HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Đặng Đình Quân

XÁC ĐỊNH TỶ LỆ TIN XẤU TRÊN BÁO ĐIỆN TỬ TIẾNG VIỆT BẰNG PHƯƠNG PHÁP HỌC SÂU

Chuyên ngành: KHOA HỌC MÁY TÍNH

Mã số: 8.48.01.01

TÓM TẮT LUẬN VĂN THẠC SĨ

HÀ NỘI - NĂM 2020

Luận văn được hoàn thành tại:

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Người hướng dẫn khoa học: PGS. TS. Trần Quang Anh

Phản biện 1: …….

Phản biện 2: …….

Luận văn sẽ được bảo vệ trước Hội đồng chấm luận văn thạc sĩ tại Học viện Công nghệ Bưu chính Viễn thông

Vào lúc: ..... giờ …..... ngày ..... tháng ..... năm .....

Có thể tìm hiểu luận văn tại:

- Thư viện của Học viện Công nghệ Bưu chính Viễn thông.

MỞ ĐẦU

Với sự phổ biến của Internet, báo điện tử đã trở thành kênh thông tin quan trọng đối với đời sống xã hội ngày nay. Khác với tạp chí là chủ yếu cung cấp thông tin mang tính tham khảo/học thuật về một lĩnh vực chuyên biệt, báo điện tử là sự phản ánh về hiện thực xã hội.

Bộ Thông tin và Truyền thông (TT&TT) đã đưa ra quan điểm rằng "cái xấu xuất hiện với tỉ lệ 30% trên mặt báo nghĩa là cái xấu trở thành cái chính của xã hội; cái xấu chiếm 20% là biểu hiện cái xấu có xu hướng trở thành cái chính trong xã hội; còn cái xấu chiếm 10% tuy không phải là cái chính nhưng đủ sức tác động đến con người". Nếu tỷ lệ cái xấu đăng tải trên một tờ báo điện tử không phản phù hợp với thực tế xã hội, tờ báo đó sẽ góp phần cung cấp cho độc giả cái nhìn sai lệch về thực trạng xã hội và làm "xói mòn niềm tin" của người dân [23].

Như vậy, việc đánh giá tỷ lệ cái xấu trên mặt báo điện tử là vô cùng cấp thiết. Các phương pháp học máy thống kê cổ điển đã được áp dụng để phân loại văn bản theo chủ đề (category) với kết quả tốt [10]. Các kỹ thuật học sâu (CNN, RNN, LSTM) tuy chưa vượt qua được các phương pháp cổ điển trong bài toán phân loại văn bản nhưng là một lựa chọn khả quan.

Từ những lý do trên, học viên lựa chọn đề tài "XÁC ĐỊNH TỶ LỆ TIN XẤU TRÊN BÁO ĐIỆN TỬ TIẾNG VIỆT BẰNG PHƯƠNG PHÁP HỌC SÂU" cho luận văn tốt nghiệp trình độ đào tạo thạc sĩ.

Mục đích, đối tượng và phạm vi nghiên cứu:

Mục đích nghiên cứu của luận văn là nghiên cứu các phương pháp học sâu dành cho dữ liệu dạng văn bản và ứng dụng vào bài toán xác định tin xấu trên báo điện tử tiếng Việt.

Đối tượng nghiên cứu của luận văn là các phương pháp học sâu dành cho dữ liệu dạng văn bản và bài toán xác định tin xấu dành cho bài báo điện tử tiếng Việt.

Phạm vi nghiên cứu của luận văn là các bài viết thuộc hai chuyên mục "đời sống" và "kinh doanh" trên báo điện tử tiếng Việt.

Phương pháp nghiên cứu:

- Về mặt lý thuyết: Thu thập, khảo sát, phân tích các tài liệu và thông tin có liên quan đến bài toán xác định tỷ lệ tin xấu trên báo điện tử tiếng Việt và các phương pháp học sâu áp dụng cho dữ liệu văn bản.

- Về mặt thực nghiệm: Xây dựng tập dữ liệu tin xấu tiếng Việt, làm thí nghiệm cài đặt và huấn luyện một số mô hình dự đoán, tổng hợp và so sánh kết quả thí nghiệm giữa các mô hình khác nhau để tìm ra ưu, nhược điểm và khả năng áp dụng của từng phương pháp.

Kết cấu của luận văn gồm 3 chương chính như sau.

Chương 1: Sơ lược về học máy, học sâu và bài toán xác định tỷ lệ tin xấu.

