh, ít lỗi chính tả.
- Mô hình với dữ liệu tăng cường có khả năng tạo sinh ra các đoạn tóm tắt tốt hơn so với mô hình dữ liệu chưa tăng cường, cả về nội dung và cú pháp. Mô hình với dữ liệu chưa tăng cường vẫn gặp

phải trường hợp bị sinh ra ký tự "*".

- Với những đoạn văn quá ngắn hoặc mang tính liệt kê, mô hình với dữ liệu chưa tăng cường có xu hướng bị sinh ra các câu trùng lặp liên tiếp, điều này không xảy ra với mô hình có dữ liệu tăng cường.

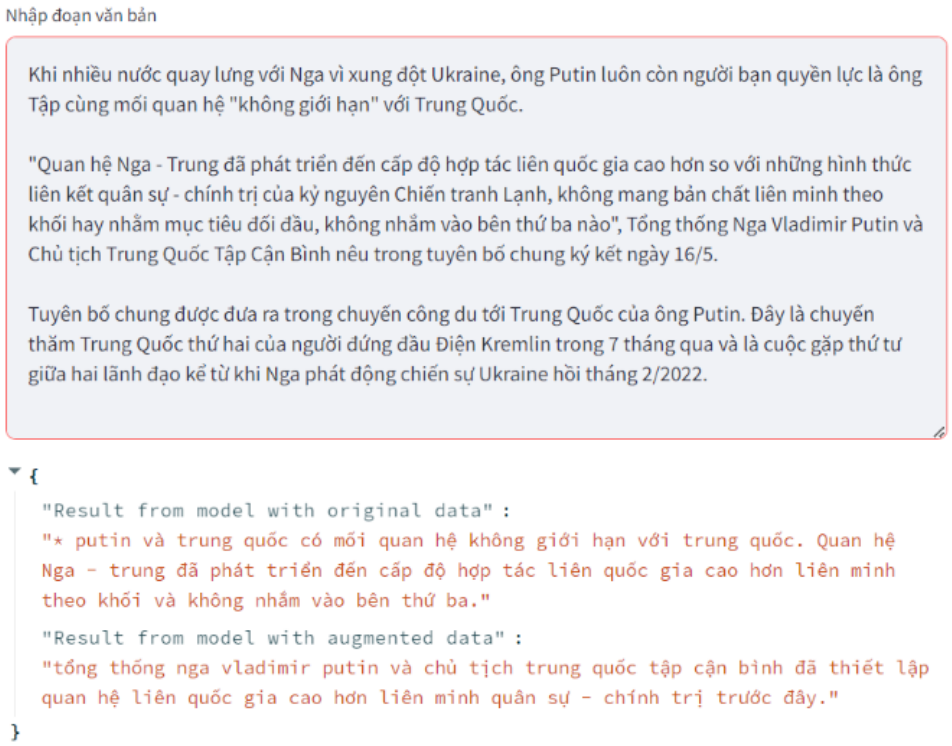
Điều này chứng tỏ mô hình có dữ liệu được tăng cường hoạt động tốt hơn so với mô hình ban đầu. Một số kết quả thu được từ mô hình tóm tắt, với đầu vào là các đoạn văn bản được lấy từ *vnexpress.net*:



Hình 3.6 Kết quả thử nghiệm các mô hình tóm tắt

Có thể thấy mô hình với dữ liệu chưa được tăng cường tạo ra ký tự "*" ở phần đầu kết quả. Một ví dụ khác cho thấy mô hình với

dữ liệu tăng cường hoạt động hiệu quả hơn so với mô hình chưa được tăng cường:



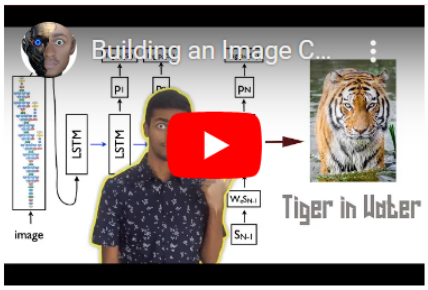
Hình 3.7 Kết quả thử nghiệm các mô hình tóm tắt

Chương trình thử nghiệm 2 mô hình được tạo bằng Streamlit và được mở hoàn toàn công khai trên nền tảng Spaces của Huggingface, thầy, cô cũng như bạn đọc có thể thử nghiệm tại:

