BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC UEH - TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ
KHOA CÔNG NGHỆ THÔNG TIN KINH DOANH
CHUYÊN NGÀNH: Khoa học dữ liệu

Khóa luận tốt nghiệp

XÂY DỰNG HỆ THỐNG PHÂN LOẠI CHỦ ĐỀ TIN TỨC TIẾNG VIỆT TỰ ĐỘNG VỚI CÁC KỸ THUẬT HỌC MÁY VÀ HỌC SÂU

Họ tên sinh viên: Nguyễn Minh Nhật
Mã sinh viên: 31191025431
Lớp: DS001 Khóa: 46
Họ tên giáo viên hướng dẫn: Đặng Ngọc Hoàng Thành

Niên khóa: 2020 - 2023
Tp Hồ Chí Minh, ngày 23 tháng 7 năm 2023

LỜI CÁM ƠN

Đầu tiên, em xin chân thành được bày tỏ lời cảm ơn đến thầy Đặng Ngọc Hoàng Thành. Thầy đã tận tâm hướng dẫn, giúp đỡ cũng như đưa ra những góp ý cần thiết trong suốt thời gian qua. Thầy cũng là người hỗ trợ em với những ý tưởng cũng như đảm bảo tính phù hợp của luận văn, qua đó có thể giúp em hoàn thiện khóa luận một cách trọn vẹn.

Ngoài ra, em cũng xin cảm ơn Ban lãnh đạo Trường Đại học UEH cũng như các phòng ban của trường đã tạo điều kiện, cơ sở vật chất để em có cơ hội và môi trường học tập và rèn luyện.

Bên cạnh đó, trong suốt quá trình học tập tại trường Công nghệ và Thiết kế - Đại học UEH, em đã nhận được rất nhiều sự hỗ trợ, giúp đỡ từ các thầy cô bộ môn, chính vì vậy nên em xin đươc gửi lời cảm ơn đến tập thể các thầy cô công tác Đại học UEH nói chung và khoa công nghệ thông tin kinh doanh nói riêng. Em cám ơn các thầy cô vì đã luôn tận tâm chỉ bảo cũng như tạo điều kiện tốt nhất cho sinh viên trong suốt quá trình học tập, nghiên cứu tại trường. Những kiến thức mà chúng em nhận được sẽ là hành trang giúp chúng em vững bước, tự tin hơn trong tương lai.

Cuối cùng, em cũng xin cám ơn gia đình và bạn bè - Những người đã hết sức ủng hộ, giúp đỡ và động viên em trong suốt quá trình học tập đã qua.

Mục lục

DANH MỤC THUẬT NGỮ VÀ CHỮ VIẾT TẮT

STT	Thuật ngữ	Chữ viết tắt	Diễn giải
1	Single class	-	Phân loại đơn lớp
2	Multi class	-	Phân loại đa lớp
3	Feature	-	Đặc trưng
4	Feature Extraction	-	Trích xuất đặc trưng
5	Term Frequency	TF	Tần suất thuật ngữ
6	Inverse Document Frequency	IDF	Tần suất tài liệu nghịch đảo
7	Global Vectors for Word Representation	Glove	Vector toàn cục để biểu diễn từ
8	Word Embedding	-	Nhúng từ
9	Weighted Word	-	Lấy trọng số từ
10	Principle Component Analysis	PCA	Phân tích thành phần chính
11	Linear Discriminant Analysis	LDA	Phân tích phân biệt tuyến tính
12	Non-negative Matrix Factorization	NMF	Phân tích hệ số ma trận không âm
13	K Nearest Neighbor	kNN	Mô hình k láng giềng gần nhất
14	Support Vector Machine	SVM	Máy Vector Hỗ trợ
15	Accuracy	-	Độ chuẩn xác
16	Precision	-	Độ chính xác
17	Recall	-	Độ phủ
18	F-Measure	-	Thước đo F
19	Decision Tree	-	Mô hình cây quyết định
20	Random Forest	RF	Mô hình rừng ngẫu nhiên
21	Stop words	-	Từ dừng
22	Token	-	Đơn vị văn bản
23	Text Tokenization	-	Phân tách văn bản
24	Bag of Words	BoW	Mô hình túi từ

25	Continuos Bag of Words	CBoW	Mô hình túi từ liên tục
26	Singular Value Decomposition	SVD	Phương pháp phân tích suy biến
27	Atrribute Selection Measure	ASM	phép đo lựa chọn thuộc tính
28	Deep Learning	DL	Học sâu
29	Multi Layer Perceptron	MLP	Perceptron đa lớp
31	Recurrent Neural Network	RNN	Mạng nơron hồi tiếp
33	Convolutional Neural Networ	CNN	Mạng Nơron tích chập
34	Long-Short Term Memory	LSTM	Mạng bộ nhớ ngắn-dài hạn
35	Neural Collaborative Filtering	NCF	Lọc Nơron cộng tác
36	True Postive	TP	Số lượng dự đoán chính xác
37	True Negative	TN	Số lượng dự đoán chính xác gián tiếp
38	False Positve	FP	Sai lầm loại I
39	False Negative	FN	Sai lầm loại II

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG BIỂU

MỞ ĐẦU

1. Lý do lựa chọn đề tài

Sự tiến bộ vượt bậc của công nghệ nói chung và mạng Internet nói riêng đã tạo điều kiện thuận lợi để mọi người từ khắp nơi có thể tiếp cận với các nguồn thông tin nhanh chóng. Một trong những nguồn thông tin quan trọng chính là các bài báo điện tử với đa dạng thể loại khác nhau. Vì lẽ đó nên số lượng các bài báo điện tử được phát hành và lưu trữ trên các nền tảng số là ngày càng lớn, từ đó đặt ra nhu cầu cấp thiết trong việc phát triển các hệ thống phân loại chủ đề tự động để người dùng có thể tiếp cận nguồn thông tin dễ dàng hơn. Chính vì vậy, đề tài này được thực hiện nhằm xây dựng hệ thống phân loại chủ đề tin tức tiếng việt tự động, với nhiệm vụ chính là phân tích dữ liệu văn bản từ các bài báo điện tử để từ đó có thể mô hình hóa và phân loại chủ đề một cách nhanh chóng và tự động.

2. Mục tiêu nghiên cứu

Đề tài gồm những mục tiêu nghiên cứu cụ thể sau:

- Tìm hiểu, hệ thống hóa các khái niệm liên quan đến hệ thống phân loại chủ đề tin tức.
- Tập trung vào giải quyết bài toán phân loại chủ đề tin tức tự động với các phương pháp học máy.
- Ứng dụng các kỹ thuật tiền xử lý và biểu diễn văn bản đối với ngôn ngữ tiếng Việt nhằm mô hình hóa dữ liệu tin tức trực tuyến.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu của đề tài bao gồm:

+ Các khái niệm liên quan đến hệ thống phân loại văn bản và các phương pháp xử lý, khai phá dữ liệu văn bản

+ Các hướng tiếp cận để xây dựng và đánh giá hệ thống phân loại chủ đề tin tức tự động.

4. Phương pháp nghiên cứu

Hai phương pháp nghiên cứu chính được thực hiện trong đề tài là: phương pháp phân tích tổng hợp và thu thập thông tin từ nguồn dữ liệu thứ cấp.

