HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

TỐNG NGUYÊN SƠN

PHÁT HIỆN CÂU CHỨA GỢI Ý TRÊN DIỄN ĐÀN TRỰC TUYẾN SỬ DỤNG MẠNG NƠ-RON

Chuyên ngành: Hệ thống thông tin

Mã số: 8.48.01.04

TÓM TẮT LUẬN VĂN THẠC SĨ KỸ THUẬT

(***Theo định hướng ứng dụng***)

HÀ NỘI – 2020

Luận văn được hoàn thành tại:

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Người hướng dẫn khoa học: TS. NGÔ XUÂN BÁCH

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 2020

LỜI NÓI ĐẦU

Trong thời gian qua, nhu cầu sử dụng mạng xã hội trực tuyến của người dùng không ngừng tăng lên, các trang mạng xã hội trực tuyến phổ biến như là Facebook, Twitter, Instagram, youtube, G+, blog v.v ngày càng phát triển. Con người sử dụng mạng xã hội trực tuyến không chỉ để giải trí như: cập nhật trạng thái, kết bạn, tán gẫu, nói chuyện mà họ còn dùng mạng xã hội trực tuyến như một nơi để chia sẻ thông tin, ý kiến trao đổi những nhu cầu, mong muốn, ý định hay dự định của họ trên các diễn đàn trực tuyến. Xuất phát từ thực tế đó việc phát hiện, phân loại những lời gợi ý mong muốn, ý định của người dùng sẽ mang lại giá trị thương mại, dịch vụ rất lớn.

Trong luận văn này, chúng tôi tập trung vào bài toán phát hiện câu chứa gợi ý trên các diễn đàn trực tuyến. Đây là bài toán có đầu vào là một câu được người dùng đăng lên các diễn đàn trực tuyến, câu đó có thể là những chia sẻ ,trao đổi cảm nhận, kinh nghiệm về các sản phẩm, dịch vụ, các vấn đề đời sống và mọi thứ xung quanh mà chính người dùng đó đã trải nghiệm và chúng ta cần phải xác định xem các chia sẻ, các câu đó có chứa gợi ý gì hay không? Nếu các câu có chứa gợi ý của người dùng thì gợi ý về nhu cầu, mong muốn, ý định v.v của người dùng đó về vấn đề gì như *:* du lịch, đồ ăn, thức uống, nghề nghiệp, giáo dục, hàng hóa & dịch vụ, sự kiện & hoạt động, không có ý định cụ thể. Bên cạnh đó, không phải tất cả những chia sẻ của người dùng đều thể hiện lời gợi ý rõ ràng và là nguồn dữ liệu, tài nguyên có ích. Vì vậy, luận văn sẽ tập trung chủ yếu vào phát hiện và phân loại các câu có chứa gợi ý của người dùng trên diễn đàn trực tuyến. Việc phát hiện, phân loại câu chứa gợi ý của người dùng đã và đang là đề tài nghiên cứu thời sự, mang tính cấp thiết hiện nay. Với các khách hàng, doanh nghiệp hay các nhà cung cấp dịch vụ việc biết được gợi ý, mong muốn của người dùng sẽ giúp họ cải tiến tốt hơn sản phẩm, hệ thống của mình để đảm bảo cung cấp đúng nội dung khách hàng cần, mở rộng số lượng người dùng quan tâm, quảng bá thương hiệu, hình ảnh. Hơn thế nữa, kết quả của bài toán phân loại câu chứa gợi ý người dùng có thể được ứng dụng làm đầu vào cho nhiều nghiên cứu khác như xây dựng hệ tư vấn xã hội dựa trên gợi ý người dùng, dự đoán sở thích người dùng, dự đoán xu hướng tương lai.

Luận văn "***Phát hiện câu chứa gợi ý trên diễn đàn trực tuyến sử dụng mạng Nơ-Ron***" thực hiện khảo sát, nghiên cứu các phương pháp xây dựng hệ thống phân loại câu chứa gợi ý được quan tâm nhất hiện nay. Từ đó đưa ra phương pháp phân loại câu phù hợp nhất cho hệ thống phân loại câu bằng tiếng Anh. Dựa trên những hướng tiếp cận đã đề cập ở trên, trong luận văn này, chúng tôi tiến hành áp dụng làm thực nghiệm dựa trên sự kết hợp một số đặc trưng ngôn ngữ tiếng Anh.

Các đặc trưng này sẽ được biểu diễn dưới dạng vectơ và làm đầu vào cho các thuật toán. Sau khi thu được kết quả của mô hình phân lớp *CNN, RNN*,LSTM luận văn sử dụng phương pháp để kiểm tra và lựa chọn kết quả tốt nhất. Kết quả thực nghiệm tốt nhất đạt được khi sử dụng thuật toán LSTM. Cụ thể kết quả thực nghiệm cho kết quả tốt nhất với bài toán " ***Phát hiện câu chứa gợi ý trên diễn đàn trực tuyến sử dụng mạng Nơ-Ron"***

Nội dung của luận văn gồm 03 chương:

Chương 1: Giới thiệu bài toán phân loại câu chứa gợi ý

Nội dung của chương, tổng quan nhất về gợi ý của người dùng trên diễn đàn trực tuyến, bài toán phân loại câu chứa gợi ý trên diễn đàn trực tuyến và cuối cùng là hướng tiếp cận nhằm giải quyết bài toán đề ra.

Chương 2: Phương pháp học máy cho bài toán phân loại câu chứa gợi ý trên diễn đàn trực tuyến

Nội dung của chương là trình bày một số phương pháp trích chọn lấy đặc trưng để giải quyết bài toán, các phương pháp học máy thống kê được sử dụng để tiến hành thực nghiệm cho bài toán phân loại câu chứa gợi ý trên diễn trực tuyến sử dụng mạng Nơ Ron.

Chương 3: Thực nghiệm và đánh giá

Nội dung chương nhằm nêu rõ và chi tiết các bước trong quá trình giải quyết bài toán. Trong chương này cũng sẽ trình bày quá trình thực hiện và thực nghiệm, đưa ra một số đánh giá, nhận xét các kết quả thu được.

Phần kết luận: Tóm lược những kết quả đạt được của luận văn. Đồng thời đưa ra những hạn chế, những điểm cần khắc phục và đưa ra định hướng nghiên cứu trong thời gian sắp tới.

CHƯƠNG 1: GIỚI THIỆU BÀI TOÁN PHÂN LOẠI CÂU CHỨA GỢI Ý

Trong chương này, luận văn trình bày giới thiệu chung về lĩnh vực xử lý ngôn ngữ tự nhiên (phần 1.1)và các ứng dụng trong thực tế (phần 1.2), cái nhìn tổng quan về bài toán phân loại câu chứa gợi ý, các cách tiếp cận bài toán, các nghiên cứu liên quan và kết quả luận văn đã đạt được.

1.1. Giới thiệu về xử lý ngôn ngữ tự nhiên

Xử lý ngôn ngữ tự nhiên (natural language processing – NLP) [2] [3] là một lĩnh vực nghiên cứu của trí tuệ nhân tạo, tập trung vào nghiên cứu các phương pháp, kỹ thuật cho phép xử lý ngôn ngữ tự nhiên bằng máy tính, từ đó xây dựng các chương trình, hệ thống máy tính xử lý ngôn ngữ của con người.

Xử lý ngôn ngữ tự nhiên được áp dụng trong nhiều bài toán và ứng dụng thực tế, trong nhiều lĩnh vực:

1.2. Bài toán phát hiện câu chứa gợi ý trên diễn đàn trực tuyến

1.2.1. Phân loại dữ liệu văn bản

Phân loại dữ liệu văn bản là quá trình phân lớp một đối tượng dữ liệu vào một hay nhiều lớp cho trước nhờ một mô hình phân lớp mà mô hình này được xây dựng dựa trên một tập hợp các đối tượng dữ liệu đã được gán nhãn từ trước gọi là tập dữ liệu học (tập huấn luyện).

1.2.2. Phát biểu bài toán phân loại phát hiện câu chứa gợi ý

Bài toán phân loại câu, phân loại văn bản được thấy rất nhiều trong các ứng dụng NLP (xử lý ngôn ngữ tự nhiên). Bài toán phát hiện câu chứa gợi ý trên diễn đàn trực tuyến sử dụng mạng Nơ- ron nhằm khai thác gợi ý có thể được định nghĩa là trích xuất các gợi ý từ văn bản phi cấu trúc, trong đó thuật ngữ ‘gợi ý’ đề cập đến cách diễn đạt các mẹo, lời khuyên, khuyến nghị,…v.v. Câu chứa gợi ý là câu thể hiện các ý kiến, góp ý, mong muốn đối với con người, thương hiệu, tranh luận xã hội, các sản phẩm thương mại, dịch vụ ..v.v thường được thể hiện thông qua các đánh giá trực tuyến, blog, diễn đàn thảo luận hoặc nền tảng truyền thông xã hội và có xu hướng chứa các biểu thức hàm ý về lời khuyên, mẹo, cảnh báo, khuyến nghị, ...v.v.

1.2.3. Ý nghĩa bài toán:

Phát hiện câu chứa gợi ý có thể được ứng dụng trong việc thu thập nhu cầu liên quan đến suy luận ngữ nghĩa trên nhiều ứng dụng của ngôn ngữ tự nhiên như: các ý kiến của người tiêu dùng đối với các thực thể thương mại như thương hiệu, dịch vụ và sản phẩm thường được

thể hiện thông qua đánh giá trực tuyến, blog, diễn đàn trực tuyến hoặc nền tảng truyền thông xã hội. Trên thế giới cũng đã có nhiều nghiên cứu về phát hiện câu chứa gợi ý sử dụng mô hình mạng Nơ-ron, Ví dụ như tại Semeval2019Task9/Subtask-A [5], đã có nhóm nghiên cứu đăng ký tham gia vào nhiệm vụ cùng nhiều nhà nghiên cứu khác thực hiện bên ngoài. Tuy nhiên tại Việt Nam chưa có nhiều dự án được nghiên cứu, triển khai và áp dụng vào trong thực tế.

1.3. Các nghiên cứu liên quan

Trong những năm gần đây, trên thế giới cũng đã có nhiều nghiên cứu về phát hiện câu chứa gợi ý sử dụng mô hình mạng Nơ-ron, Ví dụ như tại Semeval2019Task9/Subtask-A [5], đã có nhóm nghiên cứu đăng ký tham gia vào nhiệm vụ cùng nhiều nhà nghiên cứu khác thực hiện bên ngoài. Tuy nhiên tại Việt Nam chưa có nhiều dự án được nghiên cứu, triển khai và áp dụng vào trong thực tế.