`https://huggingface.co/spaces/minnehwg/demo-summarization-models`

3.4 Kết quả thử nghiệm hệ thống

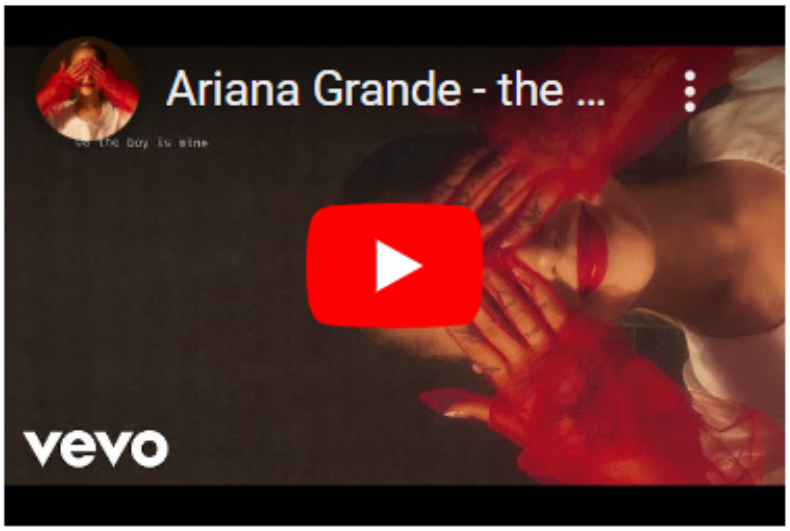
Nhằm tiến hành đánh giá kết quả của hệ thống tóm tắt, em đã tiến hành thử nghiệm hệ thống trên các video có độ dài khác nhau.



• Hiểu lưới thần kinh giúp giải quyết các vấn đề về chú thích hình ảnh bằng cách xác định các lớp hình ảnh và sử dụng các mạng thần kinh tích chập để giải quyết các vấn đề về chú t hích hình ảnh.
• Để đào tạo mạng lưới tái phát tái phát sử dụng thuật toán ptt cắt ngắn và tập trung chú ý vào các phần của hình ảnh trong khi chú ý vào các phần của hình ảnh trong khi chú ý vào các phần của hình ảnh trong khi chú ý vào các phần của hình ảnh trong khi chú ý vào các ph ần của hình ảnh trong khi tập trung vào các phần của hình ảnh trong khi tạo chú thích trực giác.
• Trình thu thập hình ảnh xác định độ lớn của hình ảnh dựa trên số lượng phần hình ảnh cần chú ý.
• Số lượng phần hình ảnh cần chú ý là 0 hoặc 1.
• Trình thu thập hình ảnh sử dụng chương trình rnn tham dự intel để tạo chú thích cho hình ảnh đầu vào bằng cách sử dụng chương trình rnn tham dự intel.

Hình 3.8 Độ dài 13 phút - Cách tạo ra các Image Captioner

Dựa vào kết quả, ta có thể thấy một số từ khóa quen thuộc có thể xuất hiện như lưới thần kinh (neural network), đào tạo mạng lưới. Tuy nhiên, do được dịch sang tiếng việt, một số khái niệm, thuật ngữ trở nên tối nghĩa và khó hiểu.



Không thể lấy được phụ đề

Hình 3.9 Trường hợp không thể lấy được phụ để

Trong trường hợp không thể lấy được phụ đề, hệ thống sẽ trả ra thông báo "không thể lấy phụ để".

Với video có độ dài lớn, phần transcript được chia nhỏ thành nhiều khối dẫn đến kết quả có nhiều gạch đầu dòng hơn, đôi khi gây mất tính thẩm mỹ cho kết quả cuối

• Nghiên cứu về máy biến áp transformer và các phương pháp tiếp cận tiên tiến trong xử lý ngôn ngữ và dịch máy có thể giúp giải quyết các vấn đề về ngôn ngữ và dịch máy.
• Mã hóa vị trí trong kiến trúc máy biến áp đòi hỏi sự nhất quán giữa các từ trong một đoạ n văn.
• Để tối ưu hóa mô hình hãy thêm thông tin về vị trí của mỗi từ trong một đoạn văn.
• Sử dụng công thức sin để xác định các từ có trong đoạn văn.
• Các bản nhúng đầu vào và mã hóa vị trí thực hiện xử lý dữ liệu trước khi chuyển đổi thàn h pt.
• Các bản sao này được sử dụng để tính điểm của các từ trong câu đầu vào.
• Các bản sao này được sử dụng để tính điểm của các từ trong câu đầu vào và kết quả sau đó được sử dụng để tính điểm của các từ trong câu đầu vào.
• Tính toán đầu vào bằng cách sử dụng truy vấn của hồi giáo và giá trị của chúng.
• Sau đó sử dụng truy vấn của hồi giáo để tính toán điểm số.
• Sau đó sử dụng ma trận trọng số để tính toán điểm số.
• Sau đó sử dụng hàm chú ý để tạo lớp tự ý cho từ đầu tiên.
• Sử dụng softmax để bình thường hóa dữ liệu bằng cách tổng hợp các giá trị có trọng số và sử dụng các hàm chú ý để tạo lớp chú ý song song hoặc đầu.

Hình 3.10 Độ dài 30 phút - cơ chế Attention và Transformers

Dựa vào kết quả từ hệ thống, ta có thể thấy một số từ khóa như tối ưu hóa mô hình, mã hóa vị trí (position embedding). Tuy nhiên do hạn chế của phần hậu xử lý, vẫn còn nhiều ý bị lặp lại, dẫn đến dư thừa thông tin không cần thiết.