Chương 2: Phương pháp xác định tỷ lệ bài viết nói về cái xấu trên báo điện tử tiếng Việt.

Chương 3: Đánh giá phương pháp xác định tỷ lệ bài viết nói về cái xấu trên báo điện tử tiếng Việt.

Chương 1 – SƠ LƯỢC VỀ HỌC MÁY, HỌC SÂU VÀ BÀI TOÁN XÁC ĐỊNH TỶ LỆ TIN XẤU

1.1. GIỚI THIỆU BÀI TOÁN XÁC ĐỊNH TỶ LỆ TIN XẤU

Để xác định tỷ lệ tin xấu của một trang báo điện tử, bài toán đặt ra đó là làm sao để gán nhãn tốt/xấu cho mỗi bài viết trên trang báo đó. Thông tin trên mỗi bài báo điện tử thường bao gồm cả chữ viết, hình ảnh, âm thanh và video. Trong đó, phần lớn các bài báo điện tử có chứa cả nội dung chữ viết và hình ảnh. Nội dung video ngày càng trở lên phổ biến nhưng chưa chiếm đa số trong các trang báo điện tử. Trên hầu hết các trang báo điện tử, hình ảnh trong bài viết đều được ghi chú bằng phụ đề miêu tả nội dung bức ảnh. Trong phạm vi thời gian cho phép của luận văn, học viên lựa chọn tập trung nghiên cứu về nội dung văn bản của các bài báo.

Bài toán tổng quát mà luận văn cần giải quyết đó là bài toán phân loại với một nhãn và hai lớp. Giải pháp cần đưa ra được nhãn chính xác đối với đầu vào là nội dung dạng text của một bài báo, từ đó tính được tỷ lệ phần trăm tin xấu trong tổng số các bài viết trên một trang báo điện tử.

1.1.1. Định nghĩa về tin xấu

Tin nói về sự việc, hiện tượng tiêu cực (không phân biệt trong nước hay thế giới), nói về những sự việc mang tính chất phản cảm, không hay, không đẹp, gây tác hại đến môi trường, kinh tế, xã hội… Tin xấu là tin nói về thực trạng đáng buồn của xã hội, khó khăn của nền kinh tế, thiên tai.

Luận văn không có mục đích đưa ra định nghĩa chuẩn về tin xấu. Thay vào đó, nghiên cứu này đặt mục tiêu thử nghiệm hiệu quả của các mô hình học máy trong việc phân biệt/phát hiện tin xấu theo một định nghĩa cụ thể.

1.1.2. Phân loại văn bản

Phân loại văn bản là bài toán cổ điển và phổ biến trong khoa học máy tính nói chung và trong lĩnh vực học máy nói riêng. Mục tiêu của bài toán là xây dựng mô hình phần mềm để tự động phân loại văn bản thành hai hoặc nhiều lớp. Bài toán phân loại văn bản được giải quyết phổ biến với các phương pháp học máy. Gần đây, học sâu đã trở thành phương pháp phổ biến để giải quyết bài toán này.

1.1.3. Phân tích cảm xúc

Phân tích cảm xúc hoặc khai phá quan điểm là nghiên cứu tính toán về ý kiến của con người, tình cảm, cảm xúc, đánh giá và thái độ đối với các thực thể như sản phẩm, dịch vụ, tổ chức, cá nhân, vấn đề, sự kiện, chủ đề và thuộc tính của họ. Sự khởi đầu và sự phát triển nhanh chóng của lĩnh vực này trùng khớp với các phương tiện truyền thông xã hội trên web.

1.2. SƠ LƯỢC VỀ HỌC MÁY

Nền tảng của trí tuệ nhân tạo là khả năng máy móc có thể nhận thức như con người nhờ việc "học" từ các ví dụ. Việc học của một cỗ máy thông minh có nhiều điểm tương đồng với quá trình học của con người. Học máy (machine learning) mô phỏng lại quá trình học nói trên để khiến cho phần mềm máy tính có thể học và nhận thức được các dữ liệu số (văn bản, hình ảnh, âm thanh…). Mô hình học máy là một chương trình máy tính có chứa một tập bất kỳ các tham số và có hai chức năng cơ bản là *học* và *dự đoán*. Mỗi mô hình học máy đều có mục tiêu xác định, một tác vụ cụ thể mà nó cần thực hiện (phân loại, phân cụm, phát hiện, lọc, khôi phục…).

