TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**NGUYỄN TƯỜNG AN - 52100380**

**LƯU MINH NHIỀU - 52100456**

**BÁO CÁO CUỐI KÌ**

**NHẬP MÔN HỌC MÁY**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**NGUYỄN TƯỜNG AN - 52100380**

**LƯU MINH NHIỀU - 52100456**

**BÁO CÁO CUỐI KÌ**

**NHẬP MÔN HỌC MÁY**

Người hướng dẫn

**GV. Lê Anh Cường**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

**LỜI CẢM ƠN**

Trong quá trình học môn "Nhập môn Học máy," chúng em may mắn được hướng dẫn và hỗ trợ nhiệt tình của Thầy Lê Anh Cường. Chúng em xin gửi lời cảm ơn sâu sắc và tri ân đến Thầy vì những kiến thức quý báu và kỹ năng quan trọng mà Thầy đã chia sẻ. Thầy Lê Anh Cường không chỉ là một người hướng dẫn giáo viên xuất sắc mà còn là người truyền đạt những kiến thức phức tạp một cách dễ hiểu và hấp dẫn. Những bài giảng, bài thực hành và hướng dẫn của Thầy đã giúp chúng em có cái nhìn tổng quan về lĩnh vực học máy và ứng dụng chúng vào thực tế. Chúng em không thể không đề cập đến sự kiên nhẫn và tận tâm của Thầy trong việc giải đáp mọi thắc mắc của chúng em. Nhờ sự hỗ trợ và sự giúp đỡ của Thầy, chúng em đã có cơ hội vận dụng những kiến thức mới nhất vào bài tập và dự án của mình. Cuối cùng, chúng tôi xin bày tỏ lòng biết ơn chân thành đến Thầy Lê Anh Cường, người đã giúp định hình cho chúng tôi một cách học máy và góp phần tạo nên hành trang kiến thức quan trọng cho hành trình sắp tới. Chúng tôi chân thành cảm ơn!

*TP. Hồ Chí Minh, ngày 16 tháng 12 năm 2023*

*Tác giả*

*(Ký tên và ghi rõ họ tên)*

*Lưu Minh Nhiều*

*Nguyễn Tường An*

**CÔNG TRÌNH ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của GV. Lê Anh Cường. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong Dự án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung Dự án của mình**. Trường Đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 16 tháng 12 năm 2023*

*Tác giả*

*(Ký tên và ghi rõ họ tên)*

*Lưu Minh Nhiều*

*Nguyễn Tường An*

**TÓM TẮT**

(Time New Romans – 13)

**ABSTRACT**

(Time New Romans – 13)

**MỤC LỤC**

[DANH MỤC HÌNH VẼ v](#_Toc143173056)

[DANH MỤC BẢNG BIỂU vii](#_Toc143173057)

[DANH MỤC CÁC CHỮ VIẾT TẮT viii](#_Toc143173058)

[CHƯƠNG 1. MỞ ĐẦU VÀ TỔNG QUAN ĐỀ TÀI 1](#_Toc143173059)

[1.1 Lý do chọn đề tài 1](#_Toc143173060)

[1.2 Mục tiêu thực hiện đề tài 1](#_Toc143173061)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 2](#_Toc143173062)

[2.1 Mạng neural hồi quy 2](#_Toc143173063)

[2.1.1 Recurrent Neural Network (RNN) 2](#_Toc143173064)

[2.1.2 Long Short-term Memory (LSTM) 2](#_Toc143173065)

[2.2 Mô hình Transformer 3](#_Toc143173066)

[2.2.1 Encoder và Decoder 3](#_Toc143173067)

[2.2.2 Attention 3](#_Toc143173068)

[CHƯƠNG 3. MÔ HÌNH ĐỀ XUẤT 4](#_Toc143173069)

[CHƯƠNG 4. THỰC NGHIỆM 4](#_Toc143173070)