KẾT LUẬN

Trong khóa luận này, em đã thực hiện việc nghiên cứu, xây dựng được một mô hình tóm tắt trừu tượng văn bản và ứng dụng nó cho hệ thống tóm tắt nội dung truyền tải từ Youtube Video, đạt được một số kết quả khả quan, các kết quả được trình bày trong Chương 3.

Tuy nhiên, do thời gian thực hiện cũng như sự giới hạn về kiến thức và tài nguyên, đề tài hiện tại còn tồn đọng một số vấn đề chưa được giải quyết:

- Phần hậu xử lý của hệ thống tóm tắt chưa thực sự tốt, dẫn đến việc kết quả đưa cho người xem chưa đạt được như kỳ vọng.
- Hệ thống không xử lý được những video chứa thông tin thuần chữ (không có âm thanh). Điều này đòi hỏi kết hợp các kỹ thuật trong thị giác máy tính và OCR, hiện nằm ngoài khả năng và phạm vi của luận văn.
- Khả năng hội tụ của mô hình vẫn có thể tiếp tục cải tiến, chưa đạt đến mức tốt nhất.
- Trong một số trường hợp, hệ thống có thể không xử lý được URL Video đầu vào do nhiều nguyên nhân khác nhau.

- Dữ liệu huấn luyện mô hình chưa thật sự trùng khớp với trường hợp sử dụng trong thực tế.

Đề xuất hướng phát triển: Trong thời gian tới, em sẽ tập trung vào việc cải thiện chất lượng đầu ra của hệ thống tóm tắt nhằm đưa đến cho người sử dụng kết quả tốt nhất. Cụ thể, em sẽ thực hiện các hướng phát triển:

- **Cải thiện phần hậu xử lý hệ thống:** Tập trung vào việc loại bỏ các lỗi cú pháp, đảm bảo tính nhất quán và mạch lạc trong văn bản tóm tắt, đồng thời làm cho văn bản dễ đọc và dễ hiểu hơn đối với người dùng cuối.
- **Cải thiện đầu ra của mô hình tóm tắt trừu tượng:** Tăng cường chất lượng văn bản được sinh ra bởi mô hình bằng việc tăng cường dữ liệu huấn luyện và số giờ huấn luyện, giúp mô hình có khả năng tổng quát hóa tốt hơn.
- **Xây dựng bộ dữ liệu chuyên biệt tóm tắt văn bản dịch tự động:** Thu thập và xây dựng một bộ dữ liệu mới có khả năng tóm tắt các văn bản được dịch tự động một cách tốt hơn.

Tài liệu tham khảo

[1] Dilawari, A. & Khan, M. U. G. (2019), "ASoVS: Abstractive Summarization of Video Sequences," *IEEE Access*, PP(99):1-1.

[2] Zhao, B., Gong, M., & Li, X. (2021), "AudioVisual Video Summarization", *IEEE Access*.

[3] Li, H., Ke, Q., Gong, M., & Drummond, T. (2022), "Progressive Video Summarization via Multimodal Self-supervised Learning", *arXiv preprint arXiv:2201.02494*.

[4] Lehal, G. & Gupta, V. (2010), "A Survey of Text Summarization Extractive Techniques", *Journal of Emerging Technologies in Web Intelligence*, 2(3).

[5] Lin, C.-Y. (2004), "ROUGE: A Package for Automatic Evaluation of Summaries", *In Text Summarization Branches Out*, pages 74–81, Barcelona, Spain.

[6] Jurafsky, D. & Martin, J. H. (2024), *Speech and Language Processing*, Chapter 3: n-gram language model.

[7] Raffel, C., et al. (2019), "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, *arXiv:1910.10683 [cs.LG]*

[8] Phan, L., Tran, H., Nguyen, H., & Trinh, T. H. (2022), "ViT5: Pretrained Text-to-Text Transformer for Vietnamese Language Generation", *arXiv:2205.06457 [cs.CL]*.

[9] Vaswani, A., et al. (2017), "Attention is All You Need", *arXiv:1706.03762 [cs.CL]*.

[10] Ladhak, F., Durmus, E., Cardie, C., & McKeown, K. (2020), "WikiLingua: A New Benchmark Dataset for Cross-Lingual Abstractive Summarization", *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4034–4048.

[11] To Quoc, H., Van Nguyen, K., Luu-Thuy Nguyen, N., & Gia-Tuan Nguyen, A. (2021), Monolingual versus multilingual bertology for Vietnamese extractive multi-document summarization, *Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation*

[12] Gemini Team Google. (2023), "Gemini: A Family of Highly Capable Multimodal Models", *arXiv:2312.11805 [cs.CL]*.

[13] Guhr, O., "Deep Multilingual Punctuation Prediction," *Retrieved from* `https://github.com/oliverguhr/deepmultilingualpunctuation`.

[14] HuggingFace., *Dataset Documentation*, *Retrieved from* `https://huggingface.co/docs/datasets/index`.

[15] Kudo, T. & Richardson, J. (2018), "SentencePiece: A simple and language-independent subword tokenizer and detokenizer for Neural Text Processing",*arXiv:1808.06226 [cs.CL]*.