Tập hợp các ví dụ mẫu được gọi là *tập dữ liệu huấn luyện* (training data). Các đặc tính được chú ý của tập dữ liệu huấn luyện đó là độ lớn và tính đại diện

(representativeness). Não bộ của con người có khả năng lựa chọn các đặc tính để dựa vào đó nhận dạng một đối tượng. Quá trình đó trong học máy được gọi là trích chọn thuộc tính. Hiệu quả của mô hình đầu ra phụ thuộc rất nhiều vào việc lựa chọn những thuộc tính tốt. Với học sâu (deep learning), quá trình trích chọn thuộc tính được tự động hóa.

1.2.1. Học máy có giám sát

Hình thức phổ biến nhất trong học máy là học máy có giám sát (supervised learning). Trong học máy có giám sát, ví dụ mẫu được cung cấp kèm theo kết quả (gọi là *nhãn*) chuẩn cho chức năng học. Điều này tương tự với việc cho học sinh biết đáp án của của bài tập khi dạy học. Các bài toán tiêu biểu được giải bằng phương pháp học máy có giám sát là:

- Phân loại (classification)
- Hồi quy (regression)
- Phát hiện hành vi bất thường (anomaly detection)

1.2.2. Học máy không giám sát

Khác biệt lớn nhất giữa học máy không giám sát và có giám sát là sự vắng mặt của nhãn trong tập mẫu. Trong học máy không giám sát, chức năng học phải tự điều chỉnh bộ tham số mà không có nhãn chuẩn cho trước. Một vài bài toán được giải bằng học máy không giám sát là:

- Phân cụm (clustering)
- Giảm chiều dữ liệu (dimensionality reduction)

1.2.3. Học máy bán giám sát

Học máy bán giám sát là trường hợp chỉ có một phần nhỏ mẫu trong tập dữ liệu huấn luyện có nhãn kèm theo. Một cách tiếp cận trong hướng này là sử dụng

các mẫu có nhãn để huấn luyện ra một mô hình thô, sau đó dùng mô hình chưa hoàn thiện này để gán nhãn cho những mẫu còn lại.

1.2.4. Hàm mục tiêu, hàm tổn thất, hàm chi phí

Hàm mục tiêu (objective function) là hàm dự đoán trong đó có chứa bộ tham số tối ưu mà ta cần đi tìm. Như vậy, hàm mục tiêu là một hàm chưa biết mà ta hy vọng có thể tìm ra.

Hàm tổn thất là một hàm số của sự khác biệt giữa kết quả dự đoán và nhãn chuẩn. Hàm chi phí là một hàm tổng hợp các giá trị hàm tổn thất trên toàn bộ tập dữ liệu. Hàm chi phí có vai trò đặc biệt quan trọng trong quá trình huấn luyện mô hình.

1.2.5. Overfitting

Khi một mô hình học máy có hiệu quả kém trên tập huấn luyện, ta gọi trường hợp đó là underfitting. Khi một mô hình có hiệu quả rất cao trên tập huấn luyện nhưng hiệu quả trên tập thử nghiệm lại thấp, ta gọi trường hợp đó là overfitting. Hai chiều hướng này được coi là ngược nhau và trên thực tế ta có thể điều khiển xu hướng dẫn đến hai tình huống nói trên bằng cách điều chỉnh độ lớn hay độ phức tạp (capacity) của mô hình.

1.3. SƠ LƯỢC VỀ HỌC SÂU

1.3.1. Mạng nơ-ron

Ứng dụng có thể coi là cơ bản và phổ biến nhất trong lĩnh vực machine learning chính là mạng nơ-ron nhân tạo (gọi tắt là mạng nơ-ron). Lấy cảm hứng từ cấu trúc của bộ não sinh học, mạng lưới thần kinh bao gồm một số lượng lớn các đơn vị xử lý thông tin (được gọi là nơ-ron) được tổ chức thành các lớp, hoạt

động đồng nhất với nhau. Nó có thể được huấn luyện để thực hiện các tác vụ, như phân loại văn bản, bằng cách điều chỉnh các trọng số của kết nối giữa các nơ-ron trong mạng.

1.3.1.1. Perceptron

Mô hình mạng nơ-ron đầu tiên có tên là perceptron, được tạo ra để mô phỏng hoạt động não bộ con người. Perceptron là một mạng nơ-ron một lớp đơn giản, chỉ có khả năng giải những bài toán tuyến tính hoặc "học" trên không gian dữ liệu tuyến tính.

