TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ────────



**BÁO CÁO BÀI TẬP LỚN**

**XỬ LÝ NGÔN NGỮ TỰ NHIÊN**

**ĐỀ TÀI: Dịch máy Anh-Việt sử dụng transformer**

Giảng viên hướng dẫn: PSG.TS Nguyễn Thị Kim Anh

Nhóm sinh viên thực hiện:

|  |  |
| --- | --- |
| Trần Đức Thọ  Trịnh Văn Thắng  Vũ Long Vũ  Nguyễn Duy Minh  Nguyễn Đức Thiều Quang | 20183634  20183630  20183675  20183591  20183615 |
|  |  |

***Hà Nội 7 - 2022***

# **LỜI NÓI ĐẦU**

Lời đầu tiên, nhóm em xin được gửi lời cảm ơn chân thành và sâu sắc tới PGS.TS. Nguyễn Thị Kim Anh đã hỗ trợ em trong học phần Xử lý ngôn ngữ tự nhiên.Trong quá trình tìm hiểu và thực hiện đề tài, bài tập trong môn học, cô đã tận tình chỉ bảo, hướng dẫn cho em những kiến thức và tài liệu quan trong để em có thể hoàn thành được những mục tiêu mà ban đầu mình đã đặt ra.

Trong bài báo cáo này của em gồm 5 phần.

Phần 1: Giới thiệu bài toán

Phần 2: Dữ liệu, tiền xử lý dữ liệu

Phần 3: Mô hình transformer

Phần 4: Loss Function, Optimizer, Metric

Phần 5: Kết quả thực nghiệm

Một lần nữa, nhóm em xin cảm ơn cô và em hi vọng nhận được những lời nhận xét để có kết quả tốt hơn trong các công việc sau này.

**Bảng phân chia công việc**

|  |  |  |
| --- | --- | --- |
| Sinh viên thực hiện | MSSV | Nhiệm vụ |
| Trần Đức Thọ | 20183618 | - Đọc paper, xây dựng mô hình, visualize dữ liệu, xây dựng hàm loss, optimizer, metric đánh giá |
| Vũ Long Vũ | 20183476 | - Đọc paper, xử lý dữ liệu, xây dựng mô hình, code inference post process |
| Trịnh Văn Thắng | 20183473 | - Đọc paper, xử lý dữ liệu, xây dựng mô hình, code inference postprocess |
| Nguyễn Duy Minh | 20183595 | - Đọc paper, xây dựng mô hình, xử lý dữ liệu, xây dựng hàm loss, optimizer, metric đánh giá |
| Nguyễn Đức Thiều Quang | 20183615 | -Đọc paper  -Xây dựng mô hình, xây dựng hàm loss, optimizer, metric đánh giá |

Mục lục

[**LỜI NÓI ĐẦU** 2](#_Toc108941914)

[**1.** **Giới thiệu bài toán** 4](#_Toc108941915)

[**2.** **Dữ liệu** 6](#_Toc108941916)

[**2.1 Dữ liệu** 6](#_Toc108941917)

[**2.2 Tiền xử lý dữ liệu, tách từ** 7](#_Toc108941918)

[**3.** **Mô hình Transformer** 9](#_Toc108941919)

[**3.1.** **Encoder** 10](#_Toc108941920)

[**3.1.1.** **Positional Encoding** 10](#_Toc108941921)

[**3.1.2.** **Scale Dot-Product Attention** 10](#_Toc108941922)

[**3.1.3.** **Multi-Head Attention** 11](#_Toc108941923)

[**3.2.** **Decoder** 11](#_Toc108941924)

[**3.2.1.** **Masked Multi-Head Attention** 12](#_Toc108941925)

[**4.** **Loss Function, Optimizer, Metric** 13](#_Toc108941926)

[**5.** **Một số kết quả thực nghiệm** 14](#_Toc108941927)

[**6.** **Kết Luận** 15](#_Toc108941928)

[**7.** **Danh mục tài liệu tham khảo** 15](#_Toc108941929)

