Global Vector

Global vector là một trong những phương pháp mới để xây dựng vector, nó được xây dựng dựa trên Co-occurrence Matrix. GloVe có bản chất là xác suất, ý tưởng xây dựng phương pháp này đến từ tỉ số:

(1)

Trong đó là xác suất xuất hiện của từ trong ngữ cảnh của từ , tương tự với .

Công thức của :

Trong đó

: số lần xuất hiện của từ trong ngữ cảnh của từ .

: số lần xuất hiện của từ trong ngữ cảnh của toàn bộ các từ còn lại ngoại trừ .

(Các giá trị này được lấy từ Co-occurrence Matrix)

Ý tưởng của GloVe: độ tương tự ngữ nghĩa hai từ có thể xác định thông qua độ tương tự ngữ nghĩa giữa từ với mỗi từ những từ có tính xác định ngữ nghĩa tốt chính là những từ làm cho (1)1 hoặc xấp xỉ bằng 0.

GloVE khởi đầu bằng việc tìm một hàm sao cho nó ánh xạ từ các vector từ trong vùng không gian sang một giá trị tỉ lệ với (1). Sau nhiều bước đơn giản hóa với tối ưu, ta xét hàm minimum cost như sau:

Trong đó:

, : các vector từ

: các bias tương ứng

: mục tương ứng với cặp từ trong Co-occurrence Matrix.

Hàm được gọi là weighting function, được thêm vào để giảm bớt sự ảnh hưởng của các cặp từ xuất hiện quá thường xuyên, hàm thỏa mãn 3 tính chất:

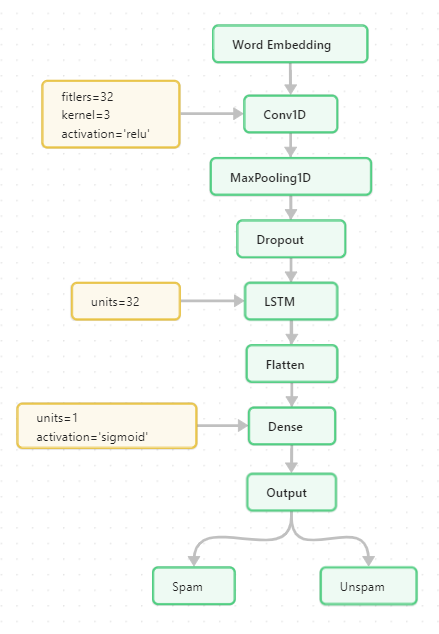
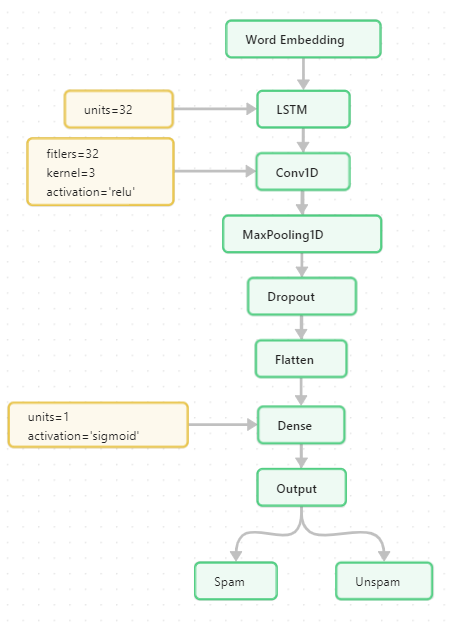
* Có giới hạn tại 0
* Là hàm không giảm
* Có giá trị nhỏ khi rất lớn

Việc thực hiện minimun cost để tìm ra các vector từ bằng nhiều cách trong đó tiêu chuẩn nhất là sử dụng Gradient Descent.

## I. Học sâu

Trong bài nghiên cứu, chúng em sử dụng mô hình dựa trên học sâu để phát hiện nhận xét ‘spam’ từ các nhận xét trong một video trên ứng dụng Youtube. Để triển khai mô hình, chúng em thử nghiệm 2 kiến trúc học sâu khác nhau dựa trên việc kết hợp giữa LSTM (Long short-term memory) và CNN (Convolutional Neural Network). Hai mô hình đã mang lại cho chúng em kết quả tốt về hiệu suất.

Trong hình 1 và hình 2, chúng em mô phỏng kiến trúc của 2 mô hình CNN-LSTM và LSTM-CNN.