[4.1 Dữ liệu thực nghiệm 4](#_Toc143173071)

[4.2 Cài đặt thực nghiệm 4](#_Toc143173072)

[CHƯƠNG 5. KẾT LUẬN 5](#_Toc143173073)

[5.1 Kết luận 5](#_Toc143173074)

[5.2 Hướng phát triển 5](#_Toc143173075)

[TÀI LIỆU THAM KHẢO 6](#_Toc143173076)

# DANH MỤC HÌNH VẼ

[Hình 2.1: Scaled Dot-Product Attention 3](#_Toc142677545)

# DANH MỤC BẢNG BIỂU

[Bảng 4.1: Thống kê kiểu thực thể trong tập VLSP 2016 4](#_Toc142677565)

# DANH MỤC CÁC CHỮ VIẾT TẮT

|  |  |
| --- | --- |
| BERT | Bidirectional Encoder Representations from Transformers |
| GEC | Grammatical Error Correction |
| MLM | Masked Language Model |
| NLP | Natural Language Processing |
| NSP | Next Sentence Prediction |

# CONTINUAL LEARNING

## Tổng quan

Continual learning (hay học tập liên tục) là ý tưởng cập nhật model (mô hình) của bạn khi có dữ liệu mới; điều này giúp mô hình của bạn theo kịp sự phân bổ dữ liệu hiện tại.

Sau khi mô hình của bạn được cập nhật, mô hình đó không thể được deploy (phát hành) một cách mù quáng vào sản xuất. Nó cần phải được thử nghiệm để đảm bảo rằng nó an toàn và tốt hơn so với mô hình hiện tại đang được sản xuất.

Continual learning thường bị hiểu sai:

- Continual learning KHÔNG đề cập đến một lớp thuật toán Machine Learning (ML) đặc biệt cho phép cập nhật mô hình dần dần khi có mọi điểm dữ liệu mới.

- Continual learning KHÔNG có nghĩa là bắt đầu công việc đào tạo lại mỗi khi có mẫu dữ liệu mới. Trên thực tế, điều này rất nguy hiểm vì nó khiến mạng lưới thần kinh dễ bị forgetting (lãng quên) một cách thảm họa.

## Mục tiêu của Continual Learning

Lý do cơ bản là để giúp mô hình của bạn theo kịp sự thay đổi phân phối dữ liệu. Có một số trường hợp sử dụng trong đó việc thích ứng nhanh chóng với việc thay đổi phân phối là rất quan trọng.

Ví dụ:

# CONTINUAL LEARNING

## Tổng quan

Continual Learning (Học Liên Tục) là một phương pháp trong lĩnh vực học máy mà mô hình được liên tục cập nhật và mở rộng để học từ dữ liệu mới mà không quên kiến thức đã học trước đó. Điều này là quan trọng trong các hệ thống thời gian thực hoặc trong các ngữ cảnh mà dữ liệu thay đổi theo thời gian. Mục tiêu là giữ cho mô hình linh hoạt, có khả năng học từ những thay đổi liên tục mà không làm giảm hiệu suất trên dữ liệu cũ.

Continual learning (hay học tập liên tục) là ý tưởng cập nhật model (mô hình) của bạn khi có dữ liệu mới; điều này giúp mô hình của bạn theo kịp sự phân bổ dữ liệu hiện tại.

Sau khi mô hình của bạn được cập nhật, mô hình đó không thể được deploy (phát hành) một cách mù quáng vào sản xuất. Nó cần phải được thử nghiệm để đảm bảo rằng nó an toàn và tốt hơn so với mô hình hiện tại đang được sản xuất.

Continual learning thường bị hiểu sai:

- Continual learning KHÔNG đề cập đến một lớp thuật toán Machine Learning (ML) đặc biệt cho phép cập nhật mô hình dần dần khi có mọi điểm dữ liệu mới.