Ngoài ra, tác giả Ahmed Husseini Orabi cùng cộng sự đã thực hiện một đề tài rất thiết thực và có ý nghĩa về việc sử dụng học sâu để phát hiện trầm cảm của người dùng Twitter: *"Deep Learning for Depression Detection of Twitter Users"* [7]. Công trình trình bày việc xử lý ngôn ngữ tự nhiên trên trực tuyến twitter, thực hiện đánh giá và so sánh trên một số mô hình học sâu, cụ thể là 3 mô hình CNN và 1 mô hình RNN và đưa ra kết quả về vấn đề rối loạn tâm thần và làm tiền đề cho hệ thống phát hiện các hành vi, cảm xúc tiêu cực của người dùng cá nhân trên trực tuyến.

1.4. Kết luận chương

Trong chương 1, luận văn đã giới thiệu tổng quan về bài toán xử lý ngôn ngữ tự nhiên. Tìm hiểu bài toán phân loại câu, văn bản và giới thiệu bài toán phát hiện câu chứa gợi ý trên diễn đàn trực tuyến, từ đó đưa ra những vấn đề cần làm rõ và giải quyết trong luận văn.

Trong chương 2, luận văn sẽ trình bày về hướng giải quyết cho bài toán phát hiện câu chứa gợi ý và đi sâu hơn trình bày về các phương pháp sẽ áp dụng để giải quyết bài toán.

CHƯƠNG 2: PHƯƠNG PHÁP PHÁT HIỆN CÂU CHỨA GỢI Ý SỬ DỤNG HỌC MÁY

Trong chương 2, luận văn tập trung trình bày một số phương pháp giải quyết bài toán (phần 2.1) và các thuật toán mô hình mạng Nơron được sử dụng khi làm thực nghiệm : CNN,RNN và LSTM (phần 2.2)

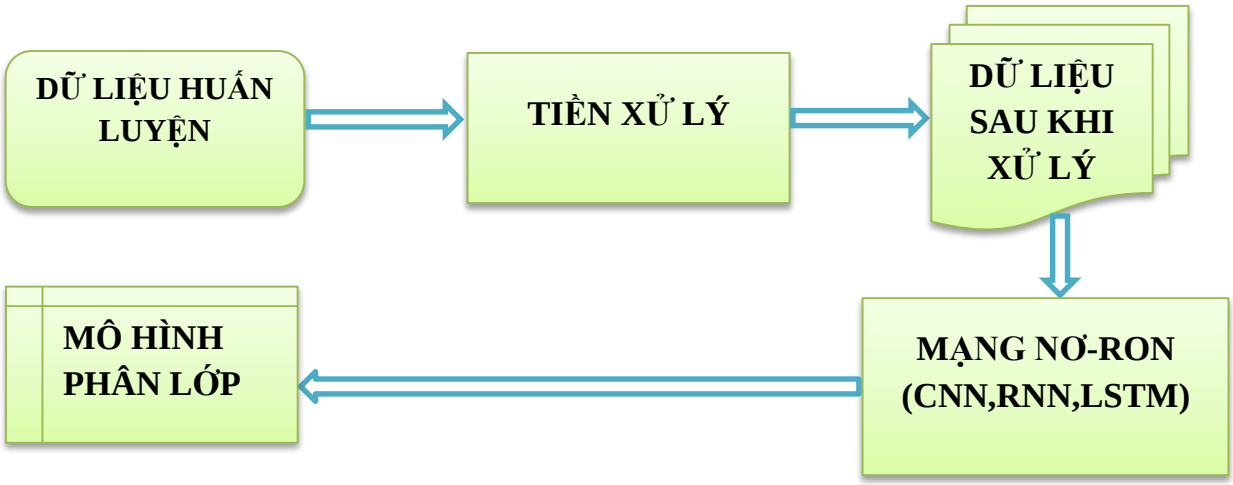
2.1. Phương pháp giải quyết bài toán:

Qua quá trình nghiên cứu, tập hợp các dòng trạng thái trên các diễn đàn trực tuyến và đã thu thập được tập các nội dung chia sẻ về những vấn đề xung quanh của người dùng qua bộ dữ liệu Semeval2019Task9/Subtask-A bao gồm: khoảng 833 câu [5], mục đích sẽ xác định nội dung các câu đó là câu có chứa gợi ý hay câu không chứa gợi ý. Luận văn đã tham khảo và tìm hiểu sau đó đưa ra được các bước thực hiện để xây dựng phương pháp giải quyết cho bài toán phát hiện câu chứa gợi ý được chia làm 2 giai đoạn sau:

- Giai đoạn huấn luyện
- Giai đoạn phân lớp.

a, Giai đoạn huấn luyện:

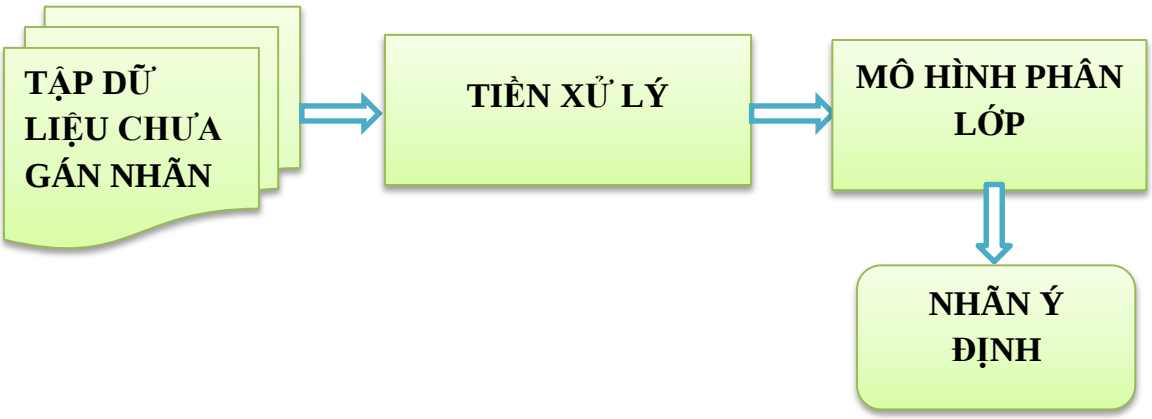
Giai đoạn này nhận đầu vào là tập dữ liệu huấn luyện gồm các nội dung dưới dạng văn bản đã được gán nhãn, sau khi xử lý tập dữ liệu và áp dụng các thuật toán huấn luyện sẽ cho đầu ra là một mô hình phân loại, cụ thể:



Hình 2.1 Mô hình giai đoạn huấn luyện

b, Giai đoạn phân lớp :

Nhận đầu vào là nội dung trạng thái của người dùng dưới dạng ngôn ngữ tự nhiên, sau quá trình xử lý và áp dụng mô hình phân loại sẽ cho ra nhãn phân loại của câu dữ liệu văn bản đầu vào, cụ thể được biểu diễn dưới sơ đồ sau:



Hình 2.2: Mô hình giai đoạn phân lớp

Tương tự như các bước trong giai đoạn huấn luyện, giai đoạn phân lớp có nhiệm vụ cụ thể :

- *Tiền xử lý:* Chuyển đổi các dòng trạng thái trong tập dữ liệu thành một hình thức phù hợp để phân loại như lọc nhiễu, loại bỏ các từ không mang ý định.
- *Mô hình phân lớp:* Sử dụng các thuật toán như Convolutional Neural network (CNN) và Recurrent Neural Network (RNN), Long Short Term Memory (LSTM) để tiến hành phân loại và gán nhãn ý định.

2.1.1. Tiền xử lý dữ liệu

Tiền xử lý dữ liệu là một bước rất quan trọng trong quá trình phân loại dữ liệu.Các kỹ thuật tiền xử lý dữ liệu phổ biến hiện nay bao gồm: xử lý dữ liệu bị khuyết (missing data), mã hóa các biến nhóm (encoding categorical variables), chuẩn hóa dữ liệu (standardizing data), co giãn dữ liệu (scaling data),v.v. Một số lỗi thường mắc phải trong khi thu thập dữ liệu là tính không đủ chặt chẽ, logic. Vì vậy, dữ liệu chứa các giá trị vô nghĩa và không có khả năng kết nối dữ liệu, ví dụ dữ liệu là các con số, các ký tự đặc biệt, các #hastag. Ở bước này chúng tôi sẽ tiến hành xử lý những dạng dữ liệu không chặt chẽ nói trên, những dữ liệu dạng này được xem như thông tin dư thừa, không có giá trị. Bởi vậy, đây là một quá trình rất quan trọng vì dữ liệu này nếu không được "làm sạch" sẽ gây nên những kết quả sai lệch nghiêm trọng.

2.1.2. Lọc nhiễu (loại bỏ từ không mang nghĩa)

Các từ không có nghĩa ở đây là các con số, các ký tự đặc biệt và không mang nghĩa. Ví dụ: "@@", "!! "EU !!!!!!!!!!!!!!!!!!!!!!!!!!!!, #2@",...

2.1.3. Loại bỏ các từ phổ biến (stop word):

Ngôn ngữ cũng giống như một đống gạo bị lẫn với thóc. Việc cần làm đó chính là chọn ra các hạt gạo chất lượng tốt nhất từ đống thóc đó. Những hạt thóc đó được gọi là stop words tức

là những từ không có ý nghĩa lắm đối với việc phân loại. Để tiết kiệm không gian lưu trữ và gia tăng tốc độ xử lý, sẽ không ghi nhận lại những từ quá phổ biến, quá chung chung và những từ này gọi là stop word [18]

{'his', 'because', 'shan', 'own', 'themselves', 'doesn', 'our', 'ourselves', 'up', 'should', 'under', 'most', 'at', 'having', 'where', 'him', 'below', 'am', 'wouldn', 'itself', 'your', 'll', 'from', 'their', 'ain', 'more', 'they', 'have', 'out', 'nor', 'of', 'weren', 'down', 'that', 'into', 'as', 'these', 'both', 'only', 'than', 'here', 'some', 'so', 'herself', 'how', 's', 'on', 'myself', 't', 'has', 'her', 'further', 'himself', 'again', 'hers', 'doing', 'before', 'very', 'just', 'd', 'between', 'in', 'during', 'yourself', 'whom', 'which', 'or', 've', 'what', 'against', 're', 'aren', 'was', 'yours', 'for', 'm', 'don', 'didn', 'she', 'not', 'y', 'been', 'its', 'mustn', 'and', 'ours', 'after', 'them', 'shouldn', 'you', 'few', 'couldn', 'mightn', 'same', 'haven', 'ma', 'be', 'theirs', 'but', 'such', 'wasn', 'were', 'those', 'a', 'to', 'an', 'did', 'too', 'with', 'about', 'who', 'isn', 'we', 'my', 'other', 'needn', 'i', 'when', 'the', 'then', 'once', 'all', 'will', 'won', 'is', 'this', 'he', 'off', 'while', 'yourselves', 'are', 'there', 'it', 'had', 'why', 'hadn', 'hasn', 'through', 'over', 'can', 'until', 'above', 'no', 'being', 'by', 'do', 'any', 'if', 'each', 'o', 'now', 'me', 'does'}

Hình 2.3: Một số stopword trong tiếng Anh [18]

Phần tiếp theo sẽ trình bày các mô hình mạng Nơ-ron được sử dụng trong luận văn.