NỘI DUNG

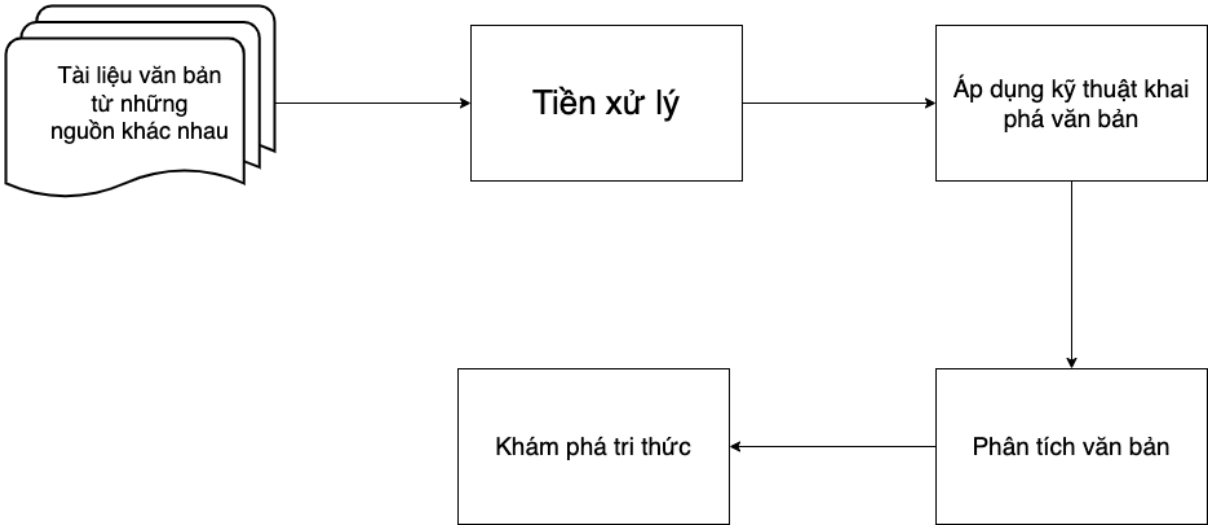
Chương 1. Tổng quan về hệ thống phân loại chủ đề tin tức

1.1. Khai phá dữ liệu văn bản

Ngày nay với sự phát triển của khoa học và công nghệ, sự tăng trưởng theo cấp số nhân của lượng dữ liệu được tạo ra từ các hoạt động hằng ngày của con người là điều không thể bàn cãi. Chính vì vậy, việc thu thập, lưu trữ, xử lý một lượng lớn dữ liệu để trích xuất những thông tin hữu ích đã trở thành nhu cầu tất yếu của doanh nghiệp, qua đó hỗ trợ doanh nghiệp có thể đưa ra những quyết định kinh doanh một cách nhanh chóng và chuẩn xác. Tuy nhiên, đa phần dữ liệu được lưu trữ trong các hệ thống điện tử đều là dạng dữ liệu phi cấu trúc, chẳng hạn như hình ảnh, đoạn phim, âm thanh hoặc văn bản. Do đó, rất khó để trích xuất thông tin bằng cách sử dụng các kỹ thuật khai thác dữ liệu chính thống vì chúng không thể xử lý dữ liệu phi cấu trúc một cách hiệu quả.

Khai phá văn bản hay còn được biết đến là khám phá tri thức từ cơ sở dữ liệu văn bản, đề cập đến quá trình trích xuất các mẫu hoặc kiến thức thú vị và không tầm thường (non-trivial) từ tài liệu văn bản. Các kỹ thuật trích xuất văn bản như trích xuất, truy hồi thông tin, mô hình hóa chủ đề (topic modelling), phân loại chủ đề (topic classification), tổng kết văn bản (summarization) giúp việc khai thác một loại dữ liệu phi cấu trúc như văn bản trở nên dễ dàng và hiệu quả hơn.

Quá trình khai phá dữ liệu văn bản bắt đầu bằng việc thu thập tài liệu văn bản từ các nguồn khác nhau. Công cụ khai thác dữ liệu văn bản sẽ truy xuất một tài liệu cụ thể và tiến hành xử lý tài liệu này bằng cách kiểm tra định dạng và bộ ký tự. Sau đó, tài liệu sẽ trải qua giai đoạn phân tích văn bản. Ở giai đoạn này văn bản sẽ được phân tích ngữ nghĩa nhằm mục đích rút trích được thông tin chất lượng cao từ văn bản. Tùy thuộc vào mục tiêu của tổ chức, các kỹ thuật phân tích văn bản khác nhau có thể được kết hợp và sử dụng. Đôi khi quá trình phân tích văn bản được lặp lại liên tục cho đến khi thông tin được trích xuất. Kết quả trích xuất sau đó có thể được đặt trong một hệ thống thông tin quản lý, mang lại một lượng kiến thức phong phú cho người sử dụng hệ thống đó.



Hình 1: Quy trình khai phá dữ liệu văn bản

Các ứng dụng khai thác dữ liệu văn bản đang dần tạo ra giá trị trong việc cải thiện hiệu suất kinh doanh và thúc đẩy tăng trưởng doanh nghiệp. Khai thác văn bản giúp doanh nghiệp tìm thấy thông tin hiệu quả có thể được sử dụng bằng cách sàng lọc dữ liệu lớn thu được từ phương tiện truyền thông xã hội, vlog, vé hỗ trợ khách hàng, phản hồi của nhân viên, dữ liệu phản hồi của khách hàng và các nguồn khác.

Các công cụ tri thức kinh doanh tiếp nhận dữ liệu văn bản đã được xử lý, phân tíc và sử dụng những dữ liệu đó để tạo ra các báo cáo, nhằm hỗ trợ việc ra quyết định dựa trên dữ liệu trở nên dễ dàng hơn. Những phát hiện được rút ra từ việc so sánh dữ liệu lịch sử và dữ liệu hiện tại có thể giúp tìm ra xu hướng, tầm ảnh hưởng theo thời gian thực về cách một sự kiện đã ảnh hưởng đến thương hiệu hoặc thông tin quan trọng khác. Hơn nữa, khi được thực hiện trên quy mô lớn và được áp dụng trên toàn tổ chức, hệ thống khai thác dữ liệu văn bản có thể được ứng dụng trong các chiến lược tiếp thị tổng hợp bao gồm nuôi dưỡng quan hệ khách hàng và quản lý danh tiếng thương hiệu.

1.2. Hệ thống phân loại chủ đề tin tức

1.2.1. Tổng quan về bài toán phân loại chủ đề tin tức

Từ lâu, tin tức đã luôn là một phương tiện truyền thông và giao tiếp quan trọng trong đời sống xã hội loài người. Về mặt định nghĩa, có thể xem tin tức như là một tập hợp thông tin về các sự kiện trong một khoảng thời gian gần so với một thời điểm nhất định. Những sự kiện này có thể thuộc bất kỳ loại nào. Hiện nay, với sự phát triển vượt bậc của khoa học và công nghệ cũng như sự ra đời của mạng internet, tin tức có thể dễ dàng được tiếp cận khắp nơi với đa dạng các chủ đề và luôn được cập nhật một cách

nhanh chóng. Vì lẽ đó nên số lượng bài báo được lưu trữ và thông hành trên các nền tảng kỹ thuật số ngày càng tăng và cần có các phương pháp, kỹ thuật mới để sắp xếp, phân loại một cách nhanh chóng, hiệu quả. Việc phân loại thủ công truyền thống các văn bản tin tức thường tốn thời gian và tiêu tốn rất nhiều nguồn tài nguyên nhân lực và tài chính, do đó khó mà đáp ứng các nhu cầu về tìm kiếm và truy xuất thông tin một cách hiệu quả.