1.3.1.2. Mạng nơ-ron truyền thẳng nhiều lớp

Các mạng nơ-ron truyền thẳng sâu – Multilayer Perceptron (MLP) – là mô hình học sâu điển hình nhất với nhiều lớp ẩn (hidden layers), có thể giải những bài toán không tuyến tính [20].

1.3.2. Hàm kích hoạt

1.3.2.1. Softmax

$$S(y_i) = \frac{e^{y_i}}{\sum_j e^{y_j}}$$

$$\begin{bmatrix} 1.2 \\ 0.9 \\ 0.4 \end{bmatrix} \longrightarrow \text{Softmax} \longrightarrow \begin{bmatrix} 0.46 \\ 0.34 \\ 0.20 \end{bmatrix}$$

Hình 1.3: Minh họa cách hoạt động của hàm kích hoạt Softmax.
Nguồn: https://towardsdatascience.com/@ManishChablani

1.3.2.2. Sigmoid

$$S(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}$$

1.3.2.3. Hàm tanh

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

Đạo hàm của nó rất dễ tính toán và hoàn toàn không phụ thuộc vào giá trị đầu vào *x*, mà chỉ phụ thuộc vào giá trị đầu ra.

$$\tanh'(x) = 1 - \tanh(x)^2$$

1.3.3. Huấn luyện mạng nơ-ron

1.3.3.1. SGD

Gần như tất cả ứng dụng về học sâu đều sử dụng một thuật toán rất quan trọng: tối ưu giảm độ dốc ngẫu nhiên (Stochastic Gradient Descent). SGD được phát triển từ thuật toán gốc Gradient Descent. Nguyên lý hoạt động của SGD coi độ dốc (gradient) là một giá trị ước lượng. Tiếp theo, thuật toán SGD sẽ di chuyển tập trọng số trong không gian trọng số theo chiều xuống dốc dựa vào độc dốc vừa tìm được. Tốc độ di chuyển tập trọng số được quy định bởi một giá trị gọi là *learning rate*. Momentum cũng là một cơ chế được thiết kế để tăng tốc độ học cho SGD [4].

1.3.3.2. Backpropagation

Để có thể tối ưu một mạng nơ-ron bằng SGD thì ta tinh chỉnh các tham số trong ma trận tham số của các lớp mạng dựa vào đạo hàm của đầu ra đối với đầu vào (phương pháp gradient descent). Như vậy, để huấn luyện một mạng nhiều lớp, ta phải tính được đạo hàm của đầu ra đối với giá trị đầu vào. Nghiên cứu [20] chỉ ra rằng một mạng như vậy được huấn luyện một cách hiệu quả dựa trên một quy trình đơn giản được gọi là *back-propagation* (việc tính đạo hàm chuỗi).

1.3.3.3. Hàm kích hoạt ReLU

Công thức của hàm *ReLU* là $g(z) = \max\{0, z\}$. Hàm ReLU giữ được giá trị đạo hàm lớn trong quá trình backpropagation nên nó không gặp phải vấn đề đạo hàm biến mất (vanishing gradient) như các hàm kích hoạt khác.

1.3.3.4. Adam

Adam [14] là một trong những thuật toán tối ưu thích nghi, được giới thiệu vào năm 2014. Một cách khái quát, Adam kết hợp khả năng tự động thích nghi learning rate cho từng trục tọa độ với cơ chế momentum. Trong thực nghiệm, thuật toán Adam có tốc độ tìm ra kết quả nhanh hơn SGD nhưng lại có xu hướng dễ bị overfitting hơn và mô hình đào tạo bởi Adam thường không tốt bằng mô hình huấn luyện bởi SGD [21].

1.3.4. Một số hàm chi phí

1.3.4.1. MSE

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

1.3.4.2. Categorical Cross Entropy

$$CE = -\sum_{i=1}^{C} y_i * \log(\hat{y}_i)$$

Chương 2 – PHƯƠNG PHÁP XÁC ĐỊNH TỶ LỆ BÀI VIẾT NÓI VỀ CÁI XẤU TRÊN BÁO ĐIỆN TỬ TIẾNG VIỆT

2.1. BIỂU DIỄN THUỘC TÍNH

2.1.1. Character-level, word-level

2.1.2. One-hot encoding

2.1.3. Word Embedding

Các phương pháp học sâu dựa trên mạng nơ-ron hồi quy (RNN) thường sử dụng đầu vào ở cấp độ từ ngữ bởi vì có một kỹ thuật cho phép huấn luyện ra những vector số thực có thể dùng để đại diện cho từ ngữ. Kỹ thuật đó gọi là word embedding [16], ra đời cùng với sự bùng nổ của các phương pháp học sâu.