- Continual learning KHÔNG có nghĩa là bắt đầu công việc đào tạo lại mỗi khi có mẫu dữ liệu mới. Trên thực tế, điều này rất nguy hiểm vì nó khiến mạng lưới thần kinh dễ bị forgetting (lãng quên) một cách thảm họa.

## Mục tiêu của Continual Learning

Lý do cơ bản là để giúp mô hình của bạn theo kịp sự thay đổi phân phối dữ liệu. Có một số trường hợp sử dụng trong đó việc thích ứng nhanh chóng với việc thay đổi phân phối là rất quan trọng.

Ví dụ: có thể có một buổi hòa nhạc ở một khu vực ngẫu nhiên vào Thứ Hai ngẫu nhiên và "mô hình Machine Learning định giá buổi hòa nhạc vào Thứ Hai" có thể không được trang bị tốt để xử lý nó.

## Stateless retraining và Stateful retraining:

### Stateless retraining:

-Khái niệm: Đào tạo lại mô hình của bạn từ đầu mỗi lần, sử dụng trọng số được khởi tạo ngẫu nhiên và dữ liệu mới.

- Đặc điểm:

+ Có thể có một số trùng lặp với dữ liệu đã được sử dụng để huấn luyện phiên bản mô hình trước đó.

+ Hầu hết các công ty bắt đầu thực hiện việc học tập liên tục bằng cách sử dụng stateless retraining.

+ Thỉnh thoảng bạn sẽ cần phải chạy stateless retraining với một lượng lớn dữ liệu để hiệu chỉnh lại mô hình.

### Stateful retraining ( fine-tuning, incremental learning):

- Khái niệm: Khởi tạo mô hình của bạn với các trọng số từ vòng huấn luyện trước và tiếp tục huấn luyện bằng cách sử dụng dữ liệu mới chưa thấy.

- Đặc điểm:

+ Cho phép mô hình của bạn cập nhật với lượng dữ liệu ít hơn đáng kể.

+ Cho phép mô hình của bạn hội tụ nhanh hơn và sử dụng ít năng lượng tính toán hơn.

+ Một số công ty đã báo cáo giảm 45% sức mạnh tính toán.

+ Về mặt lý thuyết, nó có thể tránh việc lưu trữ dữ liệu hoàn toàn sau khi dữ liệu đã được sử dụng để đào tạo (và để lại một khoảng thời gian an toàn). Về mặt lý thuyết, điều này giúp loại bỏ những lo ngại về quyền riêng tư dữ liệu.

+ Trên thực tế, hầu hết các công ty đều có thói quen theo dõi mọi thứ và không muốn vứt bỏ dữ liệu ngay cả khi nó không còn cần thiết nữa.

+ Sau khi cơ sở hạ tầng của bạn được thiết lập chính xác, việc thay đổi từ stateless retraining sang stateful retraining sẽ trở thành một nút nhấn.

+ Model iteration vs data iteration (Lặp lại mô hình và lặp lại dữ liệu): stateful retraining chủ yếu được sử dụng để kết hợp dữ liệu mới vào kiến trúc mô hình cố định và hiện có (tức là lặp lại dữ liệu). Nếu muốn thay đổi các tính năng hoặc kiến trúc của mô hình, bạn sẽ cần phải thực hiện bước đầu tiên là stateless retraining.

\* Đã có một số nghiên cứu về cách chuyển trọng số từ kiến trúc mô hình này sang kiến trúc mô hình mới (Net2Net knowledge transfer, model surgery). Có rất ít hoặc chưa có sự áp dụng những kỹ thuật này trong công nghiệp.

## Log and Wait:

### Khái niệm:

- Log (Ghi nhật ký): Ghi nhật ký trong quá trình học liên tục là quan trọng để theo dõi hiệu suất của mô hình theo thời gian và đối mặt với thách thức của việc tiếp tục học.

