ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH

CÁCH TIẾP CẬN HIỆN ĐẠI TRONG XỬ LÝ NGÔN NGỮ TỰ NHIÊN (055256)

SENTIMENT ANALYSIS

GVHD: PGS. TS. Quản Thành Thơ
Sinh viên: Nguyễn Tuấn Minh - 2110359.

Tp. Hồ Chí Minh, Tháng 5 năm 2024

Contents

1 Tập dữ liệu

Tập dữ liệu VLSP 2016 được sử dụng cho bài toán phân tích cảm xúc, được chia thành 2 phần là tập huấn luyện và tập kiểm tra. Mỗi phần bao gồm hai cột: "Class" và "Data". Cột "Class" thể hiện phân loại cảm xúc của người dùng với các giá trị -1, 0, 1 tương ứng với các loại cảm xúc khác nhau. Cột "Data" chứa nội dung của các bình luận người dùng. Tập dữ liệu huấn luyện có tổng cộng 5100 mẫu, tập kiểm tra có 1050 mẫu.

Đây là một tập dữ liệu đa dạng với phân bố cân đối giữa các lớp cảm xúc trong tập huấn luyện, cung cấp nguồn tài nguyên phong phú cho việc phát triển và đánh giá các mô hình phân tích cảm xúc trong tiếng Việt.

2 Tiền xử lý dữ liệu

Quá trình tiền xử lý dữ liệu là bước quan trọng không thể thiếu trong mọi bài toán sử dụng các mô hình Machine Learning (ML), Deep Learning (DL), cụ thể trong trường hợp này là bài toán Sentiment Analysis (phân tích cảm xúc). Trong quá trình này, dữ liệu thô được chuyển đổi thành dữ liệu sạch, phù hợp cho việc đào tạo các mô hình ML, DL. Dưới đây là các bước tiền xử lý cụ thể:

1. **Loại bỏ URLs:** Dữ liệu văn bản thường xuyên chứa các URL, có thể không góp phần vào việc phân tích cảm xúc. Do đó, bước tiền xử lý này bao gồm việc xác định và loại bỏ các URL khỏi dữ liệu văn bản.

2. **Chuyển các ký tự đại diện cho các icon thành văn bản:** các icon thể hiện cảm xúc có thể được viết dưới dạng các ký tự đặc biệt như ')', ':',..., việc chuyển đổi những ký tự này thành văn bản sẽ hỗ trợ nhiều cho việc phân tích cảm xúc.

3. **Loại bỏ Stopword:** Stopwords là những từ phổ biến thường không mang nhiều ý nghĩa và thường được lọc ra khỏi văn bản, ví dụ những từ như "là", "và", "ở".

4. **Làm sạch văn bản:** Bước này gồm 2 phần: chuyển tất cả văn bản thành chữ viết thường và loại bỏ những ký tự đặc biệt không nằm trong ngôn ngữ đang phân tích. Điều này sẽ đồng thời giúp loại bỏ những chữ số (thường không đóng góp nhiều trong việc phân tích cảm xúc).

5. **Loại bỏ khoảng trắng:** Sau khi loại bỏ các phần tử văn bản khác, có thể xuất hiện những khoảng trắng thừa. Bước tiền xử lý này sẽ chuyển đổi nhiều khoảng trắng thành một khoảng trắng duy nhất để đảm bảo sự nhất quán của văn bản.

3 Các mô hình sử dụng

Để xây dựng mô hình Deep Learning cho bài toán phân tích cảm xúc: sinh viên tập trung vào 2 phần chính Word Embedding sử dụng CBOW và mô hình Deep Learning sẽ gồm có RNN, CNN, CRNN và GRU

3.1 Word Embedding: CBOW

CBOW (Continuous Bag of Words) là một phương pháp trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP) được sử dụng để tạo ra các biểu diễn vector của các từ trong một văn bản. CBOW thuộc loại mô hình word embedding, nơi mỗi từ trong từ vựng được biểu diễn dưới dạng một vector số học có số chiều cố định.