1. **Giới thiệu bài toán**

Xử lý ngôn ngữ tự nhiên là một lĩnh vực nghiên cứu nhằm giúp cho các hệ thống máy tính hiểu và xử lý được ngôn ngữ con người. Dịch máy là một trong những ứng dụng chính của xử lý ngôn ngữ tự nhiên. Mặc dù dịch máy đã được nghiên cứu và phát triển hơn 50 năm qua, song vẫn tồn tại nhiều vấn đề cần nghiên cứu. Ở Việt nam, dịch máy đã được nghiên cứu hơn 20 năm, nhưng các sản phẩm dịch máy hiện tại cho chất lượng dịch còn nhiều hạn chế. Hiện nay, dịch máy được phân chia thành một số phương pháp như: dịch máy trên cơ sở luật, dịch máy thống kê và dịch máy trên cơ sở ví dụ. Do những khác biệt về ngữ hệ, khác biệt về văn hóa và thiếu vắng nguồn tài nguyên, nên các phương pháp dịch máy hiện hữu thường gặp trở ngại khi áp dụng vào cặp ngôn ngữ Anh – Việt. Phương pháp dịch máy trên cơ sở luật cần phải xây dựng hệ thống luật cú pháp, ngữ nghĩa và phải có một từ điển khá đầy đủ thông tin cho các mục từ như ngữ nghĩa, ngữ dụng,... Để thực hiện phương pháp dịch máy dựa trên cơ sở luật, người ta cần nhiều thời gian và tiền bạc nhưng sản phẩm dịch vẫn không đạt độ chính xác như mong đợi. Dịch máy bằng phương pháp thống kê chưa có nhiều nghiên cứu áp dụng cho cặp Anh – Việt và kết quả nghiên cứu của N.P.Thai cũng hết sức khiêm tốn. Do sự khác biệt khá lớn về cấu trúc cú pháp của câu và nguồn ngữ liệu song ngữ chuẩn, nên ảnh hưởng đến chất lượng đối sánh từ Anh – Việt, mà kết quả của đối sánh từ lại quyết định đến chất lượng dịch. Hiện nay, đối sánh từ cho cặp Anh – Trung chỉ đạt được độ chính xác 50% - 60%. Kết quả này có thể tương tự khi áp dụng đối sánh từ cho cặp tiếng Anh – Việt .Phương pháp dịch máy trên cơ sở ví dụ truyền thống sử dụng các câu mẫu hay còn gọi là câu ví dụ. Các câu này được lưu trữ trên cơ sở dữ liệu với đầy đủ các thông tin như cây chú giải, các liên kết giữa các thành phần của hai câu thuộc hai ngôn ngữ. Phương pháp này cũng cần tập luật cú pháp của các câu ngôn ngữ nguồn để xây dựng cơ sở dữ liệu cho mẫu câu ví dụ. Sự khác biệt từ sẽ được xác định thông qua từ điển phân lớp, câu nhập sẽ được phân tích bằng tập luật cú pháp và xác định cặp cây cú pháp của câu nguồn và câu đích. Một tiếp cận khác với phương pháp dịch máy trên cơ sở ví dụ là xây dựng ngân hàng mẫu câu ví dụ. Câu nguồn chỉ cần so trùng từng phần với mẫu câu ví dụ bằng các giải thuật phù hợp (có sử dụng từ đồng nghĩa trong từ điển phân lớp). Phương pháp này cần khá nhiều thời gian tìm kiếm, xử lý thông tin để so trùng mẫu. Độ chính xác của phương pháp phụ thuộc vào số mẫu được lưu trữ nhiều hay ít. Trong thực tế thật khó để có thể lưu trữ đầy đủ các câu mẫu trên cơ sở dữ liệu vì số câu song ngữ trong kho ngữ liệu là nhiều vô kể. Với hai ngôn ngữ khác biệt như tiếng Anh và tiếng Việt với nguồn tài nguyên khá nghèo nàn, thì việc xây dựng kho câu mẫu sẽ càng cần nhiều thời gian và tiền bạc và là công việc của nhiều nhà ngôn ngữ học thực hiện trong nhiều năm.

1. **Dữ liệu**

**2.1 Dữ liệu**

Dữ liệu được lấy từ The Stanford NLP Group: IWSLT'15 English-Vietnamese data

Tập dữ liệu bao gồm tập **train** gồm 2 file .txt: train.en, train.vi, có tổng cộng 133 000 dòng câu tiếng Anh và 133 000 dòng câu tiếng Việt, mỗi dòng trong file train.en tương đương với mỗi dòng trong file train.vi, tập **test** bao gồm 4 file .txt: tst2012.en, tst2012.vi, tst2013.en, tst2013.vi, và 2 file **vocab** chứa 50 000 từ vựng: vocab.en, vocab.vi

Link nguồn chứa dữ liệu: [**https://nlp.stanford.edu/projects/nmt/**](https://nlp.stanford.edu/projects/nmt/)

Text

Description automatically generated with medium confidence

Text

Description automatically generated

Hình 1. *Dữ liệu raw(ENG-VIE)*

**2.2 Tiền xử lý dữ liệu, tách từ**

Data preprocessing: loại bỏ ký tự đặc biệt, khoảng trắng, thêm các ký tự <start>, <end> vào đầu và cuối mỗi câu, mã hóa ngữ nghĩa của từ (theo index của từ trong từ điển), padding, tách từ đối với Tiếng Việt(pyvi)