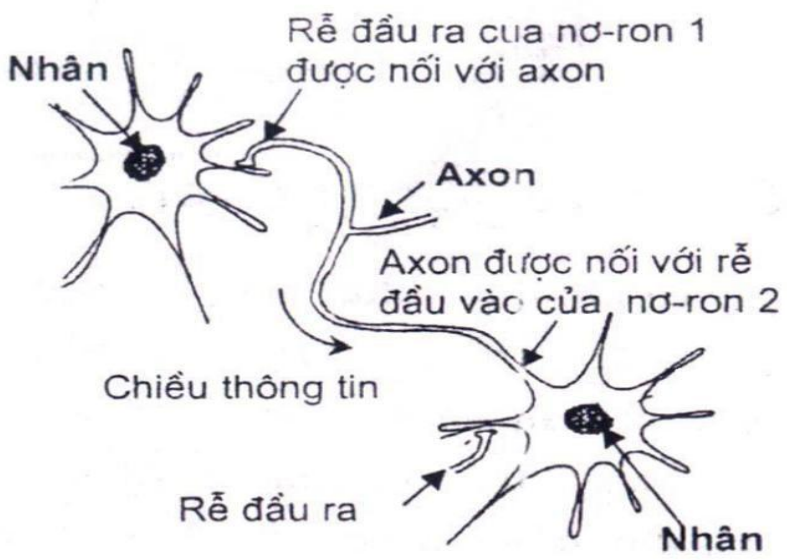
2.2. Giới thiệu chung mô hình mạng Nơ-ron:

2.2.1. Mạng Nơ-ron nhân tạo (ANN)

Mạng neural nhân tạo (Artificial Neural Network- ANN)[4] là mô hình xử lý thông tin được mô phỏng dựa trên hoạt động của hệ thống thần kinh của sinh vật, bao gồm số lượng lớn các Neuron được gắn kết để xử lý thông tin. ANN giống như bộ não con người, được học bởi kinh nghiệm (thông qua huấn luyện), có khả năng lưu giữ những kinh nghiệm hiểu biết (tri thức) và sử dụng những tri thức đó trong việc dự đoán phân loại các dữ liệu chưa biết (unseen data. Mạng neural nhân tạo đã được sử dụng rộng rãi từ những năm 1980 cho đến nay, vẫn được áp dụng rộng rãi trong nhiều ngành khoa học.Mốt số kiến trúc mạng Nơ ron phổ biến như: Mạng nơ ron tích chập(CNN), mạng nơ ron hồi qui(RNN), mạng nơ ron sâu(DNN), mạng bộ nhớ ngắn dài(LSTM),….

2.2.2. Mạng nơ-ron sinh học

Hệ thống thần kinh là tổ chức vật chất cao cấp và có cấu tạo vô cùng phức tạp. Hệ thần kinh được cấu tạo bởi nhiều yếu tố trong đó nơ-ron là khái niệm cơ bản nhất. Trong bộ não người có khoảng 10^{11} - 10^{12} tế bào thần kinh được gọi là các nơ-ron và mỗi nơ-ron lại liên kết với khoảng 10^4 nơ ron khác thông qua các khớp nối thần kinh synapse.

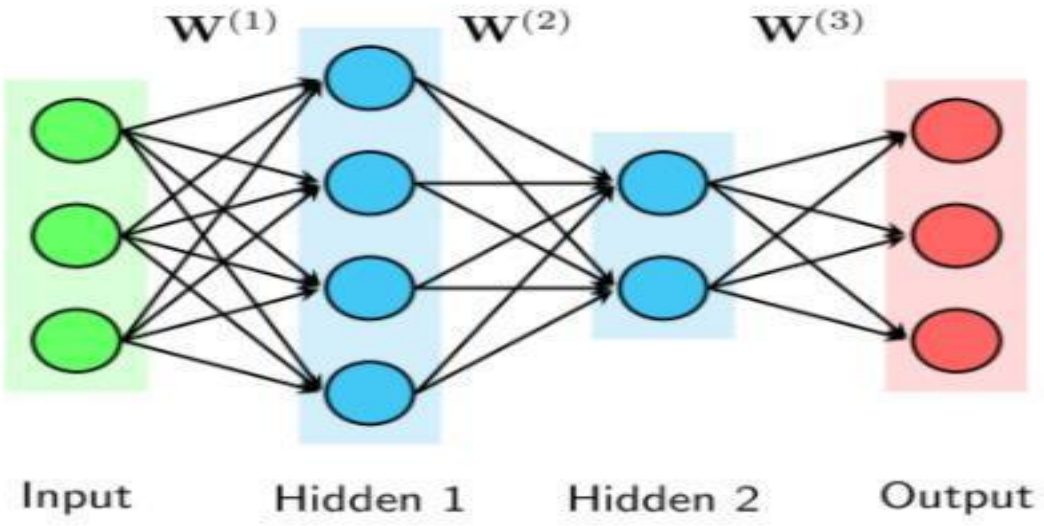


Hình 2.4: Mô hình mạng nơ ron sinh học[24]

(Nguồn: https://cs231n.github.io/)

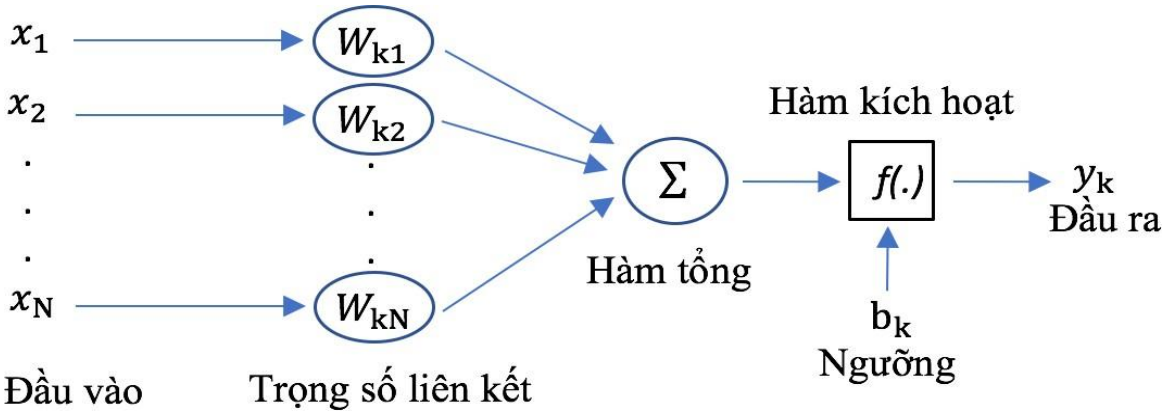
2.2.3. Kiến trúc tổng quát của mạng neural nhân tạo:

Mạng neural nhân tạo (Artificial Neural Network) gọi tắt là ANN là một mô hình xử lý thông tin phỏng theo cách thức xử lý thông tin của hệ thống nơ-ron sinh học[4][24]. Nó được tạo lên từ một số lượng lớn các phần tử gọi là neural kết nối với nhau thông qua các liên kết gọi là trọng số liên kết. Mạng neural nhân tạo thường được mô phỏng và huấn luyện từ tập mẫu. Qua quá trình huấn luyện, các trọng số liên kết sẽ được cập nhật sao cho giá trị gây lỗi là nhỏ nhất. Một mạng neural nhân tạo sẽ có 3 kiểu tầng:



Hình 2.5: Mạng neural 2 lớp ẩn[24]

(Nguồn: https://cs231n.github.io/)



Hình 2.6: Mô hình cấu tạo một neural

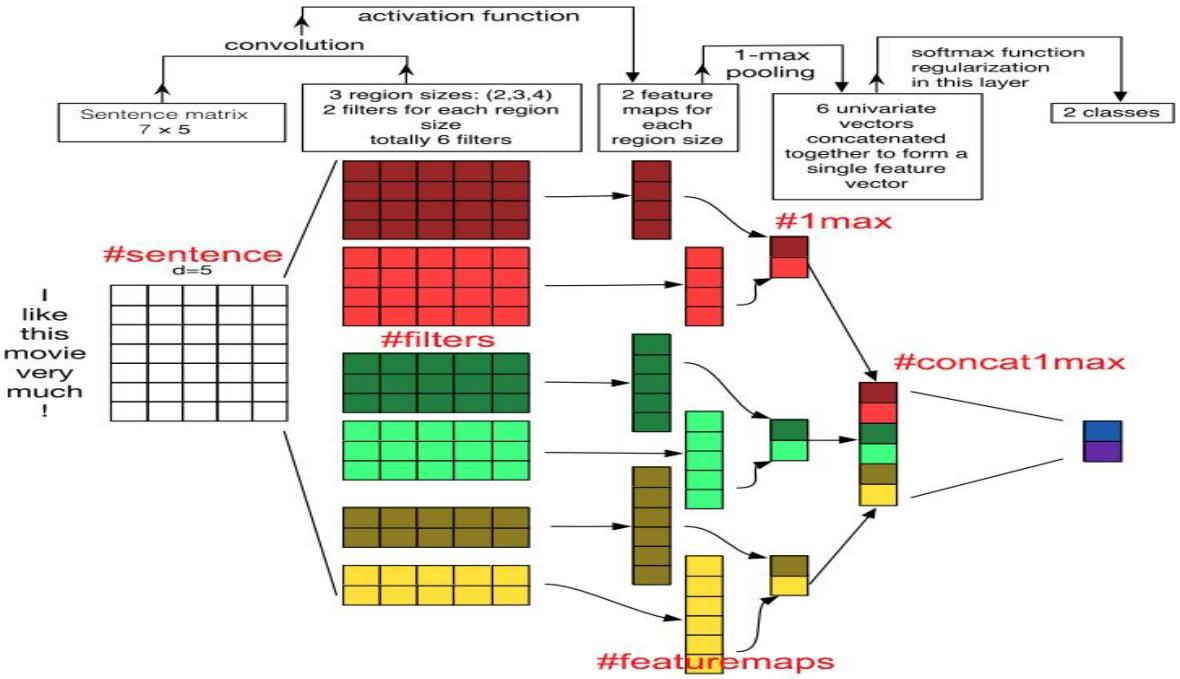
(Nguồn: https://tiendv.wordpress.com/2016/11/19/neural-networks/)