Hệ thống phân loại chủ đề tin tức tự động đã được nghiên cứu và phát triển nhằm giải quyết các thách thức như trên. Phân loại chủ đề tin tức thuộc nhóm bài toán phân loại văn bản - một trong những kỹ thuật khai phá dữ liệu văn bản quan trọng và được ứng dụng rộng rãi. Phân loại văn bản có thể được định nghĩa là hành động gắn nhãn tự động một tài liệu văn bản, dựa trên một tập hợp các danh mục cho trước. Cụ thể hơn, cho trước một tập N tài liệu D = $\{d_1, d_2, \ldots, d_n\}$ và một tập k danh mục được định nghĩa từ trước C = $\{c_1, c_2, \ldots, c_n\}$, một bộ phân loại văn bản sẽ ánh xạ chính xác một tài liệu/văn bản d_i vào một chủ đề thích hợp c_j. Như vậy đối với bài toán phân loại tin tức, tập tài liệu sẽ là các bài báo điện tử được thu thập từ các trang báo tiếng việt trực tuyến, đi kèm với những bài báo này sẽ là các chủ đề tương ứng.

Thông thường, dựa trên số lượng chủ đề tương ứng với từng bài báo, bài toán phân loại tin tức có thể được chia ra 2 loại chính là phân loại đơn lớp (single class) và phân loại đa lớp (multi class). Đối với phân loại đơn lớp, một bài báo $d_k \in$ D chỉ gắn với một chủ đề $c_j \in$ C. Đối với phân loại đa lớp, một bài báo có thể thuộc vào $c_j \in$ C chủ đề khác nhau, với 0 < nj ≤ |C|. Luận văn này sẽ tập trung vào vấn đề phân loại đơn lớp đối với các bài báo tiếng việt.

1.2.2. Quy trình hoạt động của hệ thống phân loại chủ đề tin tức

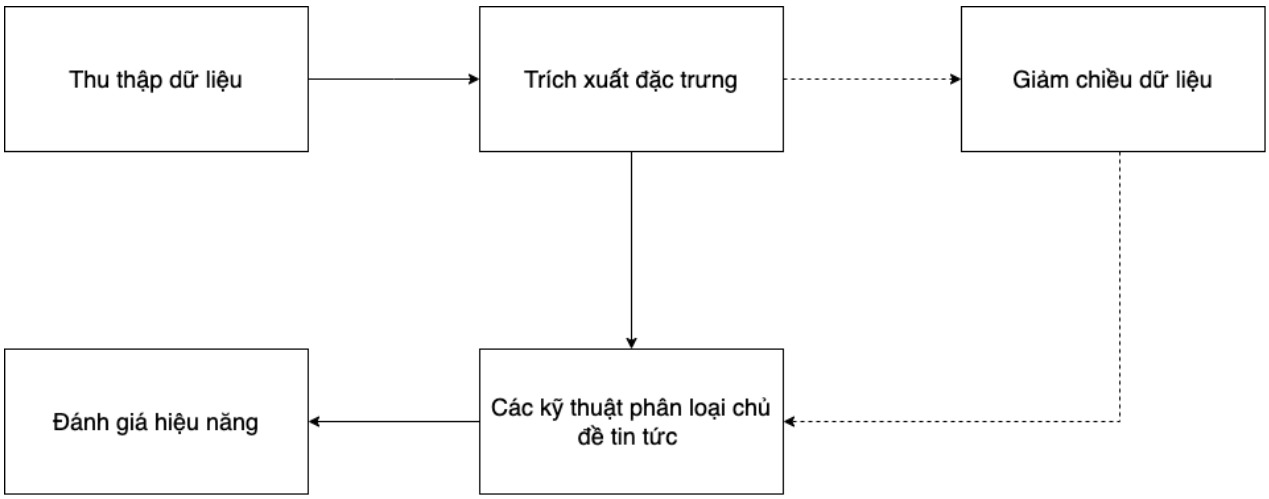
1.2.2.1. Thu thập dữ liệu

Nhìn chung, hệ thống phân loại văn bản có thể được áp dụng cho bốn mức độ khác nhau về phạm vi của văn bản, bao gồm:

- Cấp độ tài liệu: Ở cấp độ tài liệu, hệ thống phân loại được được các chủ đề liên quan của toàn bộ tài liệu.
- Cấp độ đoạn văn: Ở cấp độ đoạn văn, hệ thống phân loại được được các chủ đề liên quan của một đoạn văn (một phần của tài liệu).

- Cấp độ câu: Ở cấp độ câu, có được các phạm trù liên quan của câu đơn (một phần của một đoạn văn).
- Cấp độ câu phụ: Ở cấp độ câu phụ, thuật toán thu được các phạm trù liên quan của biểu thức phụ trong câu (một phần của câu)).

Đầu vào khởi tạo của một hệ thống phân loại tin tức bắt đầu một số tập dữ liệu văn bản thô. Những tập văn bản này có thể được thu thập từ nhiều nguồn khác nhau, chẳng hạn như từ các trang báo điện tử hoặc mạng xã hội. Mỗi một tài liệu thuộc tập văn bản này có thể chứa một số lượng câu nhất định, mỗi câu lại chứa một số lượng từ và mỗi từ lại chứa một số lượng chữ cái tương ứng. Đồng thời, mỗi tài liệu này sẽ gắn với một chủ đề/lớp tương ứng. Sau đó tập văn bản này cùng với bộ các chủ đề tương ứng có thể được lưu trữ dưới dạng bảng (tabular) nhằm phục vụ cho các bước tiếp theo. Bên cạnh đó, mỗi tài liệu có thể được áp dụng các phương pháp tiền xử lý dữ liệu văn bản cần thiết như chuyển đổi chữ hoa thành chữ thường, loại bỏ các ký tự đặc biệt,...



Hình 2: Quy trình xây dựng hệ thống phân loại chủ đề tin tức

1.2.2.2. Trích xuất đặc trưng

Mục tiêu của giai đoạn trích xuất đặc trưng trong hệ thống phân loại chủ đề tin tức là để chuyển đổi dạng dữ liệu phi cấu trúc như văn bản thành không gian đặc trưng có cấu trúc. Nhờ đó mà chúng ta mới có thể áp dụng các mô hình toán học lên tập dữ liệu và xây dựng được bộ phân loại. Đầu tiên, dữ liệu cần được làm sạch để lược bỏ những ký tự, từ không cần thiết. Sau khi dữ liệu đã được làm sạch, các phương pháp trích xuất đặc trưng chính thức có thể được áp dụng. Các kỹ thuật phổ biến của trích xuất tính năng là Tần suất tài liệu nghịch đảo thuật ngữ (TF-IDF), Tần suất thuật ngữ (TF)

(Gerard & Christopher, 1988), Word2Vec (Yoav & Omer, word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method, 2014) và Vector toàn cục để biểu diễn từ (GloVe) (Pennington, Socher, & Manning, 2014). Các kỹ thuật trích xuất đặc trưng có thể được phân thành 2 nhóm chính là nhúng từ (Word Embedding) hoặc lấy trọng số từ (Weighted Word).

