ĐẠI HỌC QUỐC GIA TPHCM

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

Báo cáo về Mô hình ngôn ngữ lớn

Đề tài: Ứng dụng LLMs vào nhận diện văn bản trùng lặp

Sinh viên thực hiện:

Mai Đức Minh Huy

Ngày 26 tháng 8 năm 2024

Mục lục

1 Mở đầu

1.1 Bài toán

1.1.1 Mô tả

Nhận diện văn bản trùng lặp, một lĩnh vực nghiên cứu lâu đời trong Xử lý Ngôn ngữ Tự nhiên (NLP), đã chứng minh giá trị ứng dụng đáng kể trong nhiều bài toán thực tiễn, bao gồm tóm tắt văn bản, phát hiện đạo văn, và đánh giá hiệu suất của các mô hình học máy.
Báo cáo này nhằm cung cấp một cái nhìn tổng quan toàn diện về các phương pháp tiếp cận và giải pháp hiện có trong lĩnh vực nhận diện văn bản trùng lặp, còn được gọi là Độ tương đồng ngữ nghĩa văn bản (Semantic Textual Similarity).

1.1.2 Cách tiếp cận

Quá trình giải quyết bài toán này có thể được phân chia thành ba giai đoạn chính: Thứ nhất, tiền xử lý dữ liệu văn bản; thứ hai, nghiên cứu và áp dụng các phương pháp phân tích ngữ nghĩa (Semantic Analysis); và cuối cùng, triển khai các giải pháp đã đề xuất bằng cách sử dụng các công cụ và thư viện phần mềm hiện có. Mỗi giai đoạn đóng vai trò quan trọng trong việc xây dựng một hệ thống nhận diện văn bản trùng lặp hiệu quả và chính xác.

1.2 Các công trình nghiên cứu liên quan

1. Trùng lặp ngữ nghĩa của từng câu: GitHub CIS1010 Demo
2. Demo code của SBert: Sentence Transformer Tutorial
3. Nguồn đọc thêm về Semantic Textual Similarity: IEEE Explore

2 Phương pháp thực hiện

2.1 Chỉ số đo lường sự trùng lặp

2.1.1 Cosine Similarity

Cosine Similarity là một phương pháp phổ biến để đo lường độ tương đồng giữa hai vector trong không gian đa chiều. Trong ngữ cảnh của NLP, mỗi văn bản được biểu diễn như một vector trong không gian từ vựng. Công thức tính Cosine Similarity:

$$\cos\theta = \frac{\mathtt{A} \cdot \mathtt{B}}{\|\mathtt{A}\|\|\mathtt{B}\|}$$

Trong đó:

- A và B là hai vector đại diện cho hai văn bản
- A · B là tích vô hướng của A và B
- ‖A‖ & ‖B‖ là độ dài Euclidean của vector A và B

Ưu điểm của Cosine Similarity là không phụ thuộc vào độ dài của văn bản, cũng như hiệu quả tính toán sẽ cao. Ngược lại, phương pháp này khi so sánh sẽ không xét đến ngữ cảnh và trật tự của từ, từ đó không nắm bắt được ý nghĩa của văn bản.

2.1.2 TF-IDF (Term Frequency - Inverse Document Frequency)

TF (Term frequency): Tần suất xuất hiện của 1 từ trong 1 document.

$$TF(t,d) = \frac{\text{Số lần xuất hiện của 1 từ}}{\text{Tổng số từ}}$$

IDF (Invert Document Frequency): Dùng để đánh giá mức độ quan trọng của 1 từ trong văn bản. Khi tính tf, mức độ quan trọng của các từ là như nhau. Tuy nhiên trong văn bản thường xuất hiện nhiều từ không quan trọng xuất hiện với tần suất cao:

- Từ nối : và, hoặc, ...
- Giới từ: ở, trong, của, để, ...

- Từ chỉ định: ấy, đó, nhỉ

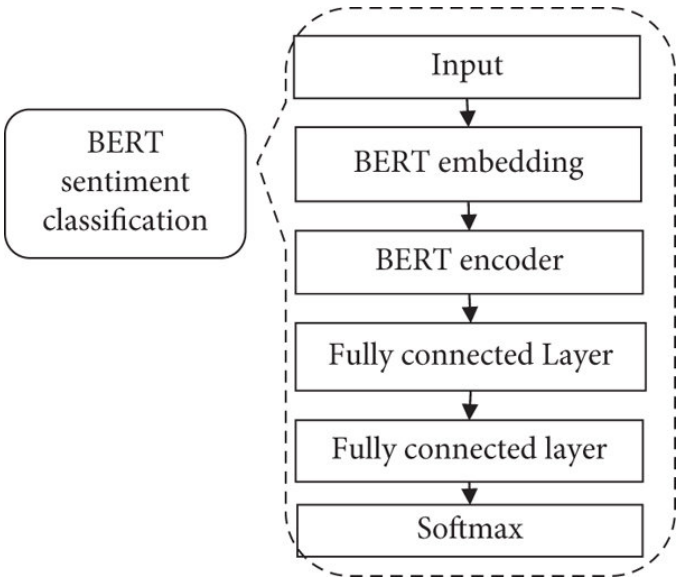
$$IDF(t, D) = \log\left(\frac{\text{Số văn bản trong tập D}}{\text{Số văn bản chứa từ t trong tập D}}\right)$$

Chỉ số TF-IDF được đo như sau: TF-IDF$(t, d, D) = TF(t, d) \cdot IDF(t, D)$
Những từ có tf-idf là những từ xuất hiện nhiều trong 1 văn bản này và xuất hiện ít trong văn bản khác. Việc này giúp lọc ra những từ phổ biến và giữ lại những từ có giá trị cao trong văn bản (keyword).