Ý tưởng cơ bản của CBOW là dự đoán từ hiện tại dựa trên ngữ cảnh của nó, tức là các từ xung quanh nó trong câu. Để làm điều này, CBOW sử dụng một cửa sổ trượt để lấy ngữ cảnh của từ hiện tại, sau đó sử dụng mô hình mạng nơ-ron để học cách dự đoán từ đó từ ngữ cảnh.

Quá trình hoạt động của CBOW như sau:

- **Tiền xử lý văn bản:** Văn bản đầu vào được chia thành các từ và được mã hóa thành các vector one-hot, tức là mỗi từ được biểu diễn bằng một vector có kích thước bằng kích thước của từ vựng, với tất cả các phần tử là 0 ngoại trừ phần tử ứng với từ đó có giá trị là 1.
- **Tạo ngữ cảnh:** CBOW chọn các từ xung quanh từ hiện tại để tạo thành ngữ cảnh, thường là một cửa sổ độ rộng cố định.
- **Dự đoán từ hiện tại:** Sử dụng ngữ cảnh được tạo ra từ bước trước để dự đoán từ hiện tại.
- **Huấn luyện mô hình:** CBOW được huấn luyện bằng cách điều chỉnh các trọng số của mạng nơ-ron để giảm thiểu sai số giữa từ dự đoán và từ thực tế.

Khi huấn luyện hoàn tất, các vector biểu diễn của các từ được sử dụng để biểu diễn ý nghĩa ngữ nghĩa và ngữ cảnh của chúng trong không gian vector. Các vector này có thể được sử dụng trong nhiều tác vụ NLP khác nhau như phân loại văn bản, dịch máy và phân tích cảm xúc.

Ở bài toán hiện tại, CBOW được sử dụng để thực hiện Word Embedding cho tất cả các cách tiếp cận Deep Learning phía sau.

3.2 Mô hình Deep Learning

3.2.1 Long Short-Term Memory (LSTM)

LSTM (Long Short-Term Memory) là một loại kiến trúc mạng nơ-ron được sử dụng rộng rãi trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP) và các tác vụ dự đoán chuỗi khác. Nó được thiết kế để xử lý vấn đề của việc truyền thông tin qua thời gian trong các mạng nơ-ron hồi quy (RNN) truyền thống, đặc biệt là vấn đề đối phó với việc biến mất hoặc phân tán gradient (vanishing or exploding gradient) khi huấn luyện mô hình trên các chuỗi dài.

LSTM giữ lại một bộ nhớ dài hạn và bộ nhớ ngắn hạn để lưu trữ thông tin trong quá trình xử lý chuỗi. Bằng cách này, nó có khả năng "nhớ" thông tin quan trọng từ quá khứ và "quên" thông tin không cần thiết. Cụ thể, LSTM sử dụng ba cổng chính để kiểm soát luồng thông tin: cổng quên (forget gate), cổng đầu vào (input gate) và cổng đầu ra (output gate).

- **Cổng quên (Forget gate):** Quyết định thông tin nào trong bộ nhớ ngắn hạn nên bị loại bỏ.
- **Cổng đầu vào (Input gate):** Quyết định thông tin nào từ đầu vào mới nên được cập nhật vào bộ nhớ.
- **Cổng đầu ra (Output gate):** Quyết định thông tin nào từ bộ nhớ nên được sử dụng để tạo ra đầu ra của mạng.

Nhờ vào cơ chế này, LSTM có khả năng xử lý các chuỗi dài và giữ lại thông tin quan trọng trong quá trình học. Điều này làm cho nó trở thành một công cụ mạnh mẽ trong các tác vụ liên quan đến chuỗi, như dịch máy, phân tích cảm xúc và sinh văn bản tự động.

3.2.2 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) là một loại mạng nơ-ron sâu được thiết kế đặc biệt cho việc xử lý dữ liệu có cấu trúc ruột (như hình ảnh và video). CNN đã đạt được nhiều thành công đáng kể trong nhiều lĩnh vực, bao gồm nhận dạng hình ảnh, phân loại văn bản, dự đoán chuỗi thời gian và nhiều ứng dụng khác trong lĩnh vực thị giác máy tính và xử lý ngôn ngữ tự nhiên.