1.2.2.3. Giảm chiều dữ liệu

Chiều của dữ liệu có thể được hiểu là số lượng của các biến/đặc trưng đo được trên mỗi quan sát. Các bộ dữ liệu đa chiều với kích thước lớn luôn ẩn chứa nhiều tiềm năng về mặt giá trị có thể được khai thác và nghiên cứu, tuy nhiên những bộ dữ liệu này cũng đặt ra nhiều thách thức về mặt tính toán và thường cần đến những kỹ thuật xử lý tinh vi, phức tạp hơn. Một trong những vấn đề với bộ dữ liệu đa chiều là trong nhiều trường hợp, không phải tất cả các biến hay các cột dữ liệu đều có chất lượng tốt và đóng góp vào hiệu suất của mô hình. Mặc dù một số phương pháp mới tốn kém về mặt tính toán nhất định có thể xây dựng các mô hình dự đoán với độ chính xác cao từ dữ liệu nhiều chiều, nhưng nhiều ứng dụng vẫn quan tâm đến việc giảm kích thước của dữ liệu gốc trước khi mô hình hóa bất kỳ dữ liệu nào.

Tác vụ phân loại văn bản nói chung và phân loại tin tức nói riêng thường phải làm việc với các tập dữ liệu lớn với một khối lượng lớn từ vựng, do đó các kỹ thuật giảm chiều dữ liệu có vai trò quan trọng trong quá trình xử lý dữ liệu và xây dựng mô hình. Giảm chiều dữ liệu giúp giải quyết vấn đề tính toán không hiệu quả cũng giúp phát hiện và khai thác mối quan hệ giữa các từ trong tài liệu. Sau khi nắm được mối quan hệ giữa các từ khóa, việc phân loại tài liệu có thể được thực hiện rất hiệu quả và dễ dàng. Thời gian xử lý sẽ được giảm xuống. Các kỹ thuật giảm chiều dữ liệu phổ biến nhất bao gồm Phân tích thành phần chính (PCA), Phân tích phân biệt tuyến tính (LDA) và phân tích hệ số ma trận không âm (NMF).

1.2.2.4. Các kỹ thuật phân loại chủ đề tin tức

Bước quan trọng nhất của quy trình phân loại chủ đề tin tức là chọn bộ phân loại tối ưu nhất. Những bộ phân loại này được xây dựng dựa trên các kỹ thuật học máy có giám sát, trong đó với một bộ dữ liệu là tập các tin tức đã được phân chia thành các chủ đề khác nhau sẽ được sử dụng như đầu vào cho các mô hình. Nhiệm vụ của những

mô hình này là nhận dạng các khuôn mẫu (pattern) mô tả mối quan hệ giữa nhóm từ khóa trong mỗi tài liệu và chủ đề của tài liệu đó. Chủ đề của những tài liệu, tin tức không xác định sau đó có thể được dự đoán và phân loại một cách tự động dựa trên mối quan hệ này.

Các kỹ thuật học máy có giám sát đã được nghiên cứu và ứng dụng rộng rãi cho bài toán phân loại chủ đề tin tức có thể kể đến như: K láng giềng gần nhất (KNN), Máy Vector Hỗ trợ (Support Vector Machine -SVM), hồi quy Logistic, … Bên cạnh đó, các bộ phân loại dạng cây như Cây Quyết Định (Decision Tree) và Rừng Ngẫu Nhiên (Random Forest) cũng đã được nghiên cứu cho tác vụ phân loại chủ đề. Mặt khác, các phương pháp học sâu đã đạt được kết quả vượt trội so với các thuật toán học máy trước đây trong các tác vụ như phân loại hình ảnh, nhận dạng khuôn mặt, xử lý ngôn ngữ tự nhiên… trong thời gian gần đây. Thành công của các thuật toán học sâu này phụ thuộc vào khả năng mô hình hóa các mô hình phức tạp và phi tuyến tính của chúng.

Chương 2. Cơ sở lý thuyết

2.1. Các phương pháp tiền xử lý dữ liệu

2.1.1. Làm sạch dữ liệu văn bản

- **Loại bỏ từ dừng**

Từ dừng (Stop Words) là những từ thường xuất hiện với tần số cao trong văn bản, tuy nhiên lại không bổ sung nhiều ý nghĩa cho một câu và chứa ít giá trị về mặt thông tin. Chính vì vậy, chúng có thể được bỏ qua một cách an toàn mà không ảnh hưởng quá nhiều đến ý nghĩa ban đầu của câu. Một số từ dừng trong tiếng việt có thể kể đến như: bị, bởi, cả, các, cái, sẽ, với, ... Kỹ thuật phổ biến nhất để xử lý những từ này là loại bỏ chúng khỏi văn bản và tài liệu. Khi loại bỏ các từ dừng, kích thước tập dữ liệu giảm và thời gian huấn luyện mô hình cũng giảm mà không ảnh hưởng lớn đến độ chính xác của mô hình.

- **Chữ viết hoa**

Một tài liệu có thể chứa nhiều cách viết hoa khác nhau để tạo thành một câu. Vì các tài liệu bao gồm nhiều câu, nên việc viết hoa đa dạng có thể là một vấn đề quan trọng khi phân loại các tài liệu với kích thước lớn. Cách tiếp cận phổ biến nhất để giải quyết vấn đề viết hoa không nhất quán là giảm mọi chữ cái thành chữ thường. Kỹ thuật này chiếu tất cả các từ trong văn bản và tài liệu vào cùng một không gian đặc trưng, nhưng nó có thể gây ra vấn đề đối với việc giải thích một số từ (ví dụ: “AI” (trí tuệ nhân tạo) thành “ai” (đại từ)) .

- **Tiếng lóng và chữ viết tắt**

Tiếng lóng và chữ viết tắt là các dạng ngoại lệ của từ cần được xử lý trong bước tiền xử lý. Chữ viết tắt là dạng rút gọn của một từ hoặc cụm từ chứa hầu hết các chữ cái đầu tiên tạo nên các từ, chẳng hạn như TP là viết tắt của thành phố hoặc UBND là viết tắt của ủy ban nhân dân. Tiếng lóng hoặc từ lóng là một tập hợp con của ngôn ngữ được sử dụng trong văn bản hoặc trong cách nói chuyện thân mật. Tiếng lóng có thể bao hàm có nhiều nghĩa khác nhau khi được sử dụng, chẳng hạn như từ gato ngoài để chỉ tên một loại bánh còn có thể được sử dụng để diễn đạt cảm giác ganh tị với một ai đó hoặc một điều gì đó. Một phương pháp để xử lý tiếng lóng và chữ viết tắt là chuyển đổi chúng sang ngôn ngữ hình thức.

- **Loại bỏ nhiễu**

Hầu hết các tập dữ liệu văn bản và tài liệu chứa nhiều ký tự không cần thiết như dấu chấm câu và ký tự đặc biệt. Những ký tự này có thể đóng vai trò quan trọng đối với sự hiểu biết của con người về tài liệu, nhưng nó có thể gây bất lợi cho các thuật toán phân loại.