Hình 2: CNN-LSTM

Hình 1: LSTM-CNN

1. Convolution 1D

Convolution 1D

Lợi ích chính của lớp Convolution là để trích xuất các đặc điểm quan trọng từ dữ liệu văn bản. Điều này được thực hiện bằng cách áp dụng phép tích chập lên các vector từ được tạo ra bởi lớp Word Embedding. Với là vector từ có chiều của từ thứ trong nhận xét. Hãy cho là đầu vào của tin nhắn, với là độ dài của tin nhắn. Đối với mỗi vị trí trong nhận xét, xem xét một vector cửa sổ với vector từ liên tiếp, được biểu diễn như sau:

Một phép tích chập bao gồm một bộ lọc được áp dụng lên cửa sổ để tạo ra một bản đồ đặc điểm mới Mỗi phần tử đặc điểm cho vector cửa sổ được tính toán như sau:

Trong đó là tích vô hướng từng phần tử, là một thuật ngữ độ lệch, và là một hàm phi tuyến. Trong trường hợp của chúng em, chúng em đã sử dụng đơn vị tuyến tính Đơn vị Đơn vị (ReLU) như một hàm phi tuyến. Nó được định nghĩa như sau:

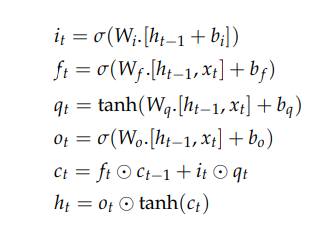
Hàm kích hoạt ReLU trả lại nếu giá trị là dương, nếu không trả lại 0. Đối với cấu hình của lớp tích chập, chúng em đã sử dụng một tích chập một chiều đi kèm với một cửa sổ bộ lọc Thuật toán 2 mô tả công việc chi tiết của thuật toán CNN.

2. Max Pooling

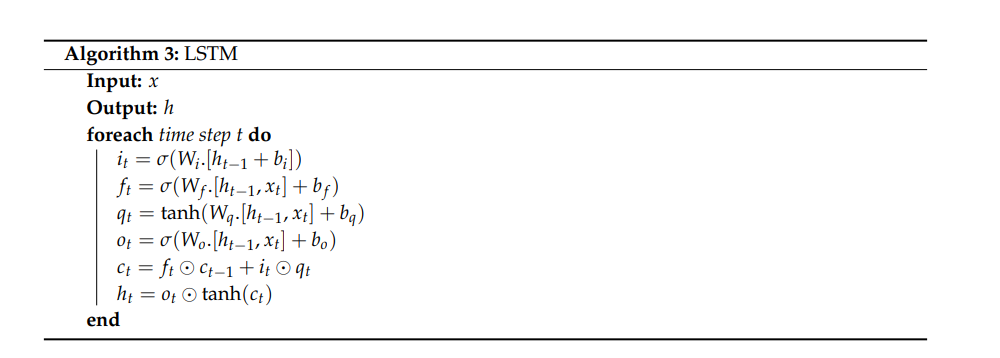
Các ‘feature map’ được tạo ra bởi phép tích chập được đặc trưng bởi một biểu diễn vector cấp cao. Để giảm biểu diễn này, chúng em đã thêm một lớp MaxPooling sau lớp Convolution để giúp chọn lựa chỉ thông tin quan trọng bằng cách loại bỏ thông tin kích hoạt yếu. Điều này hữu ích để tránh việc quá mức phù hợp do văn bản nhiễu.

3. Long short-term memory

LSTM (Long short-term memory) là một loại kiến trúc RNN (Recurrent Neural Network) có khả năng học các phụ thuộc dài hạn. Kiến trúc của LSTM bao gồm một loạt các đơn vị lặp lại cho mỗi bước thời gian. Mỗi đơn vị, tại một bước thời gian t, bao gồm một ô nhớ (phần bộ nhớ của LSTM) và ba cổng để điều chỉnh luồng thông tin bên trong đơn vị LSTM: một cổng đầu vào , một cổng đầu ra và một cổng quên . Những cổng này cùng nhau quyết định cách cập nhật ô nhớ hiện tại và trạng thái ẩn hiện tại . Các hàm chuyển đổi giữa các đơn vị LSTM được định nghĩa như sau:



Ở đây, là vector đầu vào của đơn vị LSTM, *σ* là hàm **sigmoid**, **tanh** biểu thị hàm kích hoạt và ⊙ biểu thị phép nhân Hadamard, và *W* và *b* là các ma trận trọng số và vector độ lệch cần được học trong quá trình huấn luyện. Trong kiến trúc của mô hình của chúng tôi, chúng tôi sử dụng một lớp LSTM duy nhất. Lớp này chứa 100 đơn vị LSTM sử dụng Dropout bằng 0.3 như một tham số điều chuẩn để ngăn mô hình quá mức. Thuật toán 3 mô tả quá trình hoạt động của LSTM:

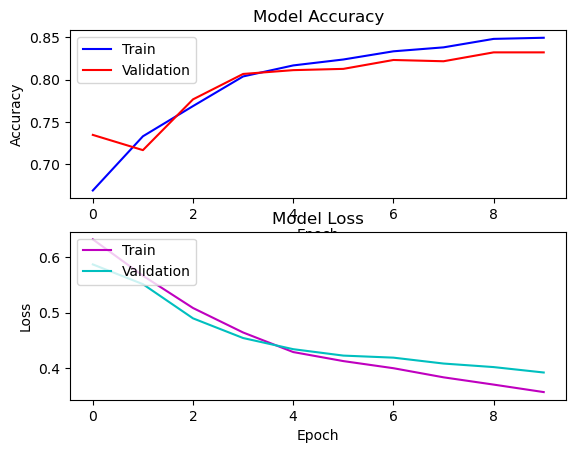
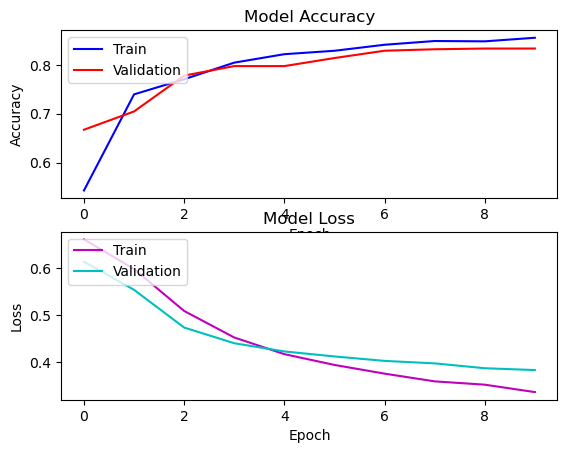


4. Dense

Lớp Dense là lớp cuối cùng của mô hình của chúng em. Nó cũng được gọi là lớp kết nối đầy đủ, và nó được sử dụng để phân loại các nhận xét văn bản dựa trên đầu ra. Vì mô hình phân loại của chúng em là nhị phân, chúng tôi đã sử dụng một lớp Dense với một neuron duy nhất và một hàm kích hoạt sigmoid để đưa ra dự đoán là 0 hoặc 1 cho hai lớp (Không phải là ‘spam’ và ‘spam’). Hàm sigmoid là một hàm logistic trả về giá trị giữa 0 và 1, như được định nghĩa bởi công thức

## II. Nhận xét

### Quá trình huấn luyện

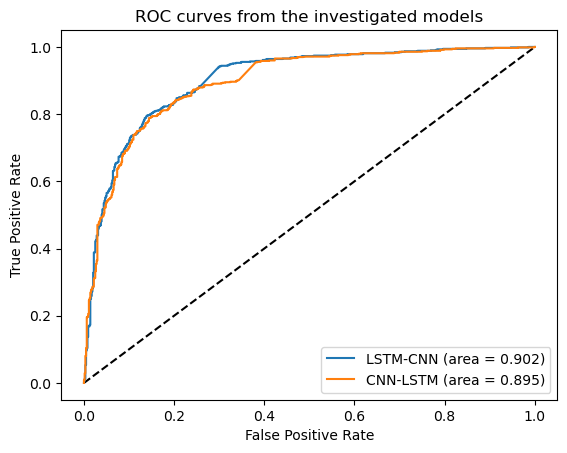


CNN-LSTM

LSTM-CNN

Qua kết quả từ 2 mô hai mô hình LSTM-CNN và CNN-LSTM sử dụng dữ liệu được biểu diễn bằng GloVe, là tích cực. LSTM-CNN và CNN-LSTM thể hiện hiệu suất xấp xỉ nhau, tính chính xác cao.. Quá trình Embedding với thuật toán GloVe thể hiện tốt qua độ chính xác của 2 mô hình trên thể hiện tính ứng dụng cao, xử lí tốt các từ mới hoặc hiếm gặp trong tập dữ liệu, điều này dẫn đến việc mất mát thông tin khi từ mới được kiểm soát tốt.

Đối với tập Test



Bằng cách kết hợp CNN và LSTM, chúng em có thể tận dụng cả khả năng của CNN trong việc nhận diện các mẫu cục bộ, và khả năng của LSTM trong việc sắp xếp văn bản. Tuy nhiên, thứ tự của các lớp trong các mô hình của chúng em sẽ đóng vai trò quan trọng trong việc họ thực hiện tốt đến đâu. Chúng em tin rằng sự khác biệt **delta** giữa các mô hình của chúng em không phải là ngẫu nhiên. Lớp tích chập ban đầu của mô hình CNN-LSTM của chúng em đang mất đi một số thông tin về thứ tự/chuỗi của văn bản. Do đó, nếu thứ tự của lớp tích chập thực sự không cung cấp cho chúng ta bất kỳ thông tin nào, lớp LSTM sẽ hoạt động chỉ như một lớp kết nối đầy đủ. Mô hình này dường như không thể khai thác được toàn bộ khả năng của lớp LSTM và do đó không đạt được tiềm năng tối đa của nó. Trên thực tế, nó thậm chí còn tồi tệ hơn so với một mô hình LSTM thông thường. Mặt khác, mô hình LSTM-CNN dường như là tốt nhất vì lớp LSTM ban đầu của nó dường như hoạt động như một bộ mã hóa sao cho đối với mỗi token trong đầu vào có một token đầu ra chứa thông tin không chỉ của token gốc, mà còn của tất cả các token trước đó. Sau đó, lớp CNN sẽ tìm các mẫu cục bộ bằng cách sử dụng biểu diễn phong phú hơn này của đầu vào gốc, cho phép đạt được độ chính xác tốt hơn.