- Wait (Chờ đợi): Trong quá trình học liên tục, "wait" có thể ám chỉ thời gian chờ đợi giữa các lượt học mới để đảm bảo rằng mô hình đã học đủ từ dữ liệu mới.

### Mục đích sử dụng:

- Log:

+ Ghi lại thông tin về hiệu suất của mô hình sau mỗi lượt học mới.

+ Ghi nhật ký về các vấn đề như quá mức quên (catastrophic forgetting) hay hiệu ứng ngữ cảnh trước đó.

- Wait:

+ Ghi lại thông tin về hiệu suất của mô hình sau mỗi lượt học mới.

+ Ghi nhật ký về các vấn đề như quá mức quên (catastrophic forgetting) hay hiệu ứng ngữ cảnh trước đó.

Một số công ty lưu trữ các features đã được tính toán cho mọi mẫu dữ liệu để có thể sử dụng lại chúng cho mục đích đào tạo học tập liên tục và bằng cách đó tiết kiệm được kha khá tài nguyên. Việc này được gọi là log and wait.

## Thách thức của Continual Learning:

- Catastrophic Forgetting: Khi học một nhiệm vụ mới, mô hình có thể quên mất kiến thức đã học từ các nhiệm vụ trước đó.

- Fresh data access challenge(Thách thức truy cập dữ liệu mới): Tốc độ lưu trữ dữ liệu vào kho dữ liệu, Tốc độ ghi nhãn.

- Knowledge Transfer: Làm thế nào để chia sẻ và sử dụng lại kiến thức đã học từ một nhiệm vụ cho những nhiệm vụ mới.

- Representation Management: Làm thế nào để biểu diễn mô hình sao cho nó có thể linh hoạt và thích ứng được với nhiều loại dữ liệu.

- Evaluation Challenge (Thách thức đánh giá): việc học hỏi liên tục sẽ mở ra cơ hội cho các cuộc tấn công đối nghịch phối hợp nhằm đầu độc các mô hình.

- Data scaling challenge (Thách thức mở rộng quy mô dữ liệu): Việc mở rộng quy mô yêu cầu quyền truy cập vào số liệu thống kê dữ liệu toàn cầu như tối thiểu, tối đa, trung bình và phương sai. (min, max, average and variance.)

- Algorithm challenge (Thách thức thuật toán): Thách thức này xuất hiện khi bạn sử dụng một số loại thuật toán nhất định và muốn cập nhật chúng thật nhanh (ví dụ: mỗi giờ).

## Các giai đoạn của một Continual Learning:

Giai đoạn 1 - Manual, stateless retraining (đào tạo (lại) thủ công bằng stateless): Các mô hình chỉ được đào tạo lại khi đáp ứng hai điều kiện: (1) hiệu suất của mô hình đã giảm sút đến mức hiện tại nó gây hại nhiều hơn là có lợi, (2) nhóm của bạn có thời gian để cập nhật mô hình.

Giai đoạn 2 - Fixed schedule automated stateless retraining (Đào tạo (lại) bằng stateless retraining theo lịch trình cố định): Giai đoạn này thường xảy ra khi các mô hình chính của một miền đã được phát triển và do đó ưu tiên của bạn không còn là tạo các mô hình mới mà là duy trì và cải thiện các mô hình hiện có. Tần suất đào tạo lại ở giai đoạn này thường dựa trên: "gut feeling". (cảm giác “gut”). Các bước trong 1 quy trình đào tạo lại bao gồm:

- Pull data: Lấy dữ liệu.

- Downsample or upsample data if necessary: Xuống mẫu hoặc lấy mẫu dữ liệu nếu cần thiết.

- Extract features: Trích xuất các đặc trưng.

- Process and/or annotate labels to create training data: Xử lý và/hoặc chú thích (gán) nhãn để tạo dữ liệu đào tạo.

- Kick off the training process: Bắt đầu quá trình đào tạo.

- Evaluate the new model: Đánh giá mô hình mới.