- **Sửa lỗi chính tả**

Lỗi chính tả hoặc lỗi đánh máy thường xuất hiện trong các văn bản và tài liệu, đặc biệt là trong các bộ dữ liệu văn bản trên mạng xã hội (ví dụ: Twitter, Facebook). Trong lĩnh vực xử lý ngôn ngữ tự nhiên, nhiều thuật toán, kỹ thuật và phương pháp đã được phát triển để giải quyết vấn đề này.

- **Tạo gốc**

Đối với ngôn ngữ như tiếng anh, một từ có thể có nhiều biến thể hình thái học khác nhau, nghĩa là những hình hài ngữ pháp khác nhau của một từ, hay còn gọi là từ hình. Những biến thể này không làm thay đổi hạt nhân ý nghĩa của từ gốc ban đầu. Ví dụ các từ như "walks", "walking", "walked" đều là các biến thể hình thái của từ gốc "walk" và đều mang ý nghĩa là "đi bộ". Một cách thức để thống nhất ngữ nghĩa của những biến thể từ này là rút gọn từ về dạng gốc của của nó. Từ gốc chính là cơ sở để tạo ra các biến thể hình thái, chẳng hạn như bằng cách thêm các hậu tố vào từ gốc để tạo ra các biến thể cho mỗi một mục đích ngữ pháp khác nhau.

Một dạng biến thể khác của từ là biến thể ngữ âm - hình thái học. Đây là những biến dạng về mặt ngữ âm và cấu tạo từ chứ không phải là hình thái ngữ pháp của nó. Ở đây có hiện tượng cùng một ý nghĩa từ vựng được định hình một cách khác nhau. Nhìn chung trong tiếng việt hiện tượng biến thể ngữ âm - hình thái học thường phổ biến hơn là biến thể hình thái học. Chẳng hạn như các cặp biến thể: trời-giời, trăng-giăng, sờ-rờ, …

2.1.2. Tách văn bản (Text Tokenization)

2.1.2.1. Tổng quan về kỹ thuật tách văn bản

Tách văn bản là quá trình tách một cụm từ, câu, đoạn văn, một hoặc nhiều tài liệu văn bản thành các đơn vị nhỏ hơn. Những đơn vị này được gọi là Token và là đơn vị tối thiểu mà máy có thể hiểu và xử lý. Chính vì vậy, bất kỳ chuỗi văn bản nào cũng cần

được tách thành các Token có ý nghĩa để từ đó có thể tiếp tục với các tác vụ xử lý phức tạp hơn.

Công đoạn tách văn bản có thể phức tạp hoặc đơn giản tùy thuộc vào nhu cầu, mục đích của ứng dụng xử lý ngôn ngữ tự nhiên cũng như độ phức tạp của ngôn ngữ được xử lý. Có 3 kỹ thuật tách văn bản chủ yếu là: tách dựa trên từ (word-based tokenization), tách dựa trên ký tự (character-based tokenization) và tách dựa trên từ phụ (subword-based tokenization):

- ***Tách dựa trên từ***: đây là kỹ thuật Tokenization được sử dụng phổ biến nhất. Kỹ thuật này tách đoạn văn bản thành các từ (ví dụ như tiếng anh) hoặc âm tiết (ví dụ như tiếng việt) dựa trên dấu phân cách. Dấu phân cách thường được sử dụng phổ biến nhất là dấu khoảng trắng (space). Tuy nhiên, kỹ thuật tách dựa trên từ có một nhược điểm là sẽ dẫn đến một kho ngữ liệu khổng lồ và một lượng từ vựng lớn, khiến cho việc tính toán đòi hỏi nhiều tài nguyên hơn. Kỹ thuật tách dựa trên ký tự là một giải pháp để có thể giải quyết vấn đề trên.
- ***Tách dựa trên ký tự:*** kỹ thuật tách dựa trên ký tự tách văn bản thô thành các ký tự riêng lẻ. Logic đằng sau kỹ thuật này là một ngôn ngữ có thể có nhiều từ khác nhau nhưng chỉ có một số ký tự cố định. Kết quả là hạn chế đáng kể được một lượng từ vựng khi tách và sử dụng ít token hơn so với kỹ thuật tách dựa trên từ. Một trong những lợi thế chính của kỹ thuật tách dựa trên ký tự là sẽ không có hoặc có rất ít từ không xác định hoặc từ OOV (Out Of Vocabulary - Ngoài phạm vi vốn từ). Chính vì vậy nó có thể biểu diễn các từ chưa biết (những từ không được nhìn thấy trong quá trình huấn luyện) bằng cách biểu diễn cho mỗi ký tự. Nhược điểm của kỹ thuật này một ký tự thường không mang đầy đủ ý nghĩa như một từ. Bên cạnh đó, tuy kỹ thuật này giúp giảm kích thước từ vựng nhưng lại làm tăng độ dài chuỗi trong mã hóa dựa trên ký tự.
- ***Tách dựa trên từ phụ:*** Kỹ thuật này được sử dụng để giải quyết các vấn đề mà tách dựa trên từ gặp phải (kích thước từ vựng rất lớn, số lượng lớn Token OOV và ý nghĩa khác nhau của các từ tương tự nhau) và mã thông

báo dựa trên ký tự (chuỗi rất dài và mã thông báo riêng lẻ ít ý nghĩa hơn). 2 nguyên tắc mà kỹ thuật này tuân theo là không chia các từ thường dùng thành các từ phụ nhỏ hơn và chia các từ hiếm thành các từ phụ có ý nghĩa.

2.1.2.2. Phương pháp tách từ đối với văn bản tiếng việt

Không giống như tiếng anh, ranh giới giữa các từ tiếng việt không phải lúc nào cũng là khoảng trắng và các từ thường được cấu tạo bởi những đơn vị ngôn ngữ đặc biệt gọi là "hình thái âm tiết". Tiếng Việt thường được coi là đơn âm tiết vì các hình thái của nó được coi là đơn âm tiết, ví dụ: "tim" có nghĩa là "trái tim". Tuy nhiên, một số từ tiếng Việt có thể bao gồm một hoặc nhiều âm tiết, bao gồm các hình vị đơn âm kết hợp với nhau để tạo ra một từ khác. Một ví dụ của từ ghép "mạnh mẽ" có nguồn gốc từ các hình vị là: "mạnh" với ý nghĩa đề cập đến sức mạnh thuần túy, "mẽ" với ý nghĩa là "kịch tính", kết hợp với nhau để tạo ra từ "mạnh mẽ", đề cập đến sự mạnh bạo hơn bình thường.

Sự nhập nhằng về mặt ngữ nghĩa của từ trong tiếng việt đôi khi có thể gây ra trở ngại khi xây dựng các bộ phân loại. Tuy nhiên, các trường hợp nhập nhằng từ có thể được giải quyết tốt với giải thuật tìm kết hợp cực đại (MM - Maximum Matching). Phương pháp này còn được xem là kết hợp dài nhất (LM - Longest Matching) trong một số nghiên cứu. MM được sử dụng để xác định ranh giới từ trong các ngôn ngữ như tiếng Trung, tiếng Việt và tiếng Thái. Phương pháp này là một thuật toán tham lam, trong đó việc tách từ được thực hiện bằng cách chọn các từ dài nhất dựa trên từ điển. Việc tách từ có thể bắt đầu từ một trong hai đầu của dòng hoặc câu mà không có bất kỳ sự khác biệt nào trong kết quả phân đoạn. Nếu kích thước của từ điển được sử dụng tách từ này là đủ lớn, phương pháp sẽ tạo ra kết quả tách từ với độ chính xác cao.