Trên lý thuyết, word embedding có ưu điểm vượt trội hơn so với one-hot encoding và term weighting. Vị trí tương đối giữa các từ được mô phỏng lại trong word embedding, những từ trái nghĩa sẽ có vị trí đối xứng trong không gian vector M chiều, những từ gần nghĩa sẽ có vị trí gần nhau và các phép tính không gian áp dụng trên các vector gần như thể hiện được nghĩa của các từ. Ví dụ [16]:

$$\text{vector}(\text{“king”}) - \text{vector}(\text{“man”}) + \text{vector}(\text{”woman”}) \approx \text{vector}(\text{“queen”})$$

2.1.4. Word2Vec

Một ứng dụng nổi tiếng của word embedding là thuật toán word2vec và bộ vector từ ngữ tiếng Anh được huấn luyện sẵn của Google. Các vector từ ngữ trong tập word2vec có độ dài 300 phần tử và được huấn luyện từ tập dữ liệu Google News có chứa 100 tỷ từ.

2.2. CÁC CẤU TRÚC MẠNG NƠ-RON SÂU

2.2.1. CNN

Mạng nơ-ron tích chập là mạng nơ-ron có ứng dụng một toán tử có tên là tích chập (convolution) trong ít nhất một lớp mạng.

2.2.1.1. Lớp tích chập

Một lớp tích chập điển hình trong các mạng nơ-ron có các tham số sau:

- **Số lượng bộ lọc**
- **Kích thước cửa sổ**

2.2.1.2. Pooling

Kỹ thuật pooling có tác dụng làm giảm độ phức tạp của dữ liệu đầu vào bằng cách chọn lấy một giá trị duy nhất từ một cửa sổ. Khi hàm max được sử dụng ở một lớp pooling, ta gọi lớp mạng đó là một lớp *max pooling*.

2.2.2. RNN

Khác với mạng nơ-ron truyền thẳng, trong mạng nơ-ron hồi quy (recurrent neural network, RNN) tồn tại các lớp mà đầu ra của nó được dùng làm đầu vào của chính nó. Hay nói cách khác, cấu trúc mạng hình thành một vòng tròn khép kín. Cấu trúc mạng như thế không chỉ có tiềm năng *mô phỏng được những logic phức tạp* (sâu, nhiều lớp) mà còn có *tính khái quát rất cao* vì một bộ tham số (parameters) được sử dụng cho tất cả các bước biến đổi dữ liệu.

2.2.3. Dropout

Ngoài việc có rất nhiều điểm tối ưu cục bộ, hay còn gọi là “bẫy” dành cho các thuật toán tối ưu dựa vào nguyên lý xuống dốc (GD-based) khiến cho việc tìm

giải pháp tối ưu trở nên khó khăn thì còn có vấn đề về overfitting. Dropout là kỹ thuật sinh ta để khắc phục tình trạng này. Nguyên lý của Dropout đó là ngẫu nhiên bỏ qua một phần nơ-ron của một lớp mạng (tạm thời "tắt" các nơ-ron đó ở một bước) để làm giảm sự phụ thuộc lẫn nhau giữa các nơ-ron trong một lớp.

2.3. THAM KHẢO TÀI LIỆU

Một ví dụ về mạng MLP dùng cho phân loại văn bản là Deep Average Network [13] với cấu trúc được minh họa trong Hình 2.1. Bằng việc tính trung bình cộng của các word vector, mô hình này đã bỏ qua thông tin về thứ tự sắp xếp của các từ trong văn bản.

Trong số các loại mạng RNN thì mạng LSTM vượt trội hơn cả vì nó có khả năng nắm bắt được ràng buộc giữa các từ trong văn bản dài hơn rất nhiều. Phương pháp Tree-LSTM [3] kết hợp nhiều đơn vị LSTM thành cấu trúc dạng cây để nắm bắt thông tin về ngữ nghĩa phức tạp trong văn bản.

Trong khi mạng RNN nhận diện đặc trưng theo thời gian thì mạng CNN lại nhận diện các dấu hiệu hiện hữu trên không gian [6]. Một trong những mô hình mạng CNN đầu tiên được ứng dụng cho bài toán phân loại văn bản là mô hình DCNN [7]. Mô hình này có mục tiêu là phân loại câu. Nó sử dụng các lớp pooling có kích thước động (dynamic k-max pooling). Đầu vào của mạng DCNN là một ma trận được hợp thành từ word vector của các từ trong một câu.