Kết quả của Sigmoid Function thuộc khoảng [0, 1] nên còn gọi là hàm chuẩn hóa (Normalized Function). Đôi khi thay vì sử dụng hàm chuyển đổi, tôi sử dụng giá trị ngưỡng (Threshold value) để kiểm soát các output của các nơ-ron tại một layer nào đó trước khi chuyển các output này đến các layer tiếp theo. Nếu output của một nơ-ron nào đó nhỏ hơn Threshold thì nó sẽ không được chuyển đến Layer tiếp theo. Mạng nơron nhân tạo đã được sử dụng để giải quyết nhiều bài toán thuộc nhiều lĩnh vực của các ngành khác nhau. Các nhóm ứng dụng mà mạng nơ-ron nhân tạo đã được áp dụng rất có hiệu quả là:

- Bài toán phân lớp (classification): Phân loại các đối tượng quan thành thành các nhóm. Ví dụ: phân loại chữ viết, nhận diện hình ảnh ...
- Bài toán dự đoán (preditive): Mạng nơ-ron nhân tạo đã được ứng dụng thành công trong việc xây dựng các mô hình dự báo sử dụng tập dữ liệu trong quá khứ để dự đoán số liệu trong tương lai. Ví dụ: dự báo thiên tai, dự báo chứng khoán ...
- Bài toán điều khiển và tối ưu hoá: Nhờ khả năng học và xấp xỉ hàm mà mạng nơ-ron nhân tạo đã được sử dụng trong nhiều hệ thống điều khiển tự động cũng như góp phần giải quyết những bài toán tối ưu trong thực tế.

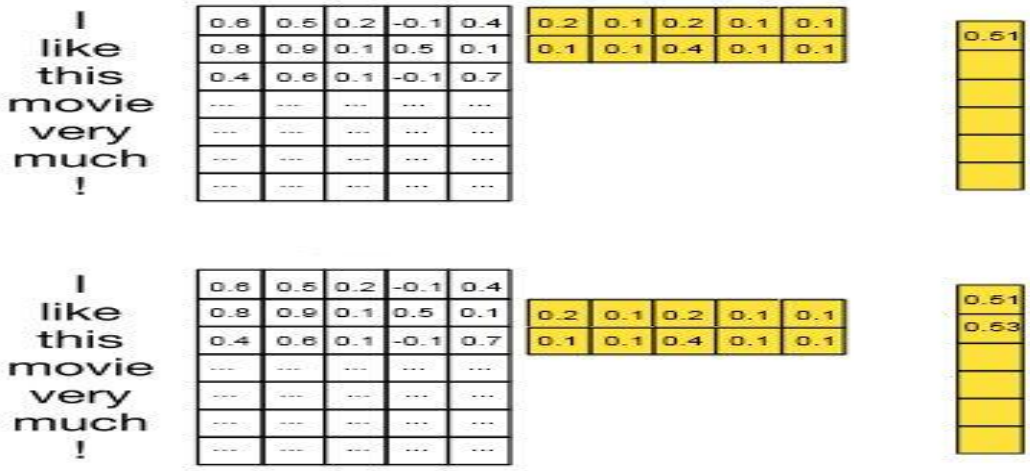
2.3. Mạng nơron tích chập CNN:

CNN- Mạng Nơ-ron tích chập ra đời nhằm khắc phục các nhược điểm của Deep neural network do các mạng lưới đào tạo ngày càng phức tạp. Mạng Nơ-ron Tích Chập được giới thiệu bởi Bengio, Le Cun, Bottou và Haffner vào năm 1998. Mạng nơron tích chập [19] là một trong những mạng truyền thẳng đặc biệt, một mô hình học sâu phổ biến và tiên tiến nhất hiện nay. Hầu hết các hệ thống nhận diện và xử lý ảnh hiện nay đều sử dụng mạng nơron tích chập vì tốc độ xử lý nhanh và độ chính xác cao. Trong mạng nơron truyền thống, các tầng được coi là một chiều, thì trong mạng nơron tích chập, các tầng được coi là 3 chiều, gồm: chiều cao, chiều rộng và chiều sâu. Mạng nơron tích chập có hai khái niệm quan trọng: kết nối cục bộ và chia sẻ tham số. Những khái niệm này góp phần giảm số lượng trọng số cần được huấn luyện, do đó tăng nhanh được tốc độ tính toán.



Hình 2.7: Mô hình thuật toán CNN [15]

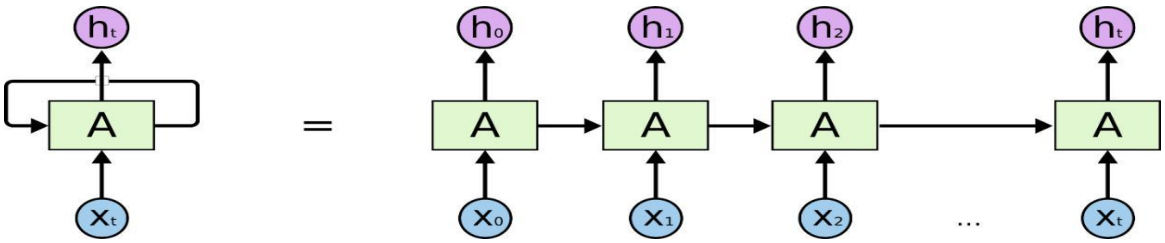
(Nguồn: Zhang, Y., & Wallace, B. (2015). A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification.)



Hình 2.8: Cách nhân tích chập giữa ma trận input với bộ lọc[27]

2.4. Mạng nơron hồi quy RNN:

a. Giới thiệu mạng nơron hồi quy RNN:



Hình 2.9: Mô hình mạng RNN không kiểm soát[21]

$$h_t = \sigma(W^H h_{t-1} + W^X x_t)$$

Hình 2.10: Công thức tính vector trạng thái ẩn tại thời điểm t

$$a_i = \frac{\exp(z_i)}{\sum_{j=1}^{C} \exp(z_j)}, \quad \forall i = 1, 2, \ldots, C$$

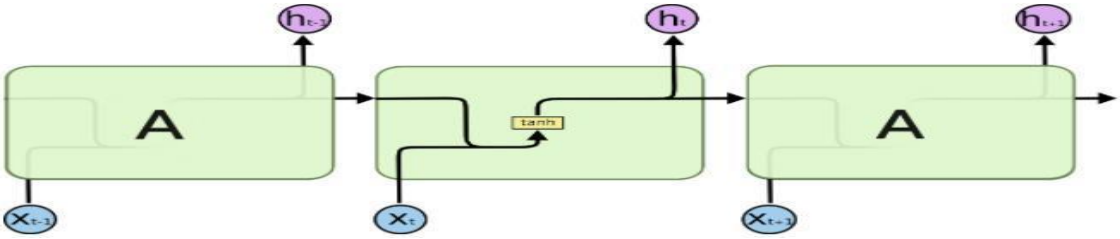
Hình 2.11: Hàm softmax

b. Vấn đề lưu trữ thông tin ngữ cảnh phụ thuộc lâu dài.

Xét một câu hỏi sau: " Số thứ nhất bằng 1. Chiếc xe đang chạy trên đường. Số thứ hai bằng 3. Tổng của hai số bằng mấy?".

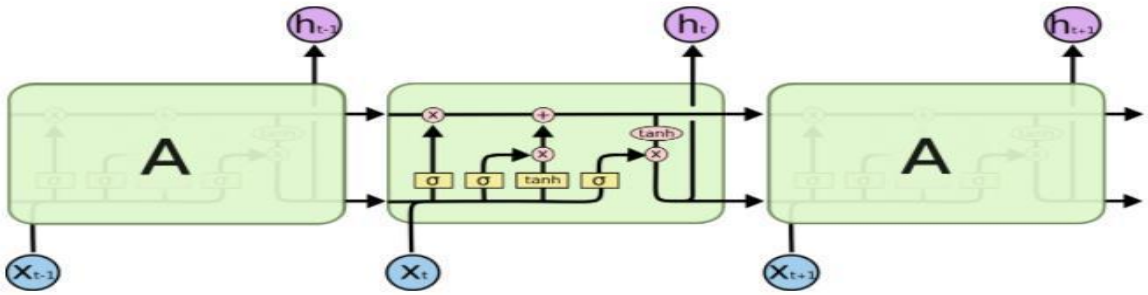
2.5. Mạng nơ-ron có bộ nhớ ngắn dài LSTM:

Mạng nơ-ron có bộ nhớ ngắn dài LSTM [21] là một loại RNN đặc biệt nó được cải tiến của mạng RNN nhằm giải quyết vấn đề học, lưu trữ thông tin ngữ cảnh có khả năng học các phụ thuộc dài. Với mô hình RNN, tại thời điểm t thì giá trị của vector ẩn ht chỉ được tính bằng một hàm tanh nói cách khác trong RNN tiêu chuẩn module lặp lại này có cấu trúc đơn giản với một lớp tanh duy nhất.



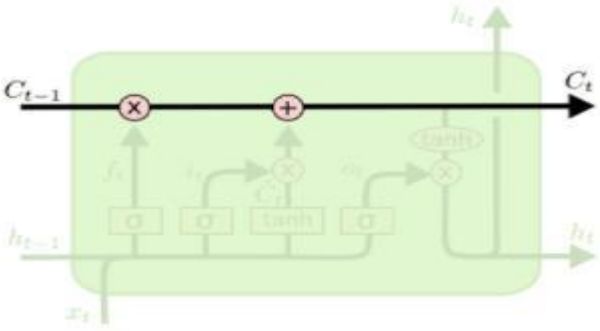
Hình 2.12: Module xử lý tính ht của RNN [21]

Các LSTM cũng có cấu trúc mắt xích giống như chuỗi này, nhưng các module lặp có cấu trúc khác hẳn. Thay vì chỉ có một layer mạng nơ ron, thì LSTM có tới bốn layer, tương tác với nhau theo một cấu trúc cụ thể rất đặc biệt.