Giả sử ta có dãy $C_1, C_2, \dots, C_n$ biểu diễn cho dãy tiếng của một chuỗi. MM sẽ dựa vào tập từ vựng để duyệt qua dãy và xác định đâu là từ. Chẳng hạn $\_C_1\_$ là từ đơn 1 tiếng thì sẽ tiếp tục tìm kiếm $\_C_1C_2\_$ nếu nó là từ 2 tiếng, và cứ tiếp tục như vậy cho đến khi sự kết hợp các tiếng tạo thành từ dài nhất. Từ hợp lý nhất sẽ là từ dài nhất. Tiến trình này sẽ được tiếp tục cho đến khi toàn bộ chuỗi đã được tách thành các từ tương ứng.

2.1.3. Vector hóa văn bản

2.1.3.1. Tổng quan về kỹ thuật vector hóa văn bản

Các thuật toán học máy hoạt động trên một không gian đặc trưng số (numeric latent space), với đầu vào thường là một mảng hai chiều trong đó các hàng là các đối tượng dữ liệu và các cột là các đặc trưng. Tuy nhiên, ngôn ngữ tự nhiên lại là dữ liệu ở dạng văn bản thô. Chính vì vậy để có thể thực hiện các tác vụ học máy trên ngôn ngữ tự nhiên, tài liệu ban đầu cần được chuyển đổi thành biểu diễn vector dưới dạng số. Quá trình này được gọi là trích xuất đặc trưng (feature extraction) hay đơn giản hơn là vector hóa và là bước đầu tiên cần thiết để tiến tới phân tích nhận biết ngôn ngữ.

Trong bài toán phân loại chủ đề tin tức tiếng việt, các đối tượng dữ liệu là toàn bộ nội dung của một bài báo. Những đối tượng này có thể khác nhau về độ dài văn bản hoặc số lượng từ vựng. Mặc dù vậy, các vector của chúng luôn có độ dài đồng nhất. Mỗi một thuộc tính (property) của biểu diễn vector chính là một đặc trưng. Đối với văn bản, các đặc trưng này sẽ đóng vai trò biểu diễn cho các thuộc tính của tài liệu như: nội dung, độ dài, tác giả, nguồn và ngày xuất bản của tài liệu. Tập hợp lại, các đặc trưng của tài liệu cùng tạo nên một không gian đặc trưng đa chiều mà ở đó, các thuật toán học máy có thể được áp dụng và phân tích. Một số kỹ thuật vector hóa phổ biến có thể kể đến như: Bag of Words (BoW), Word2Vec, Doc2Vec, TF - IDF,…

2.1.3.2. Các kỹ thuật vector hóa văn bản

- **Túi từ (Bag of Words)**

Mô hình túi từ (Bag of Words - BoW) là một biểu diễn rút gọn và đơn giản hóa tài liệu văn bản dựa trên các phần được chọn của văn bản với các tiêu chí cụ thể. Trong BoW, phần nội dung văn bản, chẳng hạn như đoạn văn hoặc câu, được coi như là một túi từ chứa các từ duy nhất. Quan hệ ngữ nghĩa và thứ tự xuất hiện giữa những từ này thường được bỏ qua. Mặc dù vậy, tính đa dạng của một từ vẫn có thể được xác định bằng cách đếm tần suất xuất hiện của từ đó trong tài liệu, nhờ vậy có thể được sử dụng sau này để xác định các điểm trọng tâm của tài liệu. Bên cạnh đó, tần suất xuất hiện của mỗi từ còn được sử dụng để biểu diễn một đoạn văn bản dưới dạng một vectơ số học, nhờ đó việc xây dựng các bộ phân loại chủ đề sẽ trở nên khả thi so với việc sử dụng dữ liệu dạng văn bản đơn thuần.

Bước đầu tiên trong việc xây dựng túi từ chính là lập từ điển. Từ điển ở đây có thể được hiểu là danh sách các từ được bỏ qua số lần xuất hiện trong tài liệu. Chẳng hạn, xét câu văn sau:

“xin chào, tên tôi là Nhật, đây là một lời chào và giới thiệu tên tôi”

Từ câu văn trên, chúng ta có được từ điển như sau:

[xin, chào, tên, tôi, là, Nhật, đây, một, lời, và, giới, thiệu]

Sau đó, số lần xuất hiện của mỗi từ trong câu văn sẽ được đếm dựa trên từ điển:

Xin	1
Chào	2
Tên	2
Tôi	2
Là	2
Nhật	1
Đây	1
Một	1
Lời	1
Và	1
Giới	1
Thiệu	1

Với kết quả như trên, câu văn ban đầu có thể được biểu diễn dưới dạng vector số là:

Vector biểu diễn = [1, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1]

Giá trị của thành phần thứ i trong vector biểu diễn chính là số lần xuất hiện của từ tương ứng trong tài liệu ban đầu.

Mặc dù là hình thái căn bản nhất trong các kỹ thuật vector hóa văn bản, mô hình túi từ vẫn có những lợi ích sau đây:

+ Tính đơn giản và khả năng giải thích: Mô hình túi từ là một biểu diễn đơn giản của dữ liệu văn bản, dễ hiểu và dễ thực hiện.
+ Dễ triển khai: Kỹ thuật túi từ chỉ cần đầu vào là văn bản đã được làm sạch và token hóa, do đó có thể được thực hiện nhanh chóng và dễ triển khai.
+ Tính thưa: Hầu hết các mục trong vector đặc trưng đều bằng không. Điều này làm cho việc lưu trữ và xử lý một lượng lớn dữ liệu văn bản trở nên hiệu quả.
+ Khả năng mở rộng: Vì đây là một mô hình có tính thưa trong cách biểu diễn dữ liệu, dễ thực hiện, nên nó có thể mở rộng tốt cho một số lượng lớn tài liệu cả về mặt không gian và khả năng tính toán.
+ Khả năng khái quát hóa: Mô hình túi từ có thể được áp dụng cho nhiều tác vụ trong xử lý ngôn ngữ tự nhiên, bao gồm phân loại văn bản, truy xuất thông tin, phân cụm và đo lường tính tương đồng của tài liệu. Nói chung, nó được sử dụng làm đầu vào cho các mô hình xử lý ngôn ngữ tự nhiên phức tạp hơn trong các ứng dụng khác nhau.

Bên cạnh đó, mô hình túi từ vẫn tồn tại một số hạn chế là:

+ Trật tự từ bị bỏ qua: Mô hình túi từ coi tất cả các lần xuất hiện của một từ là ngang nhau, bất kể thứ tự xuất hiện của chúng trong câu. Điều này có nghĩa là nó không thể nắm bắt được mối quan hệ giữa các từ trong câu và ý nghĩa mà chúng truyền tải. Chẳng hạn, xét 2 câu văn sau:

“Cậu bé làm mẹ vui”

“Mẹ làm cậu bé vui”

Mặc dù 2 câu văn với 2 ý nghĩa khác nhau, tuy nhiên mô hình túi từ vẫn biểu diễn cả 2 dưới dạng 1 vector giống nhau vì các từ và tần suất xuất hiện của 2 câu là như nhau.