2.4. PHƯƠNG PHÁP MLP

Mạng MLP được cấu tạo từ các lớp mạng kết nối toàn phần (fully connected). Trong các bộ công cụ học sâu, lớp kết nối toàn phần được nhắc đến với cái tên *dense layer* (lớp dày đặc). Trong phương pháp này, học viên sử dụng mạng MLP với 4 lớp ẩn. Lớp đầu vào có kích thước 3000 nơ-ron. Lớp ẩn thứ nhất là lớp dày đặc có 128 nơ-ron, sử dụng hàm kích hoạt *ReLU*. Lớp ẩn thứ hai là lớp dày đặc có 32 nơ-ron, sử dụng hàm kích hoạt *ReLU*. Lớp ẩn thứ ba là lớp Dropout với tỷ lệ 0.5. Lớp đầu ra có 1 nơ-ron, sử dụng hàm kích hoạt *sigmoid*.

Dữ liệu đầu vào được biểu diễn ở dạng one-hot encoding.

2.5. PHƯƠNG PHÁP LSTM

Phương pháp thứ hai được học viên áp dụng cho bài toán là mô hình mạng LSTM với đầu vào có độ dài thay đổi. Một lớp Dropout ở giữa đầu ra của M tế bào và lớp output có thể được sử dụng để tăng tính độc lập cho các tế bào LSTM và giảm khả năng mô hình bị overfitting.

2.6. PHƯƠNG PHÁP BI-LSTM-CNN

Lấy toàn bộ output của các bước thời gian từ mạng LSTM để hình thành một ma trận và ứng dụng các kỹ thuật từ mạng CNN và pooling trên ma trận đó.

Phương pháp này ứng dụng cấu trúc mạng LSTM hai chiều – Bidirectional LSTM, viết tắt là BI-LSTM. Nghiên cứu [5] cho thấy BI-LSTM có khả năng hiểu văn bản dạng ngôn ngữ tự nhiên tốt hơn so với LSTM nguyên bản. Ngoài ra, trong phương pháp này, học viên còn áp dụng thêm lớp Flatten để chuyển đầu ra của lớp Pooling từ 2D thành 1D và kỹ thuật Dropout để tránh tình trạng overfit. Thuật toán Adam [14] được lựa chọn để giúp tăng tốc quá trình huấn luyện.

Chương 3 – ĐÁNH GIÁ PHƯƠNG PHÁP XÁC ĐỊNH TỶ LỆ TIN XẤU TRÊN BÁO ĐIỆN TỬ TIẾNG VIỆT

3.1. TẬP DỮ LIỆU

3.1.1. Phạm vi dữ liệu thử nghiệm

Dữ liệu thử nghiệm được lấy từ chuyên mục "Thời sự" của báo điện tử VnExpress (*https://vnexpress.net*).

3.1.2. Thu thập dữ liệu

Dữ liệu được thu thập bằng công cụ lấy tin tự động do học viên tự phát triển.

3.1.3. Xử lý & gán nhãn dữ liệu

Loại bỏ những tin trùng lặp. Gán nhãn thủ công với hai lớp là "Tin xấu" và "Tin bình thường". Rà soát và chỉnh lại những bài viết bị tình trạng một đoạn văn lặp hai lần cũng như loại bỏ tên tác giả còn sót lại ở cuối bài viết mà công cụ tự động chưa loại bỏ hết. Loại bỏ định dạng HTML. Chuyển thành dạng chữ viết thường (lower-case). Loại bỏ các các dấu chấm, phẩy, xuống dòng… (punctuations), ký tự đặc biệt.

Sau khi thực hiện các bước thu thập và xử lý dữ liệu nêu trên, tập dữ liệu cuối cùng bao gồm tổng số 8546 bài viết. Số bài viết được gán nhãn tin xấu theo tiêu chí: 5200 bài viết.

3.2. THIẾT KẾ THÍ NGHIỆM

3.2.1. Thí nghiệm 1

So sánh hiệu quả của lớp Embedding trực tiếp và vector word2vec huấn luyện sẵn. Mô hình LSTM được sử dụng để thử nghiệm trên cùng tập dữ liệu như mô tả ở phần trước.