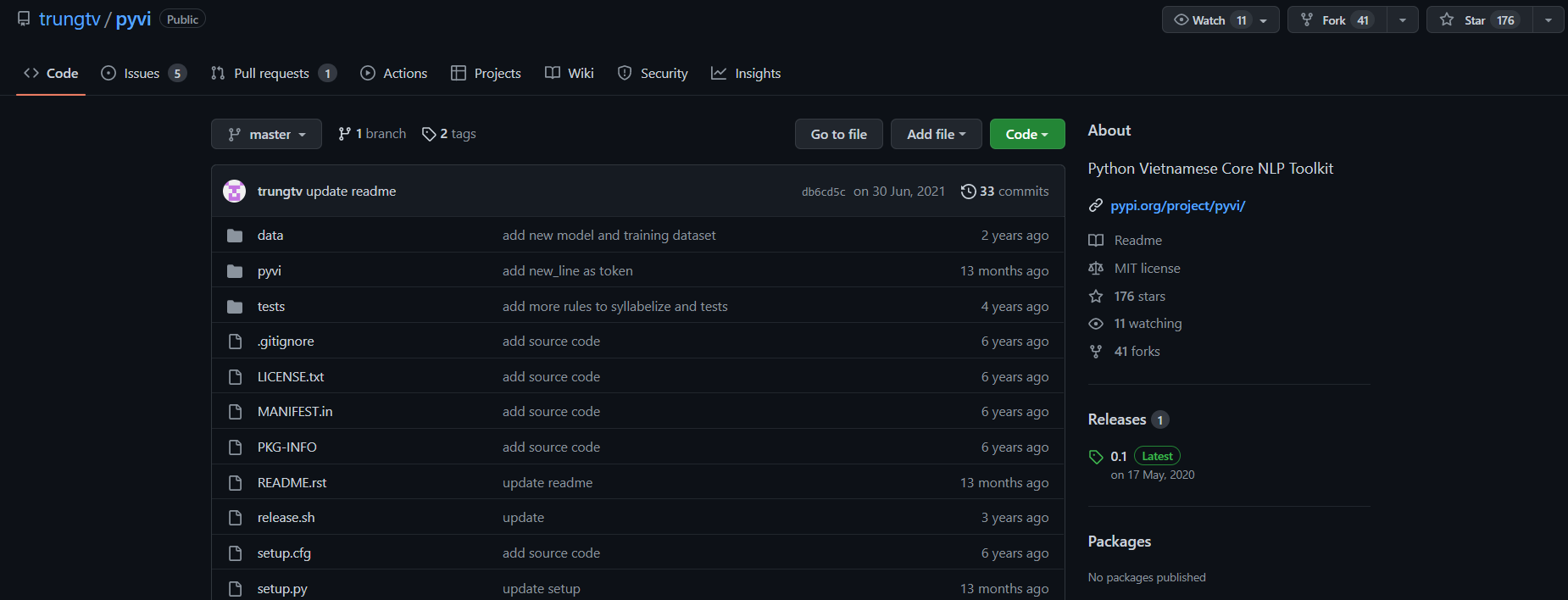
A picture containing text

Description automatically generated

Hình 2. *Dữ liệu sau khi xử lý*

Thư viện pyvi:

Thư viện chúng em dùng để tách các từ tiếng việt trong bài tập lớn lần này là pyvi- một mô hình pre-train do thầy Trần Việt Trung xây dựng



Hình 3. *Thư viện pyvi*

Tổng số câu trong tập train là 133317 câu Tiếng Anh tương ứng với 133317 câu Tiếng Việt

Chart, bar chart

Description automatically generated

Hình 4. *Biểu đồ phân phối 20 từ nhiều nhất tiếng Anh trong toàn bộ tập dữ liệu train*

Chart, bar chart

Description automatically generated

Hình 5. *Biểu đồ phân phối 20 từ nhiều nhất tiếng Việt trong toàn bộ tập dữ liệu train*

1. **Mô hình Transformer**

Diagram

Description automatically generated

Hình 6. *Mô hình Transformer*

* 1. **Encoder**

Encoder bao gồm: N=6 layers giống nhau mỗi layer bao gồm 2 phần:

* Multi-Headed Attention
* Position-wise fully connected feed-forward network

Nhiệm vụ của encoder:

