**Đề Thi Giữa Kỳ**

**Xử Lý Ngôn Ngữ Tự Nhiên và Học Sâu**

**Thời gian làm bài: 180 phút**

Cho tập dữ liệu trên trang Vnexpress gồm có các thông tin như sau:

* Tiêu đề bài báo
* Chuyên mục của bài báo đó (Gồm có 8 loại: kinh doanh, khoa học, giải trí, thể thao, pháp luật, giáo dục, sức khỏe, du lịch)

Bạn cần xây dựng một mô hình học sâu để phân loại tiêu đề của bài báo vào chuyên mục tương ứng.

Ví dụ:

*Input: Liverpool vùi dập Manchester United bảy bàn không gỡ*

*Output: Thể thao*

Tải dữ liệu ở [đây](https://drive.google.com/file/d/18Zzgt59RmMBIR4uaoj4g7MxsmhTl8KqS/view?usp=sharing)

Yêu cầu cụ thể:

1. Tiền xử lý dữ liệu **(Tổng 3đ)**
   1. Xóa bỏ các dấu câu và ký tự đặc biệt. **(0.5đ)**
   2. Tách từ và loại bỏ các từ dừng (stop words). Tải danh sách các từ dừng ở [đây](https://github.com/stopwords/vietnamese-stopwords/blob/master/vietnamese-stopwords.txt). **(0.5đ)**
   3. Biến tất cả các số thành một token đặc biệt là <num> (Ví dụ: *19 ngày vừa lo vừa vui của cô gái Việt ở Ấn Độ -> <num> ngày vừa lo vừa vui của cô gái Việt ở Ấn Độ*). **(1đ)**
   4. Chia dữ liệu thành 2 tập, tập train và tập test, theo tỉ lệ 8:2 (8 phần train, 2 phần test). Yêu cầu số lượng nhãn cũng được chia theo tỉ lệ tương ứng (Ví dụ có 10 câu về chủ đề khoa học thì 8 câu train, 2 câu test, có 20 câu về chủ đề du lịch thì 16 câu train, 2 câu test) Gợi ý: Sử dụng thư viện scikit-learn. **(1đ)**
2. Xây dựng mô hình học sâu sử dụng Keras hoặc bất cứ thư viện nào bạn thích. Phần này được chia làm hai đề khác nhau, **sinh viên chỉ chọn một trong hai đề để làm**. (Nếu làm đề a thì cả bài sẽ được tối đa 9 điểm, làm đề b sẽ được điểm tối đa)
   1. Xây dựng mô hình Mạng nơ-ron truyền thẳng (Feed Forward Neural Network - FFNN). **(Tổng 4đ)**
      1. Sử dụng TF-IDF biến văn bản thành dạng vector để đưa vào mô hình FFNN. Coi mỗi tiêu đề là một tài liệu (document). Có thể sử dụng thư viện. **(1đ)**
      2. Xây dựng mô hình FFNN.

* Mô hình 2 lớp (gồm 1 lớp ẩn - hidden layer và 1 lớp đầu ra - output layer). **(1đ)**
* Lớp ẩn có kích thước là 128. Hàm kích hoạt là hàm ReLU. **(1đ)**
* Huấn luyện mô hình sử dụng thuật toán giảm gradient (SGD), hàm mất mát là cross entropy và huấn luyện trong 15 epochs. **(1đ)**
  1. Xây dựng mô hình Bộ nhớ dài ngắn hạn (Long Short-term Memory). **(Tổng 5đ)**
     1. Sử dụng word2vec để biến các từ thành dạng vector để đưa vào mô hình LSTM. Có thể sử dụng bất kỳ bộ word2vec nào bạn muốn hoặc bạn tự train. Hoặc tải word2vec ở [đây](https://public.vinai.io/word2vec_vi_syllables_100dims.zip) (File nặng 480MB, dung lượng giải nén 1.18GB, nên tải về và giải nén trước để tránh mất thời gian). Lưu ý: các từ trong file word2vec này không được tách từ mà để dưới dạng âm tiết (không có từ “vô địch” mà chỉ có vector cho từ “vô” và từ “địch”). **(2đ)**
     2. Xây dựng mô hình LSTM.
* Mô hình 2 lớp (gồm 1 lớp LSTM và 1 lớp đầu ra - output layer). **(1đ)**
* Lớp LSTM có kích thước là 64. **(1đ)**
* Train mô hình sử dụng thuật toán giảm gradient (SGD), hàm mất mát là cross entropy và huấn luyện trong 15 epochs. **(1đ)**

1. Đánh giá độ chính xác (accuracy) của mô hình trên tập dữ liệu test. **(1đ)**
2. Chạy thử mô hình. Viết một đoạn code cho phép nhập tiêu đề vào từ bàn phím, sau khi nhập xong, ấn Enter, chương trình sẽ đưa ra kết quả là chuyên mục của tiêu đề đó. **(0.5đ)**
3. Ghi rõ version của Python cùng với version của các thư viện cần có để chạy chương trình. **(0.5đ)**

HẾT