BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN TPHCM
KHOA CÔNG NGHỆ THÔNG TIN

----------o0o----------

KHAI THÁC VĂN BẢN VÀ ỨNG DỤNG

BÁO CÁO

Project 1

Giảng viên phụ trách: Lê Thanh Tùng
Nguyễn Trần Duy Minh

Sinh viên thực hiện: Cao Trung Kiên
Huỳnh Hoàng Gia Uy
Phạm Trần Minh Duy
Nguyễn Hữu Nam

MSSV: 21127326
20127385
21127257
20127568

Lớp: 21CNTThức

TP.Hồ Chí Minh, tháng 3 năm 2024

MỤC LỤC

MỤC LỤC 2
I. Tự đánh giá: 3
II. Nội dung chính: 4
1. EDA (Exploratory Data Analysis): 4
2. phoBERT + CNN + FLC: 5
3. DeBERTa 6
TÀI LIỆU THAM KHẢO 8

I. Tự đánh giá:

Công việc	Thành viên thực hiện	Tiến độ	Tự đánh giá về độ hoàn thiện(*)
Tìm hiểu về đề tài	Cả nhóm	100%	
EDA (Exploratory Data Analysis)	Huỳnh Hoàng Gia Uy	100%	
Thử nghiệm trên mô hình tự xây dựng: phoBERT + CNN + FLC	Phạm Trần Minh Duy Nguyễn Hữu Nam	100%	
Tìm hiểu về mô hình DeBERTa đa ngôn ngữ	Cao Trung Kiên	100%	

(*): Xanh lá: tốt, Vàng: vừa, Đỏ: Tệ

II. Nội dung chính:

1. EDA (Exploratory Data Analysis):

Exploratory Data Analysis (EDA) (phân tích khám phá dữ liệu) là một bước quan trọng trong quy trình phân tích dữ liệu. Trong giai đoạn này, chúng ta khám phá tập dữ liệu để hiểu rõ và diễn giải đặc điểm của nó trước khi tiến hành phân tích chi tiết.

Các công việc trong EDA bao gồm:

- **Nắm rõ dữ liệu**: Hiểu về kích thước, dạng dữ liệu, thống kê mô tả các biến, xác định mối quan hệ giữa các biến, phát hiện pattern và xu hướng, tìm ra bất thường và ngoại lai trong dữ liệu.
- **Đặt câu hỏi**: Đặt ra nhiều câu hỏi để tìm nhiều hướng khai thác và góc nhìn dữ liệu khác nhau. Điều này giúp xác định phần nào trong tập dữ liệu cần tập trung và xây dựng mô hình nào.
- **Sáng tạo và linh hoạt**: Đặt câu hỏi chất lượng là quá trình sáng tạo. Mỗi câu hỏi mới giúp bạn nhìn ra khía cạnh mới trong dữ liệu và tăng khả năng nhìn ra vấn đề.
- **Thực hiện trước khi phát triển giả thuyết hoặc sau khi xác định mục tiêu phân tích**.

EDA giúp đảm bảo chất lượng dữ liệu và chuẩn bị cho các bước phân tích tiếp theo. Nó là một phần quan trọng của phân tích dữ liệu và giúp chúng ta hiểu rõ hơn về dữ liệu để đưa ra quyết định kinh doanh.

Quá trình thực hiện:

- Tương đối thuận lợi

- Khó khăn: Gặp rắc rối trong việc tìm kiếm biểu đồ cho các phân tích, mô tả

2. phoBERT + CNN + FLC:

- **Khái quát mô hình:**
  - PhoBERT Embedding Layer: Sử dụng mô hình PhoBERT để trích xuất các embedding cho câu hỏi và văn bản liên quan.
  - CNN Layers:
    - Convolutional Layer: Sử dụng một lớp Conv1d với kernel size là 3 để trích xuất đặc trưng từ các embedding. Các đặc trưng này giúp mô hình hiểu cấu trúc và ý nghĩa của câu hỏi và văn bản.
    - Max Pooling Layer: Sử dụng lớp MaxPool1d để giảm kích thước của đặc trưng bằng cách lấy giá trị lớn nhất trong mỗi cửa sổ trượt.
  - Fully Connected Layer (FC): Lớp này nhận các đặc trưng đã được trích xuất từ CNN và thực hiện phân loại thành hai lớp: có thể trả lời hoặc không thể trả lời.
  - Hàm Kích hoạt: Sử dụng hàm kích hoạt ReLU sau mỗi lớp Convolutional để tạo tính phi tuyến tính.
  - Loss Function: Sử dụng hàm Cross-Entropy Loss để tính toán mất mát giữa các dự đoán và nhãn thực tế.
  - Optimizer: Sử dụng thuật toán tối ưu Adam để điều chỉnh các tham số của mô hình.
  - Training Loop: Mô hình được huấn luyện qua nhiều epochs với dữ liệu huấn luyện được chia thành các batch. Mỗi batch được truyền

qua mạng và sau đó mất mát được tính toán và cập nhật cho mỗi epoch.

- Hàm Evaluate: Đánh giá hiệu suất của mô hình trên tập dữ liệu validation bằng cách tính toán mất mát trung bình và độ chính xác.
- Tối ưu hóa: Tối ưu hóa mô hình thông qua việc cập nhật các tham số dựa trên mất mát tính toán từ tập dữ liệu huấn luyện.

- **Quá trình thực hiện:**
  - Thuận lợi:
  - Khó khăn:
    - Ít khi được làm việc trực tiếp với nhau
    - Nhiều kiến thức mới khiến tiếp thu khó khăn
    - Xử lý sai dữ liệu khiến kéo dài thời gian làm việc
    - Tài liệu tham khảo hạn chế

3. DeBERTa

DeBERTa (viết tắt của **Decoding-enhanced BERT with Disentangled Attention**) là một kiến trúc mô hình được đề xuất trong bài báo "DeBERTa: Decoding-enhanced BERT with Disentangled Attention" của các tác giả **Pengcheng He, Xiaodong Liu, Jianfeng Gao, Weizhu Chen**. Mô hình này dựa trên hai mô hình nổi tiếng khác là **BERT** của Google (phát hành vào năm 2018) và **RoBERTa** của Facebook (phát hành vào năm 2019). Được công bố vào năm 2020, nó đã thu hút sự chú ý trong cộng đồng xử lý ngôn ngữ tự nhiên (NLP) và đã được sử dụng trong nhiều nhiệm vụ khác nhau như MNLI, SQuAD v2.0 và RACE.

Các điểm nổi bật của DeBERTa bao gồm:

1. **Cơ chế chú ý không bị vướng víu**: Mỗi từ được biểu diễn bằng hai vector, một vector mã hóa nội dung và một vector mã hóa vị trí. Trọng số chú ý giữa các từ được tính toán bằng ma trận không bị vướng víu về nội dung và vị trí tương đối của chúng.

2. **Mô hình giải mã tăng cường**: DeBERTa sử dụng một giải mã tăng cường để dự đoán các từ bị che khuất trong quá trình huấn luyện mô hình.

So với **RoBERTa-Large**, một mô hình DeBERTa được huấn luyện trên một nửa dữ liệu huấn luyện thường có hiệu suất tốt hơn trên nhiều nhiệm vụ xử lý ngôn ngữ tự nhiên (NLP), bao gồm **MNLI**, **SQuAD v2.0** và **RACE**.

Bạn có thể tìm hiểu thêm về DeBERTa và tải mã nguồn mô hình đã được huấn luyện tại đây.

- **Kết quả: code chạy khá lâu**
- **Quá trình thực hiện:**
    - Thuận lợi:

      BERT là mô hình được phát triển với nhiều kiểu mô hình anh em khác nhau, nên có rất nhiều lựa chọn thử
    - Khó khăn:
        - Ít khi được làm việc trực tiếp với nhau
        - Nhiều kiến thức mới khiến tiếp thu khó khăn
        - Thời gian chạy khá lâu

TÀI LIỆU THAM KHẢO

1. https://huggingface.co/microsoft/mdeberta-v3-base
2.