* Học mối tương quan giữa từng từ trong câu với các từ còn lại (cơ chế self Attention)
* Bổ sung mối quan hệ này vào Embedding của từng từ đầu
  + 1. **Positional Encoding**

|  |  |
| --- | --- |
| - Do cơ chế Attention không ghi nhận thông tin vị trí của từ trong câu vì vậy khi dịch các từ sẽ bị đảo lộn -> cơ chế mã hóa thông tin vị trí: hàm sinusoid  Text, letter  Description automatically generated | png A picture containing text, watch, gauge  Description automatically generated |

* + 1. **Scale Dot-Product Attention**

|  |  |
| --- | --- |
| * Cơ chế Attention là 1 phép ánh xạ: (Q, K) -> V giúp các từ học mối quan hệ với các từ trong câu * MatMul (phép dot product): phép nhân vô hướng cho phép biết được độ tương đồng giữa 2 vector ở gần nhau hay không. * Phép nhân vô hướng có xu hướng tăng cường độ lớn => Hàm softmax sẽ trả về những giá trị rất nhỏ * Cơ chế scale: Var(QKT) = dk   Text  Description automatically generated | Diagram  Description automatically generated |
|  |  |

* + 1. **Multi-Head Attention**

|  |  |
| --- | --- |
| * Cơ chế Multi-Head Attention cho phép mô hình Attention ở từng không gian con trong câu (từng vị trị trí khác nhau trong câu) (h = 8)   Text  Description automatically generated | Diagram  Description automatically generated |
|  |  |
|  |  |

* 1. **Decoder**

Decoder bao gồm: N=6 layers giống nhau mỗi layer bao gồm 3 phần:

* Mask multi-head attention giữa các từ phía trước trong câu mục tiêu
* Multi-head Attention các từ này (V) với đầu ra của Encoder (Q, K)
* Point-wise fully connected FFW

Nhiệm vụ của decoder:

* Học mối quan hệ giữa các từ đứng trước nó, bổ sung mối quan hệ này vào embedding của từng từ (Add)
* Tiếp tục attention với encoder và bổ sung mối quan hệ vào embedding của từng từ
* Sử dụng các từ này để dự đoán từ tiếp theo
  + 1. **Masked Multi-Head Attention**

Look Ahead Mask

* Phép Masking cho phép cơ chế Attention: các từ sau chỉ attention với những từ đứng trước nó

Calendar

Description automatically generated

* + 1. **Position-wise feed forward network**

Mô hình sử dụng lại kiến trúc residual learning cải thiện việc training, tránh vanishing gradient

Ngoài ra do x đang lưu giữ thông tin về vị trí của từ trong câu nên layer này sẽ lưu trữ thông tin vị trí từ thấp lên cao

Layer Norm

Position-wise Feed Forward Networks

(d\_ff = 2048, d\_model = 512)

|  |  |
| --- | --- |
| Logo, company name  Description automatically generated | Residual Block Explained | Papers With Code |

1. **Loss Function, Optimizer, Metric**
   * Hàm loss được sử dụng là CrossEntropy, với optimizer được nhóm sử dụng là là Adam
   * Learning rate được setup lấy cảm hứng từ Cycle Learning Rate

Chart, line chart

Description automatically generated

- Tăng learning rate ở 4000 bước đầu (warmup\_steps = 4000) và giảm dần ở các bước tiếp theo

- Accuracy = TP =

Chart, line chart

Description automatically generated Chart, line chart

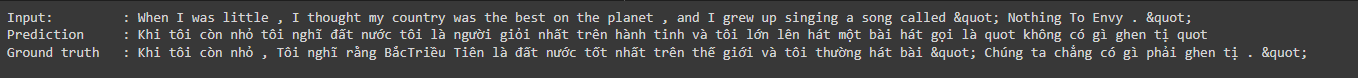
Description automatically generated

Giải thích về độ đo Accuarcy:

* Nhóm chúng em quyết định sử dụng Accuarcy: độ đo dựa trên số index (trong bước tiền xử lí) đúng với ground truth mà model prediction
* Phương pháp đánh giá này chỉ sử dụng Precision 1-gram thay vì trung bình n-gram như BLEU

1. **Một số kết quả thực nghiệm**

Dịch một câu ngẫu nhiên:



Dịch các câu trong tập test:

A picture containing graphical user interface

Description automatically generated

1. **Kết Luận**

* Nhóm chúng em đã hoàn thành cơ bản bài tập dịch máy
* Hiểu được cách tách từ, các khái niệm cơ bản trong xử lý ngôn ngữ tự nhiên

1. **Danh mục tài liệu tham khảo**

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, *Attention Is All You Need,* <https://arxiv.org/abs/1706.03762>

[2] Leslie N. Smith, *Cyclical Learning Rates for Training Neural Networks*, <https://arxiv.org/abs/1506.01186>