3.2.2. Thí nghiệm 2

So sánh 3 phương pháp MLP, LSTM và BI-LSTM-CNN với các tiêu chí recall, precision, accuracy và điểm số F_1 được sử dụng làm độ đo chung để so sánh các mô hình. Kỹ thuật *k-fold cross validation* với $k = 5$ được thực hiện để đánh giá kết quả trong thí nghiệm này.

3.2.3. Các độ đo để đánh giá kết quả

$$\mathrm{ACC} = \frac{\mathrm{tp} + \mathrm{tn}}{N}$$

$$\mathrm{Recall} = \frac{\mathrm{tp}}{\mathrm{tp} + \mathrm{fn}}$$

$$\mathrm{Precision} = \frac{\mathrm{tp}}{\mathrm{tp} + \mathrm{fp}}$$

$$\mathrm{F_1} = 2 \times \frac{\mathrm{recall} \times \mathrm{precision}}{\mathrm{recall} + \mathrm{precision}}$$

Công thức của F_β như sau:

$$\mathrm{F_\beta} = \frac{(1 + \beta^2) \times \mathrm{tp}}{(1 + \beta^2) \times \mathrm{tp} + \beta^2 \times \mathrm{fn} + \mathrm{fp}}$$

Trong công thức trên, fn được coi là có chi phí cao gấp β lần so với fp.

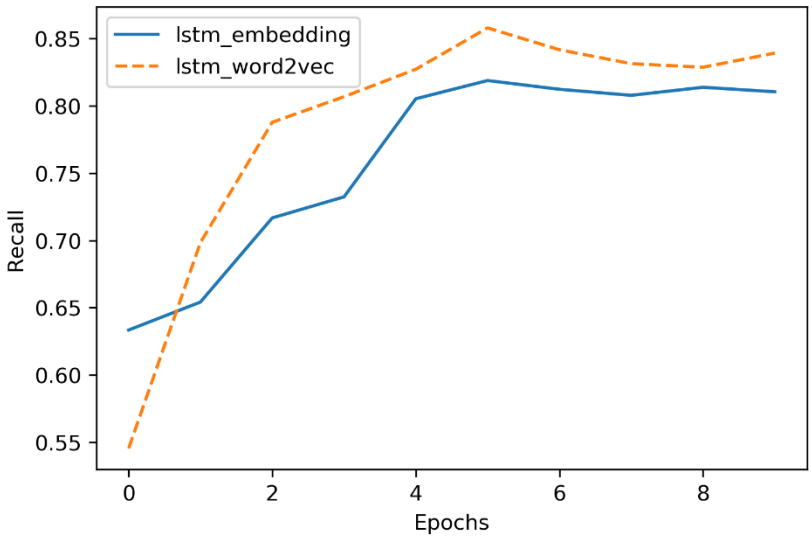
3.2.4. Kiểm chứng chéo

Trong quá trình huấn luyện, mô hình học máy chỉ được học từ các mẫu trong phần *train* và không được tiếp cận các mẫu trong phần *test*. Sau khi huấn luyện đạt kết quả như ý muốn với tập *train*, mô hình được áp dụng trên tập *test* để đánh giá hiệu quả với dữ liệu mới mà mô hình phân loại chưa từng thấy.

Trên thực tế, một cách làm khác thường được áp dụng nhiều hơn đó là *k-fold cross validation*. Với kỹ thuật này, tập dữ liệu mẫu được chia làm *k* phần đồng đều và mô hình được huấn luyện và thử nghiệm *k* lần rồi lấy kết quả trung bình. Ở mỗi lần, một trong *k* phần được sử dụng làm tập *test*, những phần còn lại được hợp lại dùng làm tập *train*.