- Deploy: Triển khai

Giai đoạn 3 - Fixed schedule automated stateful training (Đào tạo stateful theo lịch trình cố định): Để đạt được điều này, bạn cần phải cấu hình lại tập lệnh của mình và cách theo dõi dòng dữ liệu cũng như mô hình của bạn.

Giai đoạn 4 - Continual learning: Trong giai đoạn này, phần lịch trình cố định của các giai đoạn trước được thay thế bằng một số re-training trigger mechanism (trình kích hoạt đào tạo lại cơ chế). Các triggers có thể là:

- Time-based: Dựa trên thời gian

- Performance-based: Dựa trên hiệu suất

- Volume-based: Dựa trên khối lượng

- Drift-based: Dựa trên trôi dạt

* Tần suất cập nhật mô hình của bạn: trước tiên bạn cần hiểu và xác định bạn nhận được lợi ích gì khi cập nhật mô hình của mình với dữ liệu mới. Càng đạt được nhiều lợi ích thì càng phải đào tạo lại thường xuyên hơn.
* Đo lường giá trị của độ mới dữ liệu: Một cách để định lượng giá trị của dữ liệu mới là huấn luyện cùng một kiến trúc mô hình với dữ liệu từ 3 khoảng thời gian khác nhau, sau đó kiểm tra từng mô hình dựa trên dữ liệu được gắn nhãn hiện tại (xem hình ảnh).
* Nếu bạn phát hiện ra rằng việc để mô hình cũ trong 3 tháng sẽ gây ra sự khác biệt 10% về độ chính xác của dữ liệu thử nghiệm hiện tại và 10% là không thể chấp nhận được, thì bạn cần đào tạo lại sau chưa đầy 3 tháng.

## Khi nào thì nên cập nhật lại mô hình?

Hầu hết các kiến thức tôi đã đề cập đều là liên quan đến việc cập nhật mô hình với dữ liệu mới (tức là lặp lại dữ liệu). Tuy nhiên, trong thực tế, đôi khi bạn cũng có thể cần thay đổi kiến trúc mô hình của mình (tức là lặp lại mô hình).Dưới đây là một số gợi ý về khi bạn nên và không nên xem xét việc lặp lại mô hình:

- Nếu bạn liên tục giảm ngưỡng kích hoạt tái đào tạo dữ liệu và bạn không đạt được nhiều hiệu suất, bạn có thể nên tìm kiếm một mô hình tốt hơn (nếu doanh nghiệp của bạn cần).

- Nếu việc chuyển sang một kiến trúc mô hình lớn hơn và yêu cầu 100 lần công suất tính toán mang lại 1% cải thiện hiệu suất, nhưng giảm ngưỡng kích hoạt tái đào tạo xuống 3 giờ cũng mang lại tăng hiệu suất 1% với 1 lần công suất tính toán, hãy ưu tiên lặp lại dữ liệu hơn là lặp lại mô hình.

- Câu hỏi "khi nào nên lặp lại mô hình so với lặp lại dữ liệu" vẫn chưa có câu trả lời tốt cho tất cả các nhiệm vụ. Bạn sẽ cần thực hiện các thử nghiệm trên nhiệm vụ cụ thể của mình để tìm ra lúc nào nên thực hiện lựa chọn nào.

# TEST PRODUCTION

## Tổng quan:

Test Production (Kiểm Thử Sản Xuất) là một giai đoạn quan trọng trong vòng đời của một mô hình học máy, chuyển từ giai đoạn phát triển và đào tạo sang triển khai trong các kịch bản thực tế. Trong quá trình Kiểm Thử Sản Xuất, mục tiêu chính là đánh giá hiệu suất của mô hình khi đối mặt với dữ liệu thực tế, thường là dữ liệu động từ môi trường mà nó được thiết kế để hoạt động.