+ Cấu trúc ngữ pháp bị bỏ qua: Mô hình túi từ bỏ qua dấu câu và cấu trúc ngữ pháp, do đó các câu với ý nghĩa khác nhau đều có thể được biểu diễn dưới cùng 1 vector BoW giống nhau.

+ Hạn chế thông tin ngữ nghĩa: Mô hình túi từ chỉ ghi lại sự hiện diện hay vắng mặt của một từ trong tài liệu, chứ không phải ý nghĩa hoặc ngữ cảnh mà từ đó xuất hiện.
+ Số lượng chiều dữ liệu lớn: Nếu một ngữ liệu văn bản chứa một số lượng lớn các từ duy nhất, thì ma trận dùng để biểu diễn túi từ sẽ có số lượng đặc trưng (cột) lớn, điều này có thể dẫn đến tình trạng quá khớp (overfit) của mô hình.

- **TF - IDF**

TF-IDF Vectorizer không chỉ tính đến số lần một từ xuất hiện trong tài liệu mà còn cả mức độ quan trọng của từ đó trong toàn bộ khối văn bản (corpus). Kỹ thuật TF - IDF kết hợp 2 khái niệm chính, bao gồm tần số thuật ngữ (TF) và tần số tài liệu nghịch đảo (IDF):

- Tần số thuật ngữ (TF): là số lần xuất hiện của một thuật ngữ cụ thể trong một tài liệu. TF cho biết mức độ quan trọng của một thuật ngữ cụ thể trong tài liệu. TF biểu diễn mọi văn bản từ dữ liệu dưới dạng ma trận có các hàng là số lượng tài liệu và các cột là số lượng các thuật ngữ riêng biệt trong tất cả các tài liệu.
- Tần số tài liệu (DF): là số lượng tài liệu chứa một thuật ngữ cụ thể. Tần số tài liệu cho biết mức độ phổ biến của thuật ngữ.
- Tần số tài liệu nghịch đảo (IDF): là trọng số của một thuật ngữ, IDF đóng vai trò giảm trọng số của một thuật ngữ nếu các lần xuất hiện của thuật ngữ này nằm rải rác trong tất cả các tài liệu. IDF có thể được tính như sau:

Ý tưởng cơ bản của phương pháp TF-IDF là từ lý thuyết mô hình hóa ngôn ngữ mà các thuật ngữ trong một tài liệu nhất định có thể được chia thành hai loại: những từ quan trọng và những từ không quan trọng, tức là liệu một từ có liên quan với chủ đề của một tài liệu nhất định hay không. Hơn nữa, mức độ quan trọng của một thuật ngữ đối với một tài liệu nhất định có thể được đánh giá bằng TF và IDF và trong phép nhân giữa TF và IDF, nó được sử dụng để đo tầm quan trọng của một thuật ngữ trong toàn bộ tài liệu.

- **Word Embedding**

Nhúng từ (Word Embedding) là một trong những phương pháp biểu diễn dữ liệu văn bản phổ biến nhất trong lĩnh vực xử lý ngôn ngữ tự nhiên và đã góp phần tạo nên sự thành công cho các mô hình học sâu khi xử lý các bài toán trong lĩnh vực này. Với phương pháp nhúng từ, các từ trong văn bản sẽ được biểu diễn bởi một vectơ số thực trong không gian vectơ R chiều. Qua đó cho phép nắm bắt ngữ nghĩa, sự tương đồng về ngữ nghĩa giữa các từ cũng như thông tin cú pháp cho các từ.

Phương pháp nhúng từ Word2Vec lần đầu được giới thiệu bởi Mikolov và cộng sự vào năm 2013 [10]. Biểu diễn vector của từ bởi kỹ thuật Word2Vec có thể thu được bằng 2 thành phần riêng biệt là Skip Gram và Continuous Bag of Word (CBOW):

+ CBOW: Ngữ cảnh của mỗi từ sẽ được sủ dụng để làm đầu vào để mô hình có thể dự đoán từ tương ứng dựa trên ngữ cảnh.
+ Skip Gram: Mô hình Skip Gram được coi là phiên bản đảo ngược của mô hình CBOW. Cho trước một vị trí ngữ cảnh, mô hình cần đưa ra được phân bố xác suất của mỗi từ ở vị trí đó. Trong cả hai trường hợp, mạng sử dụng lan truyền ngược để học ra biểu diễn vector của từ.

2.2. Giảm chiều dữ liệu

2.2.1. Giới thiệu phương pháp phân tích suy biến

Phương pháp phân tích suy biến (singular value decomposition) được viết tắt là SVD là một trong những phương pháp thuộc nhóm phân rã ma trận được phát triển lần đầu bởi những nhà hình học vi phân. Mục tiêu của phương pháp này là tìm ra một ma trận xấp xỉ với ma trận gốc nhưng có kích thước nhỏ hơn. Phương pháp này hiện đã được ứng dụng rộng rãi trong các lĩnh vực như hồi qui tuyến tính, xử lý hình ảnh, các thuật toán phân cụm, các thuật toán nén và đặc biệt là các thuật toán giảm chiều dữ liệu. Người ta đã chứng minh được rằng ma trận xấp xỉ tốt nhất được biểu diễn dưới dạng tích của 3 ma trận rất đặc biệt bao gồm 2 *ma trận trực giao* và 1 *ma trận đường chéo*. Điều đặc biệt của ma trận đường chéo đó là các phần tử của nó chính là những giá trị riêng của ma trận gốc. Những điểm dữ liệu trong không gian mới có thể giữ được 100% thông tin ban đầu hoặc chỉ giữ một phần lớn thông tin của dữ liệu ban đầu thông qua các phép truncate SVD. Bằng cách sắp xếp các trị riêng theo thứ tự giảm dần trên đường chéo chính thuật toán SVD có thể thu được ma trận xấp xỉ tốt nhất mà vẫn đảm

bảo giảm được hạng của ma trận sau biến đổi và kích thước các ma trận nhân tử nằm trong giới hạn cho phép. Do đó nó tiết kiệm được thời gian và chi phí tính toán và đồng thời cũng tìm ra được một giá trị dự báo cho ma trận gốc với mức độ chính xác cao.

2.2.2. Phát biểu phép phân tích suy biến

2.2.2.1. Hệ trực giao và ma trận trực giao

Một hệ véc tơ cơ sở $\{u_1, u_2, u_3, .. u_D\} \in R^k$ được gọi là một *hệ trực giao* (*orthogonal*) nếu thỏa mãn hệ điều kiện sau:

$$\begin{cases} \|u_i\|_2^2 > 0 \\ u_i^T u_j = 0 \ \forall \ i \neq j \end{cases}$$

Với $\|u_i\|_2^2$ chính là bình phương chuẩn bậc 2 (L2 norm) của vector u_i. Như vậy các chiều của hệ trực giao là vuông góc với nhau đôi một. Khi giá trị $\|u_i\|_2^2 = 1 \ \forall \ i$, ta được một trường hợp đặc biệt của hệ trực giao là hệ trực chuẩn. Một tập hợp các vector đơn vị bất kỳ trong không gian K chiều sẽ tạo thành một hệ trực chuẩn.