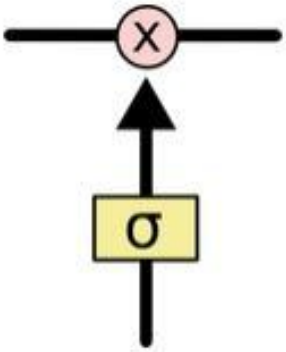
Hình 2.13: Module lặp lại của mạng LSTM chứa 4 lớp tương tác[21]

Các ký hiệu sử dụng trong mạng LSTM gồm có:



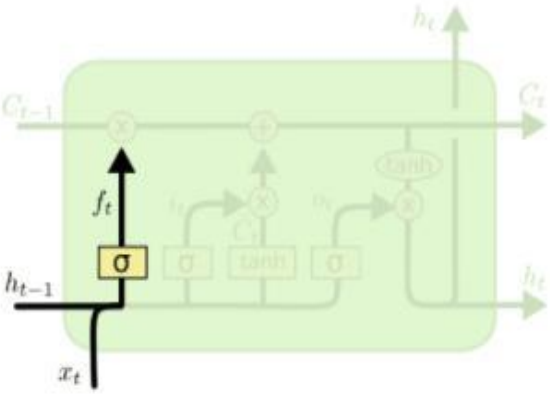
Hình 2.14: Cell state trong LSTM giống như một băng chuyền[21]

LSTM có khả năng loại bỏ hoặc thêm bớt thông tin vào cell state, được điều chỉnh quy định một cách cẩn thận bởi các cấu trúc gọi là cổng (gate). Các gate này là tuỳ chọn để định nghĩa thông tin đi qua. Chúng được tạo bởi lớp mạng thần kinh sigmoid và một nhân các thao tác toán tử pointwise.



Hình 2.15: Cổng trạng thái LSTM [21]

.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

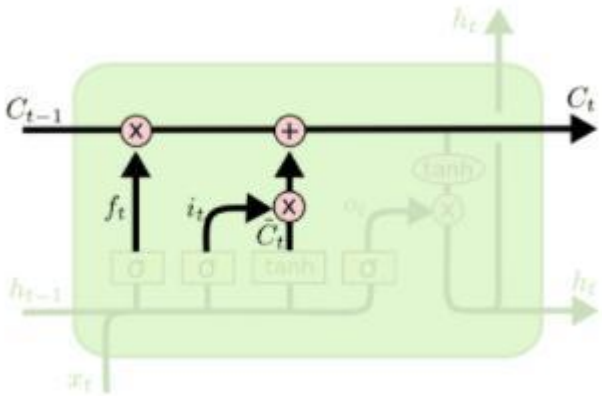
Hình 2.16: Cổng chặn ft[21]

Bước thứ hai, tại cell state cần quyết định thông tin nào cần được lưu lại. Có hai phần

là single sigmoid layer được gọi là "input gate layer"- cổng vào quyết định các giá trị sẽ cập nhật. Tiếp theo, một tanh layer tạo ra một vector mới $\tilde{C}t$ được thêm vào trong cell state.

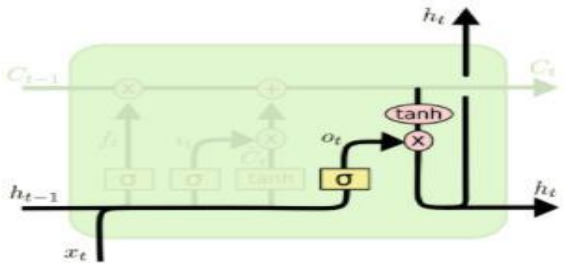
Hình 2.17: Cổng vào it và tanh[21]

Bước thứ ba, sẽ kết hợp hai thành phần này lại để cập nhật vào cell state. Lúc cập nhật vào cell state cũ, C_{t-1}, vào cell state mới C_t. Đưa state cũ hàm f_t, để quên đi những gì trước đó. Sau đó, sẽ thêm (i_t*C_t). Đây là giá trị ứng viên mới, co giãn (scale) số lượng giá trị mà ta muốn cập nhật cho mỗi state.



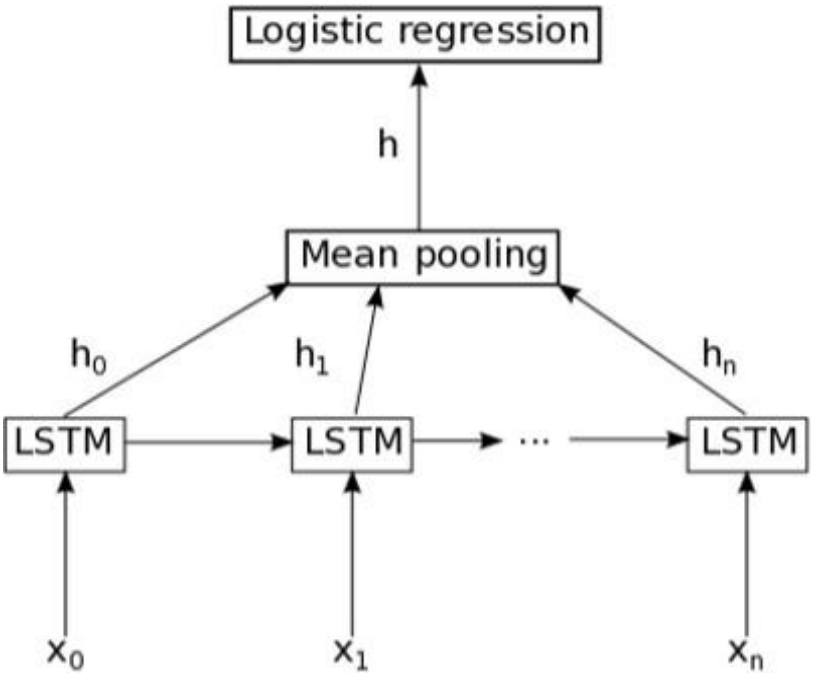
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Hình 2.18: Giá trị state Ct [21]



$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Hình 2.19: Giá trị cổng ra và vector trạng thái ẩn ht[21]



Hình 2.20: Mô hình LSTM luận văn sử dụng

(*http://hoctructuyen123.net/tong-quan-ve-phan-tich-cam-xuc-trong-tieng-viet/)*

Từ một chuỗi đầu vào x0, x1,… xn sử dụng các cơ chế tính toán nêu trên của các cổng vào, cổng ra và cổng chặn sẽ tính được tương ứng giá trị vector trạng thái ẩn h0, h1,…hn . Giá trị vector trạng thái ẩn tại các thời điểm sau đó được tính trung bình trên tất cả các dấu thời gian để được vector trạng thái h. Vector h sẽ đại diện cho câu đang xét. Cuối cùng, vector h được đưa vào một lớp hồi quy để gán nhãn, phân loại cho kết quả đầu ra.

2.6. Kết luận chương 2:

Chương 2 đã giới thiệu về hướng tiếp cận, các công trình nghiên cứu, kỹ thuật liên quan để phục vụ giải quyết bài toán. Chương này đi sâu về áp dụng phương pháp học máy phân lớp và phương pháp biểu diễu các đặc trưng mô hình trong bài toán phát hiện câu chứa gợi ý trên diễn đàn trực tuyến sử dụng mạng Nơ ron.

Chương tiếp theo sẽ trình bày về hệ thống phát hiện câu chứa gợi ý trên diễn đàn trực tuyến, mô hình giải quyết bài toán, tập dữ liệu sử dụng, cách thức tiến hành thực nghiệm, kết quả thực nghiệm.

CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ

Dựa vào những thuật toán áp dụng cho bài toán phân loại câu và tìm hiểu các phương pháp giải quyết bài toán, trong chương này, luận văn trình bày chi tiết quá trình thực nghiệm gồm có quá trình thu thập, thiết lập thực nghiệm, các phương pháp làm thực nghiệm, kết quả và đánh giá sau thực nghiệm.

3.1. Thông tin về bộ dữ liệu

Luận văn sử dụng dữ liệu thực nghiệm được thu thập từ Bộ dữ liệu tiếng Anh Semeval2019Task9/Subtask-A [5] tổng hợp từ các câu trên diễn đàn trực tuyến về sự phát triển nền tảng Window với dữ liệu huấn luyện 8500 câu và dữ liệu kiểm thử 833 câu. Mỗi câu được gắn nhãn phân loại theo các mục "có gợi ý", "không gợi ý" . Với dữ liệu này giúp chúng ta huấn luyện cho mạng neural của máy.

Mô tả file dữ liệu gồm:

Bảng 3.1: Mô tả dữ liệu thực nghiệm

Số thứ tự	**Tên cột**	**Ghi chú**
1	ID	Mã của câu
2	Classification	Phân loại câu: Giá trị sẽ là "có gợi ý" hay "không gợi ý"
3	Sentence	Nội dung của câu (tối đa 140 ký tự)

(Nguồn: https://github.com/Semeval2019Task9/Subtask-A *)*

Bộ dữ liệu tiếng Anh Semeval2019Task9/Subtask-A [5] này gồm các thư mục để huấn luyện, kiểm thử và cho kết quả

- Data:

File "**V1.4_Training_new.csv**" sử dụng cho huấn luyện

- File "**SubtaskA_EvaluationData_labeled.csv**" sử dụng cho kiểm thử

Bảng 3.2: Mô tả phân loại nhãn cho các tập dữ liệu thực nghiệm

Tập dữ liệu	***Câu Có gợi ý***	***Câu Không gợi ý***	***Tổng cộng***
Data/train	2085	6415	8500
Data/test	87	746	833