Vì vậy, để khai thác được tính thứ tự của văn bản, đồng thời kết hợp với các phương pháp đo lường như trên, chúng ta cần sử dụng các mô hình Encoder để có khả năng biến đổi văn bản thành vector (bao gồm cả ngữ nghĩa) - gọi là embeddings.

2.2 BERT & PhoBERT

BERT, hay còn được gọi là (Bidirectional Encoder Representations from Transformers), được giới thiệu vào năm 2018 bởi các nhà nghiên cứu tại Google AI Language. Đứng đầu nhóm nghiên cứu là Jacob Devlin, cùng với Ming-Wei Chang, Kenton Lee và Kristina Toutanova.



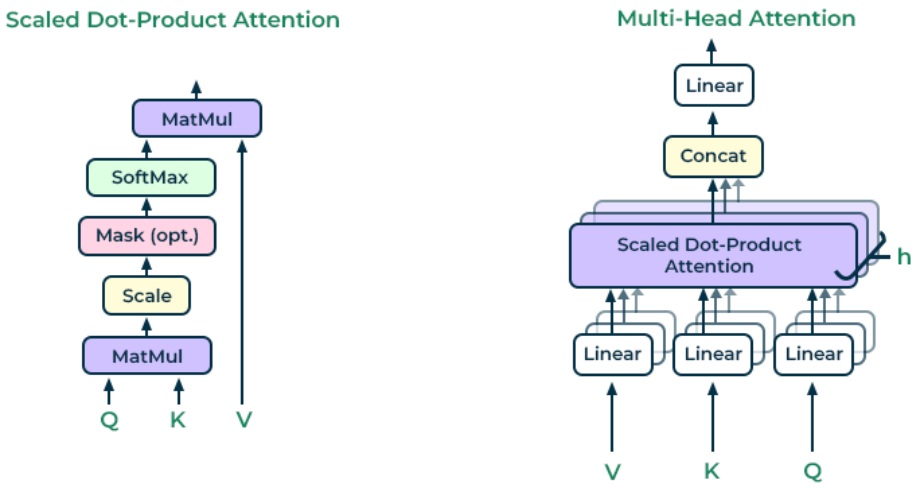
Hình 1: Sơ đồ BERT cho task sentiment classfication

BERT có thể được sử dụng cho nhiều tác vụ downstream khác nhau, trong đó có bài toán Semantic Textual Similarity của chúng ta. PhoBERT là một phiên bản đã được finetune trên tập dataset Việt ngữ nhằm tạo ra một mô hình phù hợp với ngôn ngữ Việt Nam. [GitHub - Mã nguồn mở huấn luyện mô hình] [Huggingface - Dùng để tải model]

Tuy nhiên, các mô hình LLMs BERT-based chỉ cho max input token là 512 (nôm na là đoạn văn bản đầu vào không được quá 512 chữ). Longformer là một kiến trúc mô hình khác được tạo ra để giải quyết điều đó.

2.3 Longformer

Longformer là một phiên bản của Transformer (kiến trúc LLM khai phá nền tảng NLP), được giới thiệu là có khả năng nhận nhiều hơn giới hạn của BERT-based (>4096 tokens đối với 512 tokens).



Hình 2: Sơ đồ của Longformer

Và với kết quả và khả năng nghiên cứu hiện tại, Longformer PhoBert-based là chìa khóa giải quyết vấn đề chính - vector embeddings của bài toán nhận diện văn bản trùng lặp đã đưa ra.

3 Hướng đi trong tương lai

3.1 Domain Adaptation

Domain Adaptation là một kỹ thuật trong huấn luyện AI, được sử dụng để cải thiện hiệu suất của mô hình ở các lĩnh vực cụ thể. BERT và các mô hình ngôn ngữ khác thường được huấn luyện trên đa dạng các loại kiến thức khác nhau, nên việc thực hiện kỹ thuật này có thể cải thiện hiệu suất đáng kể khi giải quyết bài toán Tương đồng Ngữ nghĩa Văn bản.

3.2 Tiền xử lý dữ liệu

Chúng ta có thể cải thiện hiệu suất bằng việc thu thập thêm nhiều dữ liệu, cũng như xử lý các dữ liệu sẵn có thật sạch. Lấy ví dụ là các văn bản hành chính thì chúng ta có thể loại bỏ các tiêu đề, đầu mục lặp lại, các chỉ số đánh dấu của văn bản để có thể thu thập được nội dung tốt nhất cho việc đánh giá.

Tài liệu

A Disclaimer

Vì đây là thực hiện áp dụng trên tập dataset tiếng Việt cũng như là vào các lĩnh vực giấy tờ hành chính/ chuyên ngành Điện Tổng công ty Điện lực miền Trung (EVNCPC), nên không thể đạt được các kết quả tốt như trên các tập dataset gốc từ tiếng Anh.
Các nguồn tham khảo (đang cập nhật):

- Blog của Phạm Đình Khánh về Computer Vision & NLP
- VIBLO - Consine Similarity & TF-IDF
- Bài báo gốc về kiến trúc BERT cũng như code mẫu
- Mô hình đã sử dụng