3.3. KẾT QUẢ THÍ NGHIỆM

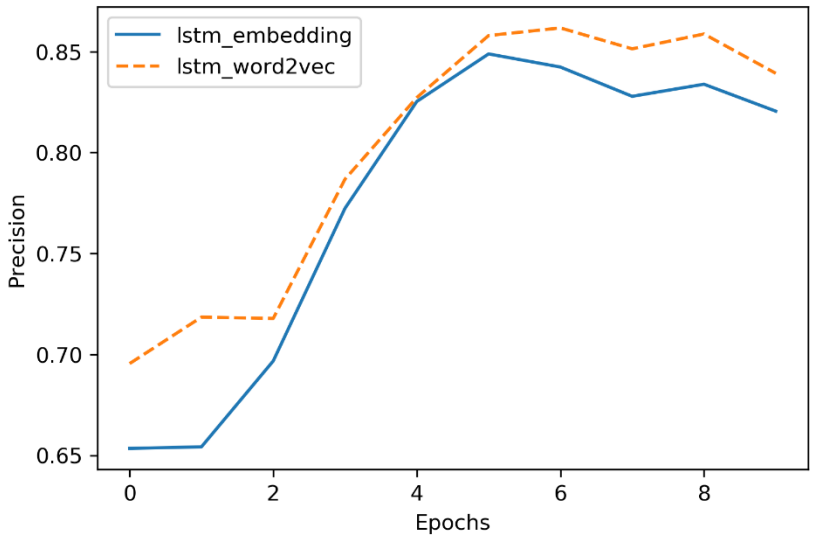
3.3.1. Thí nghiệm 1



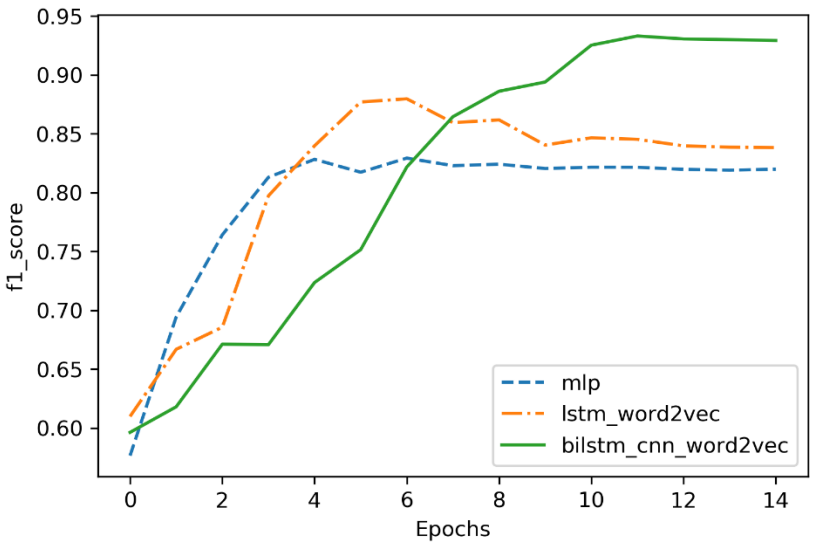
Hình 3.1: Biểu đồ độ đo Recall qua 10 epochs huấn luyện mô hình LSTM với lớp Embedding và với vector word2vec huấn luyện trước.

3.3.2. Thí nghiệm 2

Ở các chỉ số, phương pháp BI-LSTM-CNN đều đạt mức xấp xỉ 0.9 (90%) với accuracy cao nhất đạt 0.91615, điểm số F_1 đạt cao nhất 0.93304.



Hình 3.2: Biểu đồ độ đo Precision qua 10 epochs huấn luyện mô hình LSTM với lớp Embedding và với vector word2vec huấn luyện trước.



Hình 3.6: Kết quả thí nghiệm 2 với tiêu chí F_1 của 3 mô hình phân loại.

KẾT LUẬN

Trong luận văn này, học viên đã tiến hành nghiên cứu tài liệu về các phương pháp giải bài toán phân loại văn bản để áp dụng cho bài toán xác định tỷ lệ tin xấu trên báo điện tử tiếng Việt. Các kiến thức nền tảng về học máy và học sâu đã được trình bày theo trình tự từ cơ bản đến nâng cao. Luận văn cũng đi từ những vấn đề lý thuyết đến các ứng dụng mang tính thực nghiệm với mục tiêu xuyên suốt là để giải quyết bài toán đề ra một cách hiệu quả nhất. Thông qua quá trình tham khảo tài liệu, trong nhiều phương pháp được tóm tắt và thảo luận thì học viên đã lựa chọn và điều chỉnh ba phương pháp học sâu tiêu biểu để giải bài toán xác định tỷ lệ tin xấu: (1) phương pháp MLP đại diện cho nhóm các mạng nơ-ron truyền thẳng truyền thống, (2) phương pháp LSTM đại diện cho nhóm mạng nơ-ron hồi qui và (3) phương pháp BI-LSTM-CNN đại diện cho nhóm mô hình kết hợp nhiều cấu trúc mạng khác nhau. Qua thử nghiệm trên tập dữ liệu do học viên tự thu thập và xử lý, phương pháp BI-LSTM-CNN cho hiệu quả tốt và ổn định hơn hai phương pháp còn lại một cách đáng kể.