3.2. Môi trường thực nghiệm:

3.2.1. Ngôn ngữ lập trình python:

Python là một ngôn ngữ lập trình thông dịch do Guido van Rossum tạo ra năm 1990 [22]. Python hoàn toàn tạo kiểu động và dùng cơ chế cấp phát bộ nhớ tự động, do vậy nó Có

```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

Hình 3.1: Mô tả cú pháp, các dòng lệnh trong Python

(Nguồn: https://www.python.org/)

Hiện nay ngôn ngữ **Python** được xếp hạng thứ 3 trong Top 10 các ngôn ngữ lập trình phổ biến nhất đang được thế giới sử dụng:

Bảng 3.3: Bảng xếp hạng các ngôn ngữ lập trình năm 2020

Feb-20	Feb-19	Change	Programming Language	Ratings	Change
1	1		Java	17.36%	1.48%
2	**2**		**C**	**16.77%**	**4.34%**
3	3		Python	9.35%	1.77%
4	4		C++	6.16%	-1.28%
5	7	^	C#	5.93%	3.08%
6	5	v	Visual Basic .NET	5.86%	-1.23%
7	6	v	JavaScript	2.06%	-0.79%
8	8		PHP	2.02%	-0.25%
9	9		SQL	1.53%	-0.37%
10	20	^^	Swift	1.46%	0.54%

(Nguồn: https://www.tiobe.com/)

3.2.2 Giới thiệu về thư viện TensorFlow

TensorFlow là một thư viện do nhóm phát triển Google Brain của Google phát triển và phát hành mã nguồn mở vào tháng 11/2015. TensorFlow được cho là sử dụng trong nhiều sản phẩm thương mại của Google [23]. Hiện tại được sử dụng nhiều trong quá trình hiện thực hoá mạng neural trong Deep Learning.

Các khái niệm cơ bản:

Tensor: Tensor là khái niệm cơ bản nhất trong TensorFlow.

Bảng 3.4: Mô tả rank của tensor [23]

Rank	đơn vị số học	Ví dụ Python
0	Scalar	*s = 483*
1	Vector	*v = [1.1, 2.2, 3.3]*
2	Matrix	*m = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]*
3	3-Tensor	*t = [[[2], [4], [6]], [[8], [10], [12]], [[14], [16], [18]]]*
n	n-Tensor	*(n chiều) ….*

Shape: là chiều của tensor.

Ví dụ: t = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] có shape là [3, 3].

Bảng 3.5: Mô tả cú pháp shape của tensor [23]

Rank	Shape	Số chiều (Dimension)	Ví dụ
0	[]	0-D	*A 0-D tensor. A scalar.*
1	[D0]	1-D	*A 1-D tensor with shape [5].*
2	[D0, D1]	2-D	*A 2-D tensor with shape [3, 4].*
3	[D0, D1, D2]	3-D	*A 3-D tensor with shape [1, 4, 3].*
n	[D0, D1, … Dn-1]	n-D	*A tensor with shape [D0, D1, ... Dn-1].*

Type: kiểu dữ liệu.

Bảng 3.6: Mô tả kiểu dữ liệu trong tensorflow [23]

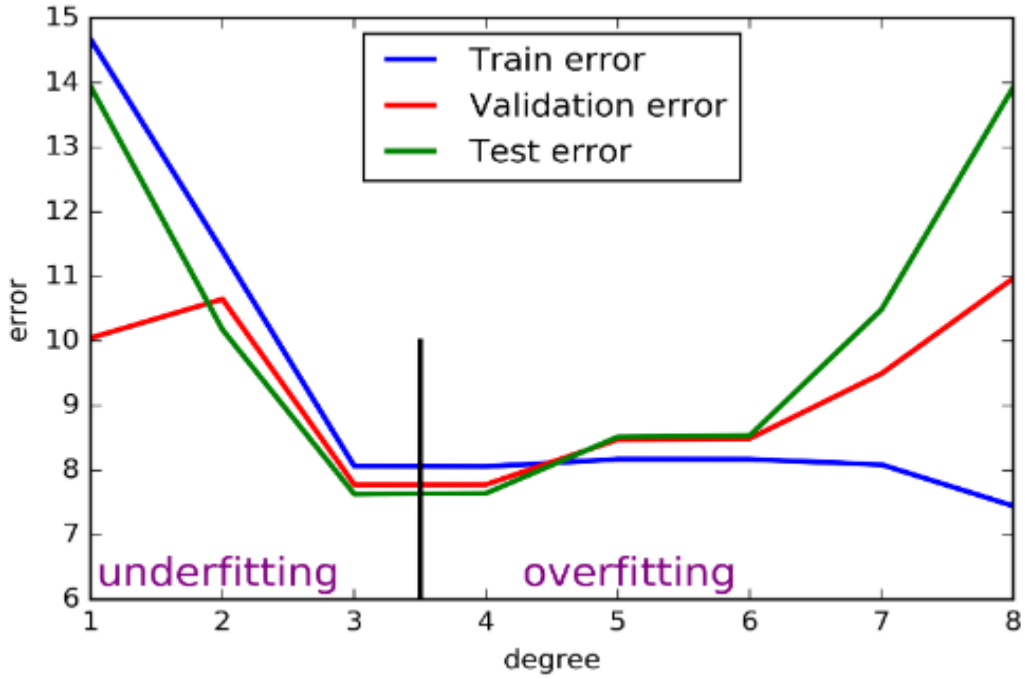
Loại dữ liệu	Loại python	Mô tả
DT_FLOAT	tf.float32	*32 bits floating point.*
DT_DOUBLE	tf.float64	*64 bits floating point.*
DT_INT8	tf.int8	*8 bits signed integer.*
DT_INT16	tf.int16	*16 bits signed integer.*
DT_INT32	tf.int32	*32 bits signed integer.*
DT_INT64	tf.int64	*64 bits signed integer.*
DT_UINT8	tf.uint8	*8 bits unsigned integer.*
DT_STRING	tf.string	*Variable length byte arrays. Each element of a Tensor is a byte array.*
DT_BOOL	tf.bool	*Boolean.*
DT_COMPLEX64	tf.complex64	*Complex number made of two 32 bits floating points: real and imaginary parts.*
DT_QINT8	tf.qint8	*8 bits signed integer used in quantized Ops.*
DT_QINT32	tf.qint32	*32 bits signed integer used in quantized Ops.*
DT_QUINT8	tf.quint8	*8 bits unsigned integer used in quantized Ops.*

Variable: lưu trạng thái (state) sau khi tính toán graph.

3.3. Phương pháp thực nghiệm:

3.3.1. Cách chia dữ liệu:

Với việc chia tập dữ liệu ra thành hai tập nhỏ: training data và test data. Phương pháp sẽ là trích từ tập training data ra một tập con nhỏ và thực hiện việc đánh giá mô hình trên tập con nhỏ này. Tập con nhỏ được trích ra từ training set này được gọi là validation set. Lúc này, training set là phần còn lại của training set ban đầu**.** Train error được tính trên training set mới này, và có một khái niệm nữa được định nghĩa tương tự như trên validation error, tức error được tính trên tập validation.



Hình 3.2: Lựa chọn mô hình dựa trên validation[26]

3.3.2. Cách thức đánh giá:

Để đánh giá hiệu quả phân lớp luận văn sử dụng các độ đo F-score. Giá trị F-Score phụ thuộc vào Precision và Recall. Trong đó, Precision là độ đo thể hiện độ chính xác của bộ phân lớp, được xác định bằng số bình luận được phân lớp đúng trên tổng số bình luận được phân vào lớp đó. Recall là độ đo thể hiện khả năng không phân lớp sai các bình luận, được xác định bằng số bình luận được phân lớp đúng trên tổng số bình luận thực tế thuộc lớp đó. F-score là độ được xác định thông qua Precision và Recall (giá trị của các độ đo này càng cao thì bộ phân lớp càng có hiệu quả phân lớp tốt). Cụ thể:

$$\Pr ecision = \frac{TP}{TP + FP}$$

$$\Re call = \frac{TP}{TP + FN}$$

$$F - score = \frac{2 \times \Pr ecison \times \Re call}{\Pr ecison + \Re call}$$

3.4. Tiến hành thực nghiệm

3.4.1. Xây dựng các thành phần chung cho các mô hình:

- **Xây dựng class và khai báo tham số đầu vào cho các mô hình mạng:**

Mỗi mô hình mạng neural sẽ được khai báo bởi 1 class với hàm init để khởi tạo các tham số đầu vào của mạng. Các tham số đầu vào cho mô hình mạng:

- ***Định nghĩa dữ liệu đầu vào để truyền cho mô hình mạng***

tf.placeholder: Tạo ra một biến vị trí để đưa vào mạng khi thực hiện huấn luyện hay được kiểm thử. Tham số thứ 2 là hình của bộ dữ liệu đầu vào, sử dụng **None** cho phép mạng có thể xử lý các lô kích thước tùy ý.

- **Lớp nhúng (Embedding layer)**

tf.device('/cpu:0'): Buộc một hoạt động được thực hiện trên CPU. Mặc định TensorFlow sẽ cố gắng đưa hoạt động trên GPU nếu có sẵn.

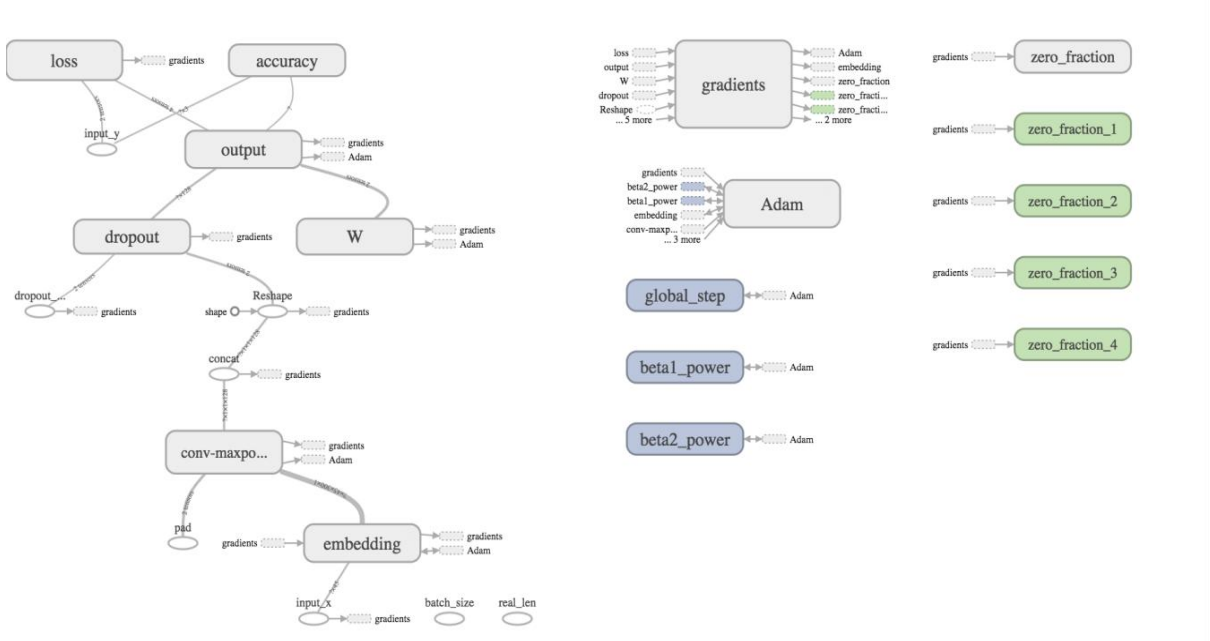
- **Tính tỉ lệ lỗi và độ chính xác:**

3.4.2. Mã lệnh cài đặt các mô hình bằng ngôn ngữ Python trên Tensorflow:

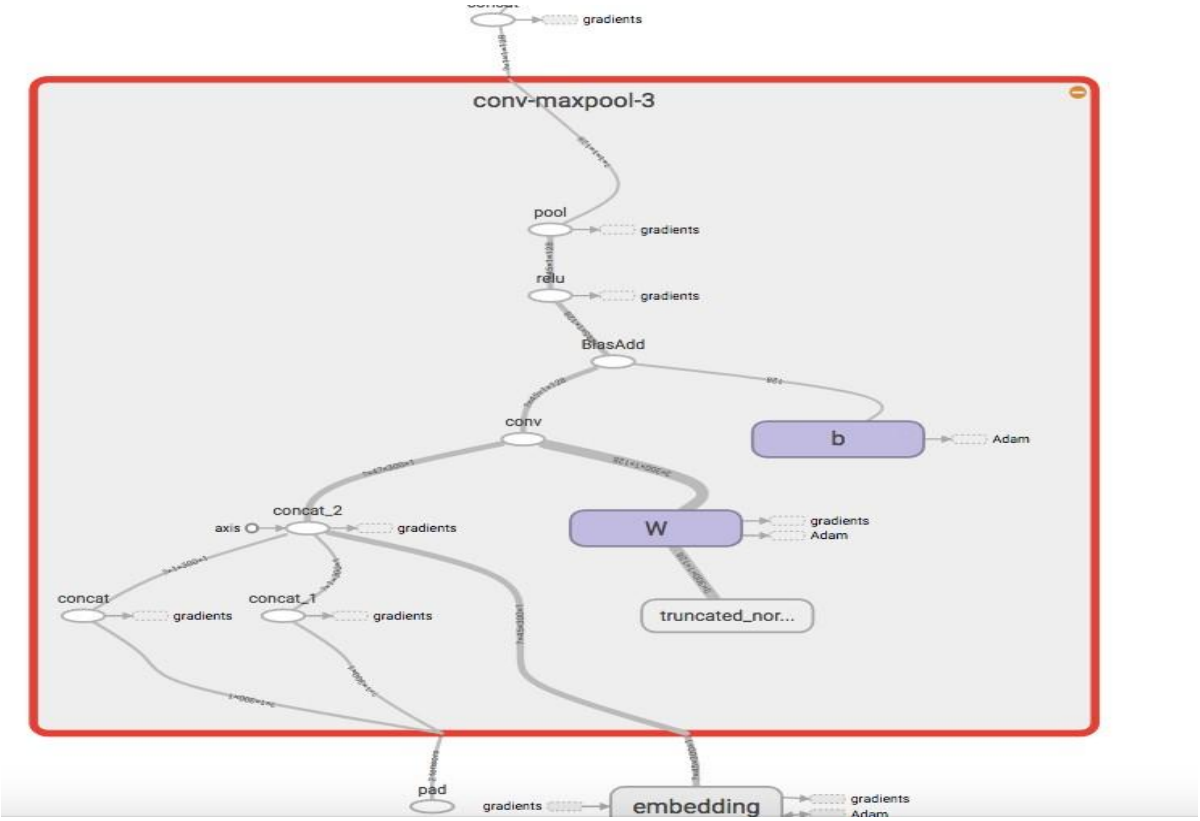
3.4.2.1. Mô hình mạng neural CNN (Lớp tích chập và max-pooling)

Xây dựng lớp tích chập cho CNN và theo sau đó là max-pooling sử dụng bộ lọc có kích thước bằng 3.

Mô hình mạng trong TensorBoard:



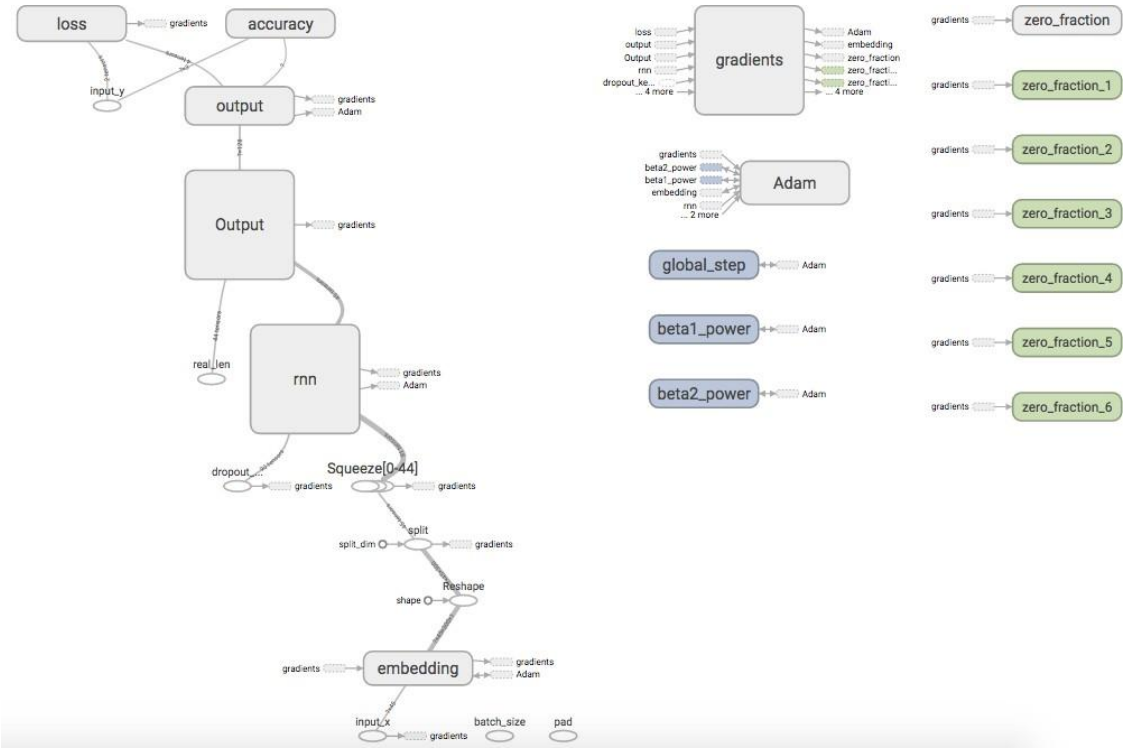
Hình 3.3: Mô hình mạng CNN trong nghiên cứu.



Hình 3.4: Mô hình conv-maxpool trong mạng CNN

3.4.2.2. Mô hình mạng neural RNN (Lớp ẩn sử dụng GRU cell)

Mô hình mạng trong TensorBoard:



Hình 3.5: Mô hình mạng RNN nghiên cứu

3.4.2.3. Mô hình LSTM:

3.5. Kết quả chạy thực nghiệm

Kết quả với mô hình CNN:

Bảng 3.7: Kết quả sử dụng mô hình CNN

CNN									
Epoch	5			10			20		
Batch size	32			64			128		
Accuracy %	71.16			74.25			81.52		
Độ đo	***Pre%***	***Re%***	***F1%***	***Pre%***	***Re%***	***F1%***	***Pre%***	***Re%***	***F1%***
Gợi ý	77.58	70.20	73.71	80.98	78.32	79.63	84.07	79.16	81.45
Không gợi ý	77.32	72.03	74.62	77.99	78.67	78.33	88.91	82.42	83.04

❖ ***Kết quả với mô hình RNN:***

Bảng 3.8: Kết quả sử dụng mô hình RNN

RNN									
Epoch	5			10			20		
Batch size	32			64			128		
Accuracy %	69.46			71.28			76.81		
Độ đo	***Pre%***	***Re%***	***F1%***	***Pre%***	***Re%***	***F1%***	***Pre%***	***Re%***	***F1%***
Gợi ý	69.49	60.67	64.81	80.76	67.02	73.03	82.35	73.68	77.81
Không gợi ý	54.6	89.0	67.7	76.92	66.56	71.42	72.72	86.02	78.88

❖ ***Kết quả với mô hình LSTM:***

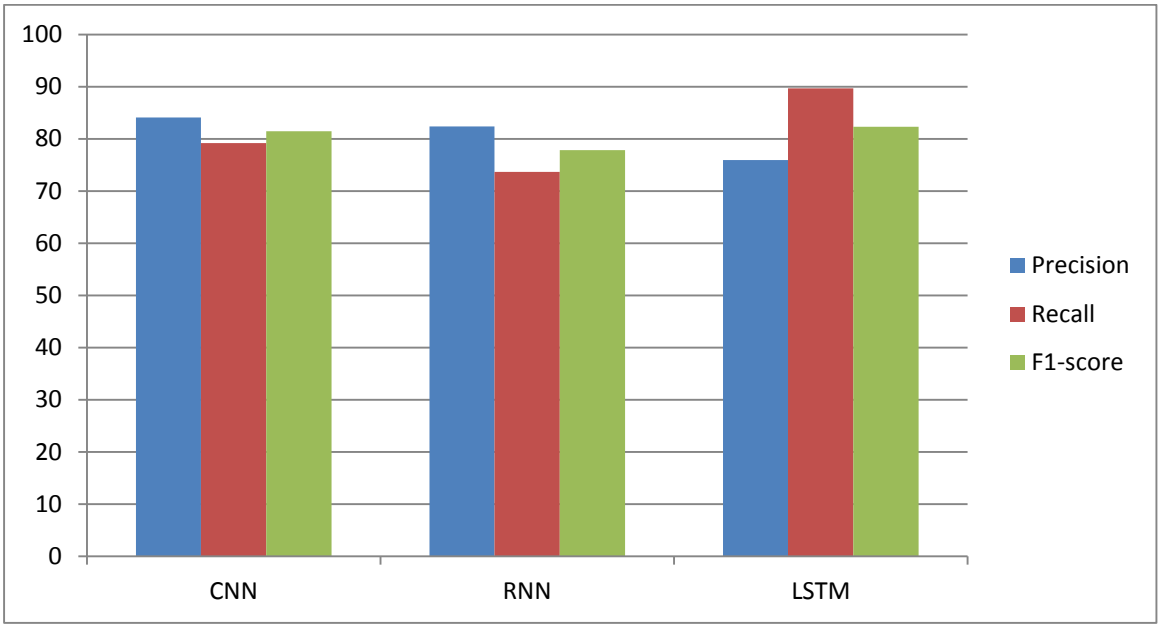
Bảng 3.9: Kết quả sử dụng mô hình LSTM

LSTM									
Epoch	5			10			20		
Batch size	32			64			128		
Accuracy %	72.42			75.07			83.26		
Độ đo	***Pre%***	***Re%***	***F1%***	***Pre%***	***Re%***	***F1%***	***Pre%***	***Re%***	***F1%***
Gợi ý	68.18	80.03	73.61	72.72	86.02	78.81	75.93	89.70	82.29
Không gợi ý	66.29	83.09	73.75	72.47	88.76	79.76	83.57	86.22	84.87

Bảng 3.10: Kết quả so sánh giữa các mô hình

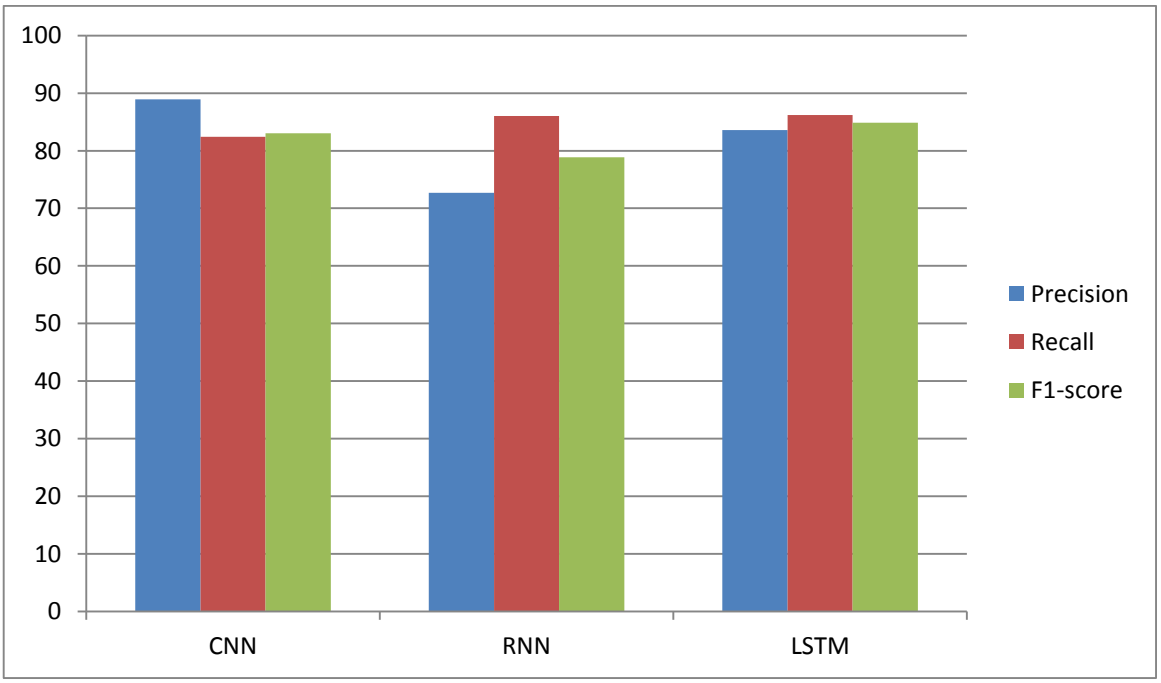
Mô hình	**CNN**			**RNN**			**LSTM**		
Accuracy %	81.52			76.81			83.26		
Độ đo	***Pre%***	***Re%***	***F1%***	***Pre%***	***Re%***	***F1%***	***Pre%***	***Re%***	***F1%***
Gợi ý	84.07	79.16	81.45	82.35	73.68	77.81	75.93	89.70	82.29
Không gợi ý	88.91	82.42	83.04	72.72	86.02	78.88	83.57	86.22	84.87