CNN bao gồm một số lớp chính:

- **Lớp tích chập (Convolutional layer):** Lớp này áp dụng một hoặc nhiều bộ lọc tích chập qua đầu vào để tạo ra các tính năng ẩn. Các bộ lọc này học được các đặc trưng của dữ liệu đầu vào, như cạnh, góc và mẫu.

- **Lớp kích hoạt (Activation layer):** Sau mỗi lớp tích chập, một hàm kích hoạt như ReLU (Rectified Linear Unit) thường được áp dụng để tạo ra phi tuyến tính và kích hoạt các tính năng quan trọng.
- **Lớp gộp (Pooling layer):** Lớp này thực hiện gộp các đặc trưng đã được trích xuất từ lớp tích chập để giảm kích thước của dữ liệu và giảm thiểu độ phức tạp tính toán. Phổ biến nhất là lớp gộp tối đa (MaxPooling), trong đó chỉ giữ lại giá trị lớn nhất trong mỗi vùng gộp.
- **Lớp kết nối đầy đủ (Fully connected layer):** Sau khi thông tin đã được giảm cỡ thông qua các lớp tích chập và gộp, thông tin này được duỗi và đưa vào các lớp kết nối đầy đủ để thực hiện các bước cuối cùng của quá trình phân loại hoặc dự đoán.

CNN có khả năng học các đặc trưng cấp cao từ dữ liệu đầu vào mà không cần phải tạo ra các đặc trưng thủ công. Điều này giúp CNN trở thành một công cụ mạnh mẽ trong việc xử lý dữ liệu có cấu trúc như hình ảnh và video, đặc biệt là trong các tác vụ như nhận dạng hình ảnh, phân loại văn bản và nhiều ứng dụng khác.

3.2.3 Convolutional Recurrent Neural Network (CRNN)

CRNN (Convolutional Recurrent Neural Network) là một kiến trúc mạng nơ-ron sâu kết hợp giữa lớp tích chập (Convolutional layer) và lớp nơ-ron hồi quy (Recurrent layer). Kiến trúc này thường được sử dụng cho các tác vụ liên quan đến xử lý dữ liệu chuỗi, như nhận dạng văn bản trong hình ảnh, nhận dạng âm thanh và các ứng dụng khác trong lĩnh vực thị giác máy tính và xử lý ngôn ngữ tự nhiên.

CRNN kết hợp sức mạnh của các tính năng trích xuất từ lớp tích chập trong việc xử lý dữ liệu không gian (như hình ảnh) với khả năng mô hình chuỗi của lớp nơ-ron hồi quy trong việc xử lý dữ liệu thời gian (như văn bản). Điều này cho phép CRNN hiệu quả trong việc xử lý dữ liệu có cấu trúc phức tạp và có mối quan hệ không tuyến tính giữa các phần tử.

Cụ thể, CRNN thường có cấu trúc như sau:

- **Lớp tích chập (Convolutional layer):** Lớp này trích xuất các đặc trưng không gian từ dữ liệu đầu vào, như các đường biên, góc và mẫu.
- **Lớp kích hoạt (Activation layer):** Sau mỗi lớp tích chập, một hàm kích hoạt như ReLU thường được áp dụng để tạo ra phi tuyến tính và kích hoạt các tính năng quan trọng.

- **Lớp gộp (Pooling layer):** Lớp này thực hiện gộp các đặc trưng đã được trích xuất từ lớp tích chập để giảm kích thước của dữ liệu và giảm thiểu độ phức tạp tính toán.
- **Lớp nơ-ron hồi quy (Recurrent layer):** ớp này xử lý các đặc trưng chuỗi được trích xuất từ lớp tích chập để giải quyết các tác vụ liên quan đến thời gian, như dự đoán chuỗi thời gian hoặc nhận dạng văn bản trong hình ảnh.

CRNN là một kiến trúc mạnh mẽ và đa năng, thích hợp cho nhiều ứng dụng khác nhau trong xử lý dữ liệu có cấu trúc chuỗi.