Để kiểm tra đầy đủ các mô hình của bạn trước khi phổ biến rộng rãi, bạn cần cả đánh giá ngoại tuyến trước khi triển khai (pre-deployment offline evaluations) và thử nghiệm trong sản xuất (testing in production). Chỉ đánh giá ngoại tuyến là không đủ.

Tốt nhất là các quy trình đánh giá này được tự động hóa và khởi động khi có bản cập nhật mô hình mới. Việc nâng cấp giai đoạn cần được xem xét tương tự như cách đánh giá CI/CD ( Continuous Integration (tích hợp liên tục)/Continuous Delivery (chuyển giao liên tục) ) trong công nghệ phần mềm.

## Phương pháp đánh giá:

### Model Offline Evaluation (Hay đánh giá trước khi triển khai):

Chúng ta đã quen với việc đánh giá một mô hình Machine Learning trong đó chúng ta huấn luyện một mô hình và đánh giá hiệu suất của nó trên một bộ dữ liệu xác thực chưa được nhìn thấy; điều này được thực hiện thông qua các số liệu (accuracy: độ chính xác), và biểu đồ đường cong ( precision-recall curve). Việc đánh giá 1 mô hình Machine learning trong sản xuất có thể dựa trên nhiều yếu tố tùy thuộc vào mục đích sử dụng của mô hình đó. Ta có 2 yếu tố để có thể đánh giá một mô hình trước khi triển khai:

- Baseline (Đường cơ sở): bạn cần một cái gì đó để so sánh. Chúng ta có nhiều loại baseline:

+ Đường cơ sở ngẫu nhiên (Random baseline)

+ Heuristic đơn giản (Simple heuristic)

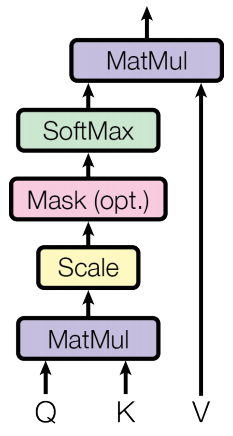
+ Đường cơ sở không có quy tắc (Zero rule baseline)

+ Cơ sở con người (Human baseline)

+ Cơ sở giải pháp hiện có (Existing solution baseline)

Các phương pháp đánh giá ngoài các số liệu ML tổng thể:

#### Scaled Dot-Product Attention



Hình 2.1: Scaled Dot-Product Attention

(Nguồn: (Vaswani et al., 2023))

Sẽ có 03 ma trận , và được sử dụng trong kỹ thuật attention này, tương ứng với các khái niệm query, key và value. Các dòng trong trong ma trận và sẽ có kích thước , các dòng trong ma trận sẽ có kích thước . Các ma trận này được tạo ra từ việc cho vector embedding đầu vào đi qua 03 bộ trọng số , và .

#### Multi-head Attention

# MÔ HÌNH ĐỀ XUẤT

…

# THỰC NGHIỆM

## Dữ liệu thực nghiệm

Dữ liệu trong VLSP-2016 đã được chia sẵn thành hai tập huấn luyện và kiểm thử. Thống kê chi tiết của các kiểu thực thể và các tập dữ liệu được mô tả trong Bảng 4.1**.**

Bảng 4.1: Thống kê kiểu thực thể trong tập VLSP 2016

|  |  |  |
| --- | --- | --- |
| **Label** | **Train** | **Test** |
| **LOC** | 1210 | 1377 |
| **ORG** | 7478 | 274 |
| **PER** | 6230 | 1294 |
| **MISC** | 250 | 47 |
| **Total** | 15168 | 2992 |

## Cài đặt thực nghiệm

…

# KẾT LUẬN

## Kết luận

## Hướng phát triển

# TÀI LIỆU THAM KHẢO

Tiếng Việt

…

Tiếng Anh

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-term Memory. *Neural Computation*, *9*, 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023). *Attention Is All You Need* (arXiv:1706.03762). arXiv. https://doi.org/10.48550/arXiv.1706.03762