Sau khi tiến hành thực nghiệm cho kết quả giữa thuật toán CNN,RNN và LSTM tôi thực hiện so sánh kết quả của các phương pháp theo từng nhãn *" Gợi ý", "Không gợi ý"* được biểu diễn bằng các biểu đồ sau:



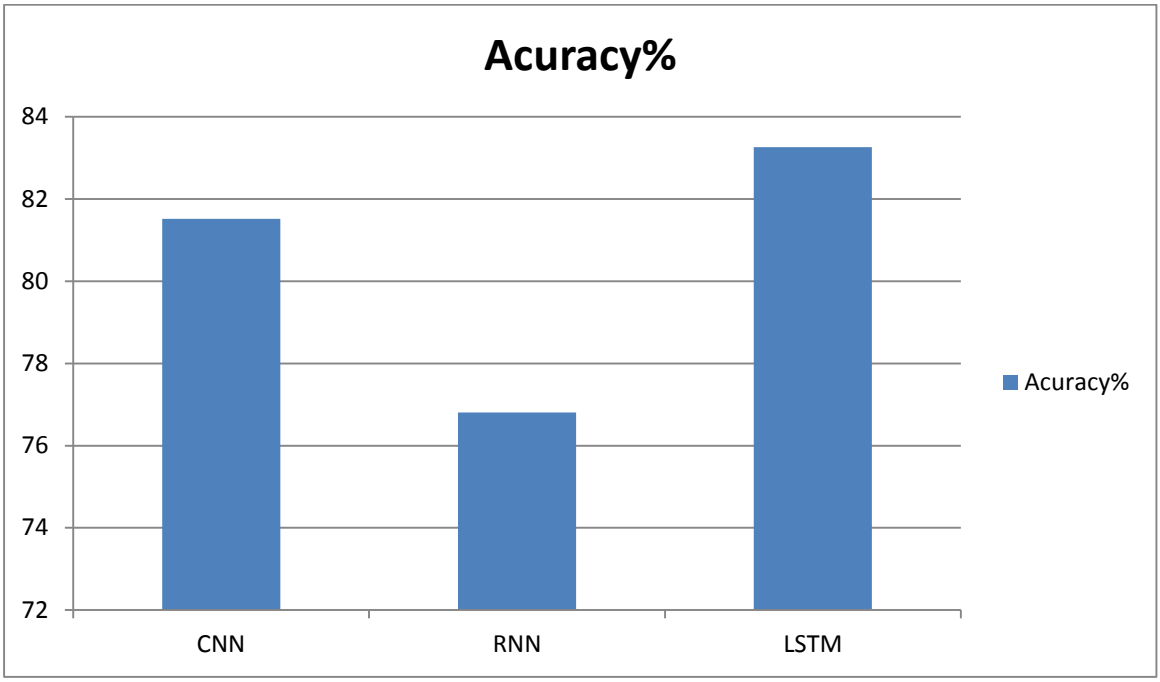
Hình 3.6: Biểu đồ so sánh giữa mô hình CNN, RNN, LSTM với nhãn "Gợi ý"

➢ Với nhãn " ***Gợi ý***" ta có thể thấy kết quả khả quan nếu sử dụng mô hình LSTM



Hình 3.7: Biểu đồ so sánh giữa mô hình CNN, RNN, LSTM với nhãn "Không gợi ý"

➢ Với nhãn " **Không *gợi ý***" ta có thể thấy kết quả khả quan nếu sử dụng mô hình LSTM



Hình 3.6: Biểu đồ so sánh độ chính xác của các mô hình

Từ bảng kết quả hình 3.8 và bảng 3.10, tôi thấy rằng :

3.6. Nhận xét và đánh giá

Dựa vào các số liệu trên, kết quả trên bộ ngữ liệu tiếng Anh là khá tốt, kết quả khi sử dụng model LSTM cho kết quả tốt hơn so với các thuật toán CNN, RNN để lựa chọn mô hình áp dụng cho đề tài " ***Phát hiện câu chứa gợi ý trên diễn đàn trực tuyến sử dụng mạng No-Ron"***, tôi đề xuất và đánh giá cao mô hình LSTM hơn cả. Bên cạnh đó, các số liệu trung bình cũng như độ chênh lệch độ chính xác của mô hình LSTM cho kết quả khả quan nhất.

Tóm lại các mô hình mạng neural CNN và RNN, LSTM cho thấy một cách nhìn mới trong việc phân loại câu văn bản nói riêng và xử lý ngôn ngữ tự nhiên nói chung, bằng cách sử dụng học chuyên sâu và kết hợp mô hình mạng neural. Luận văn cũng đã đưa ra các nhận xét, đánh giá và so sánh các mô hình, các bộ phân lớp, từ đó đưa ra được một mô hình tốt nhất trong việc giải quyết bài toán phân loại câu chưa gợi ý người dùng trên diễn đàn trực tuyến đã nêu.

KẾT LUẬN

Xữ Lý Ngôn Ngữ Tự Nhiên nói chung và đặc biệt là phân loại câu chứa gợi ý người dùng nói riêng ngày càng đóng vai trò quan trọng trong các hoạt động thương mại, mua bán,du lịch… hiện nay.Trong luận văn này, chúng tôi tiến hành nghiên cứu phương pháp nhằm cải thiện độ chính xác cho bài toán phân loại câu văn bản, cụ thể là cải thiện độ chính xác cho bài toán phân loại câu chứa gợi ý trên diễn đàn trực tuyến. Bài toán này được xác định là một bài toán có độ phức tạp và có nhiều ứng dụng trong thực tế. Phương pháp giải quyết của luận văn tập trung vào việc nâng cao độ chính xác trong việc phân loại được các ý định của người dùng thông qua diễn đàn trực tuyến. Bằng việc sử dụng mô hình phân lớp quen thuộc CNN và RNN cùng với tập dữ liệu thu được từ diễn đàn trực tuyến, luận văn đã đưa ra số phương pháp để giải quyết cho bài toán đề ra. Quá trình thực nghiệm đạt được kết quả khả quan, cho thấy tính đúng đắn của việc lựa chọn cũng như kết hợp các phương pháp, đồng thời hứa hẹn nhiều tiềm năng phát triển hoàn thiện.

Nhìn chung, luận văn đã đạt được một số kết quả như:

- Trình bày một cách khái quát, tổng quan nhất và nêu lên ý nghĩa, vai trò quan trọng của bài toán phân loại câu chứa gợi ý người dùng trên diễn đàn trực tuyến.
- Nghiên cứu các mô hình khác nhau cho bài toán phân loại câu chứa gợi ý.
- Nghiên cứu và làm thực nghiệm với các thuật toán học máy khác nhau.
- So sánh và phân tích các kết quả thực nghiệm, đưa ra kết quả tốt nhất.

Luận văn vẫn còn một số hạn chế như:

- Nghiên cứu dựa trên số lượng dữ liệu còn ít và chưa đầy đủ.
- Kết quả thực nghiệm đạt được vẫn chưa thực sự cao
- Chỉ thử nghiệm đối với tập dữ liệu bằng tiếng anh

Về hướng phát triển tương lai, chúng tôi sẽ tiến hành thu thập và phát triển trên một tập dữ liệu lớn hơn và dựa trên nhiều đặc trưng hơn để góp phần cải thiện khả năng phân loại. Bên cạnh đó chúng tôi cũng sẽ nghiên cứu và thử nghiệm với một số thuật toán khác để tìm ra thuật toán phù hợp nhất với bài toán phân loại câu chứa gợi ý người dùng trực tuyến bằng tiếng Việt. Khắc phục lỗi trong quá trình xử lý để nâng cao kết quả thực nghiệm.