3.2.4 Gated Recurrent Unit (GRU)

GRU (Gated Recurrent Unit) là một kiến trúc mạng nơ-ron hồi quy (RNN) cải tiến, được thiết kế để giải quyết vấn đề biến mất gradient và cải thiện khả năng học của mạng trong việc xử lý dữ liệu chuỗi dài.

Tương tự như LSTM (Long Short-Term Memory), GRU cũng được thiết kế để xử lý các chuỗi dữ liệu có thể kéo dài và giữ lại thông tin quan trọng qua nhiều bước thời gian. Tuy nhiên, GRU thường có cấu trúc đơn giản hơn và ít tham số hơn so với LSTM.

Một GRU bao gồm hai cổng chính:

1. **Cổng cập nhật (Update gate):** Quyết định phần nào của thông tin cũ nên được giữ lại và phần nào của thông tin mới nên được cập nhật.
2. **Cổng khôi phục (Reset gate):** Quyết định phần nào của thông tin cũ nên được "quên" để tạo ra thông tin mới.

Thông qua cách kết hợp của hai cổng này, GRU có khả năng học cách xử lý thông tin trong các chuỗi dữ liệu mà không cần phải duy trì một bộ nhớ dài hạn như LSTM. Điều này làm cho GRU trở thành một lựa chọn phổ biến trong việc xử lý dữ liệu chuỗi trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP) và các tác vụ liên quan đến thời gian, như dịch máy, nhận dạng giọng nói và nhiều ứng dụng khác. Đồng thời, với ít tham số hơn, GRU cũng có thể huấn luyện nhanh hơn và thích hợp cho các tác vụ với số lượng dữ liệu nhỏ.

3.3 Nhận xét đặc điểm các mô hình

Các kiến trúc RNN, CNN, CRNN và GRU đều có thể được áp dụng vào bài toán sentiment analysis, nhưng sự phù hợp của từng kiến trúc sẽ phụ thuộc vào đặc tính của dữ liệu và yêu cầu cụ thể của bài toán. Dưới đây là một số điểm mạnh và điểm yếu của mỗi kiến trúc và cách chúng có thể phù hợp với sentiment analysis:

1. **RNN (Recurrent Neural Network)**:
    - **Điểm mạnh**: RNN có khả năng xử lý dữ liệu chuỗi và giữ lại thông tin từ quá khứ, điều này có thể hữu ích trong sentiment analysis để hiểu được ngữ cảnh của từng từ trong câu.
    - **Điểm yếu**: RNN thường gặp vấn đề biến mất gradient khi xử lý các chuỗi dài, dẫn đến việc mất mát thông tin quan trọng. Điều này có thể là một hạn chế khi áp dụng RNN vào các văn bản dài và phức tạp.
2. **CNN (Convolutional Neural Network)**:
    - **Điểm mạnh**: CNN được sử dụng rộng rãi trong xử lý hình ảnh nhưng cũng có thể áp dụng vào xử lý văn bản. Với việc sử dụng các kernel convolutional để trích xuất các đặc trưng từ văn bản, CNN có thể hiệu quả trong việc phát hiện các mẫu ngữ cảnh quan trọng trong sentiment analysis.
    - **Điểm yếu**: CNN không tự định hình được các mối quan hệ thời gian giữa các từ trong một câu, điều này có thể là một hạn chế trong việc xử lý các văn bản có cấu trúc phức tạp.
3. **CRNN (Convolutional Recurrent Neural Network)**:
    - **Điểm mạnh**: CRNN kết hợp cả hai tính năng của CNN và RNN, giúp kết hợp cả khả năng trích xuất đặc trưng không gian và khả năng hiểu ngữ cảnh thời gian. Điều này có thể làm tăng hiệu suất của mô hình trong việc phân tích sentiment.
    - **Điểm yếu**: Kiến trúc CRNN có thể phức tạp hơn và đòi hỏi nhiều tài nguyên tính toán hơn so với các kiến trúc đơn giản hơn.
4. **GRU (Gated Recurrent Unit)**:
    - **Điểm mạnh**: GRU cung cấp một giải pháp đơn giản hóa so với LSTM nhưng vẫn giữ được khả năng học các mối quan hệ thời gian trong dữ liệu chuỗi. Điều này có thể làm giảm được vấn đề biến mất gradient và giúp mô hình học được các đặc trưng quan trọng trong sentiment analysis.
    - **Điểm yếu**: GRU có thể không đủ mạnh mẽ để xử lý các văn bản dài và phức tạp như LSTM, đặc biệt khi mối quan hệ thời gian giữa các từ rất quan trọng.

4 Kết quả

4.1 Chỉ số

Để đánh giá kết quả huấn luyện các mô hình, sinh viên sử dụng 4 chỉ số thường được dùng là Accuracy, Precision, Recall và F1. Kết quả huấn luyện 4 mô hình như sau:

	Accuracy	Precision	Recall	F1 Score	Model
0	0.599048	0.660619	0.599048	0.554997	LSTM
1	0.599048	0.635382	0.599048	0.565326	CNN
2	0.619048	0.632591	0.619048	0.600286	CRNN
3	0.620000	0.639313	0.620000	0.598782	GRU

Hình 1: Chỉ số đạt được khi kiểm tra 4 mô hình trên tập test

4.2 Nhận xét

- **Mô hình LSTM:** Có độ chính xác (Accuracy) và Recall khá tương đồng, khoảng 0.599, nhưng Precision cao hơn một chút so với Recall, và F1 Score thấp nhất trong số bốn mô hình, chỉ khoảng 0.554. Điều này cho thấy mô hình có phần không cân đối giữa precision và recall, với việc nhận dạng đúng các mẫu tích cực là tốt hơn so với việc phát hiện tất cả mẫu tích cực thực sự.
- **Mô hình CNN:** Cũng có độ chính xác giống hệt LSTM, nhưng có F1 Score cao hơn một chút, khoảng 0.565. Điều này có nghĩa là mô hình CNN có hiệu quả tổng hợp giữa Precision và Recall tốt hơn LSTM một chút.
- **Mô hình CRNN:** Đây là sự kết hợp của CNN và RNN, và nó cho thấy sự cải thiện về mọi mặt so với hai mô hình trước. Với Accuracy cao nhất là khoảng 0.619, cũng như Precision và Recall tương đồng, và F1 Score đạt khoảng 0.600. Điều này cho thấy mô hình CRNN cung cấp sự cân bằng tốt giữa việc phát hiện và phân loại chính xác các mẫu.
- **Mô hình GRU:** Mô hình này có độ chính xác gần như tương đương với CRNN, tuy nhiên có Precision cao hơn và F1 Score cũng ở mức khá cao,

khoảng 0.598. Điều này cho thấy GRU cũng là một lựa chọn tốt cho bài toán phân tích cảm xúc với hiệu suất tổng thể khá cân đối.

Tổng kết, mô hình CRNN và GRU cho thấy kết quả tốt hơn hai mô hình còn lại, với CRNN nổi bật hơn cả về mặt độ chính xác tổng thể. Mô hình CNN và LSTM có hiệu suất thấp hơn, nhưng tùy thuộc vào yêu cầu cụ thể của bài toán và tập dữ liệu, cũng như cách chúng được tối ưu hoá, có thể cần phải xem xét lại chúng trong các hoàn cảnh khác nhau.

References

[1] PGS. TS. Quản Thành Thơ - Trường Đại học Bách Khoa TPHCM, *Slide & Giáo trình môn học* Cách tiếp cận hiện đại trong xử lý ngôn ngữ tự nhiên. 2024.

[2] Huyen, N. T. (2024, April 24). *Recurrent Neural Network: Từ RNN đến LSTM.* Viblo. https://viblo.asia/p/recurrent-neural-network-tu-rnn-den-lstm-gGJ597z1ZX2

[3] *Sentiment analysis.* (2024, April 23). Wikipedia. https://en.wikipedia.org/wiki/Sentiment_analysis

[4] Van, C. P. (2024, April 24). *GRU - Mạng Neural hồi tiếp với nút có cổng.* Viblo. https://viblo.asia/p/gru-mang-neural-hoi-tiep-voi-nut-co-cong-3P0lPGevZox