Ma trận trực giao (*orthogonal matrix*) là ma trận vuông thỏa mãn các dòng và cột của nó là một hệ trực chuẩn. Điều đó có nghĩa là một ma trận trực giao $U \in R^{D \times D}$ thoả mãn:

$$U^T U = I_D$$

Với I_D là ma trận đơn vị D chiều.

2.2.2.2. Ma trận đường chéo

Một ma trận **D** được gọi là ma trận đường chéo khi các phần tử của nó thỏa mãn:

$$d_{ii} \neq 0, d_{ij} = 0 \ \forall \ i \neq j$$

Hay nói cách khác ma trận có các phần tử trên đường chéo chính khác 0 và các phần tử còn lại bằng 0. Ma trận đường chéo có thể không vuông. Ma trận đơn vị là một dạng ma trận đường chéo khi nó vừa là một ma trận vuông và đồng thời các phần tử trên đường chéo chính bằng 1. Bên cạnh đó, một ma trận $U \in R^{D \times D}$ có các cột tạo thành một hệ trực giao thì tích của nó với ma trận chuyển vị của nó sẽ tạo thành một ma trận đường chéo.

2.2.2.3. Định nghĩa phép phân tích suy biến

Phép phân tích suy biến thuộc nhóm phương pháp phân rã ma trận, trong đó một ma trận gốc sẽ được phân tích thành tích của các ma trận số thực hoặc ma trận số phức. Với ma trận A_{mn}, phép phân tích suy biến trên ma trận này sẽ được biểu diễn như sau:

$$A_{mn} = U_{mm}\Sigma_{mn}V_{nn}^{T}$$

Trong đó:

- A_{mn} là ma trận $A \in R^{m \times n}$
- U_{mm}, V_{nn} là các ma trận trực giao
- Σ_{mn} là ma trận đường chéo

2.3. Các mô hình phân loại chủ đề tin tức

2.3.1. Nhóm các mô hình học máy

2.3.1.1. Mô hình Naive Bayes

a. Giới thiệu về mô hình Naïve Bayes

Kỹ thuật phân loại văn bản Naïve Bayes đã được sử dụng rộng rãi cho các nhiệm vụ phân loại tài liệu từ những năm 1950. Phương pháp phân loại Naïve Bayes về mặt lý thuyết dựa trên định lý Bayes, được xây dựng bởi Thomas Bayes trong khoảng thời gian 1701–1761. Kỹ thuật này là một mô hình tạo sinh (generative model), là phương pháp phân loại văn bản truyền thống nhất. Một mô hình tạo sinh có nhiệm vụ mô hình hóa phân phối đầu vào cho một lớp hoặc danh mục nhất định. Cách tiếp cận này dựa trên giả định rằng các đặc trưng của dữ liệu đầu vào là độc lập có điều kiện đối với lớp, cho phép thuật toán đưa ra dự đoán nhanh chóng và chính xác.

2 hướng tiếp cận chính đối với bài toán phân loại dựa trên mô hình Naive Bayes là mô hình Bernoulli đa biến (Multivariate Bernoulli Model) và mô hình đa thức (Multinomial). Về cơ bản, cả hai mô hình đều tính toán xác suất hậu nghiệm của một lớp, dựa trên sự phân bố của các từ trong tài liệu. Các mô hình này bỏ qua vị trí thực tế của các từ trong tài liệu và hoạt động với giả định "túi từ". Sự khác biệt chính giữa hai mô hình này là giả định về việc tính đến (hoặc không tính đến) tần số từ và cách tiếp cận tương ứng để lấy mẫu xác suất. Bất kể chúng ta lập mô hình các tài liệu trong mỗi lớp như thế nào (có thể là mô hình Bernoulli đa biến hoặc mô hình đa thức), các mô hình lớp thành phần (nghĩa là các mô hình tạo sinh cho các tài liệu trong mỗi lớp) có thể được sử dụng cùng với định lý Bayes để tính toán xác suất hậu nghiệm của lớp

cho một tài liệu nhất định và sau đó lớp có xác suất sau cao nhất có thể được gán cho tài liệu.

Theo đó, với bộ dữ liệu chứa N tập tài liệu D = $\{d_1, d_2, \ldots, d_n\}$ tương ứng với k tập chủ đề C = $\{c_1, c_2, \ldots, c_n\}$, Định lý Bayes có thể được áp dụng như sau:

$$P\ (c|d)\ = \frac{P(d|c) \times P(c)}{P(d)}$$

Xác suất của mỗi tài liệu thuộc vào các chủ đề trong tập chủ đề cho trước có thể tính được dựa vào biểu thức trên. Từ đó ta có thể xác định được chủ đề cho tài liệu bằng cách chọn ra chủ đề với xác suất cao nhất:

$$C\ = \arg max_{(c \in C)} P(d|c)P(c)$$
$$= \arg max_{(c \in C)} P(x_1, x_2, \ldots, x_n|c)p(c)$$

b. Mô hình đa thức Naive Bayes

Mô hình đa thức Naive Bayes chủ yếu được sử dụng trong phân loại tài liệu mà trong đó vector đặc trưng của tài liệu được tính theo hướng tiếp cận túi từ. Lúc này, mỗi tài liệu sẽ được biểu diễn dưới dạng 1 vector với độ dài k với k chính là số lượng từ trong từ điển. Khi đó, thuật toán Naïve Bayes có thể được viết dưới dạng:

$$P\ (c|d) = \frac{P(c) \prod_{w \in d} P(d|c)^{n_{wd}}}{P(d)}$$

Trong đó n_{wd} là số lần từ w xuất hiện trong tài liệu, khi đó xác suất của từ w trong chủ đề c có thể được tính như sau:

$$P(w|c) = \frac{1 + \sum_{d \epsilon D_c} n_{wd}}{k + \sum w' \sum_{d \epsilon D_c} n_{w'd}}$$

2.3.1.2. Mô hình hồi quy Logistic

Hồi quy logistic là một kỹ thuật phân lớp học có giám sát được sử dụng để dự đoán xác suất của một biến mục tiêu. Hay nói cách khác, Hồi quy logistic là một mô hình xác suất dự đoán giá trị đầu ra rời rạc từ một tập các giá trị đầu vào. Biến mục tiêu hoặc biến phụ thuộc là nhị phân về mặt bản chất, có nghĩa là chỉ có thể có hai lớp. Chính vì vậy, hồi quy Logistic được sẽ được sử dụng hợp lý nhất đối với các loại dữ liệu nhị phân.

Mặc dù thường được sử dụng để dự đoán các biến mục tiêu nhị phân, hồi quy logistic có thể được mở rộng và phân loại thêm thành ba loại khác nhau. Những loại này bao gồm:

+ Nhị thức: Trong đó biến mục tiêu chỉ có thể có hai kiểu khả dĩ. Ví dụ: dự đoán một email có phải là thư rác hay không.

+ Đa thức: Trong đó biến mục tiêu có thể có ba loại trở lên, có thể không có bất kỳ ý nghĩa định lượng nào. Ví dụ: dự đoán bệnh.

+ Thứ tự: Trong đó biến mục tiêu có thứ tự được sắp xếp. Ví dụ: Xếp hạng một trang web từ 1 đến 5.

Phương pháp hồi quy Logistic bắt đầu với một hàm hồi quy tuyến tính như sau:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n$$

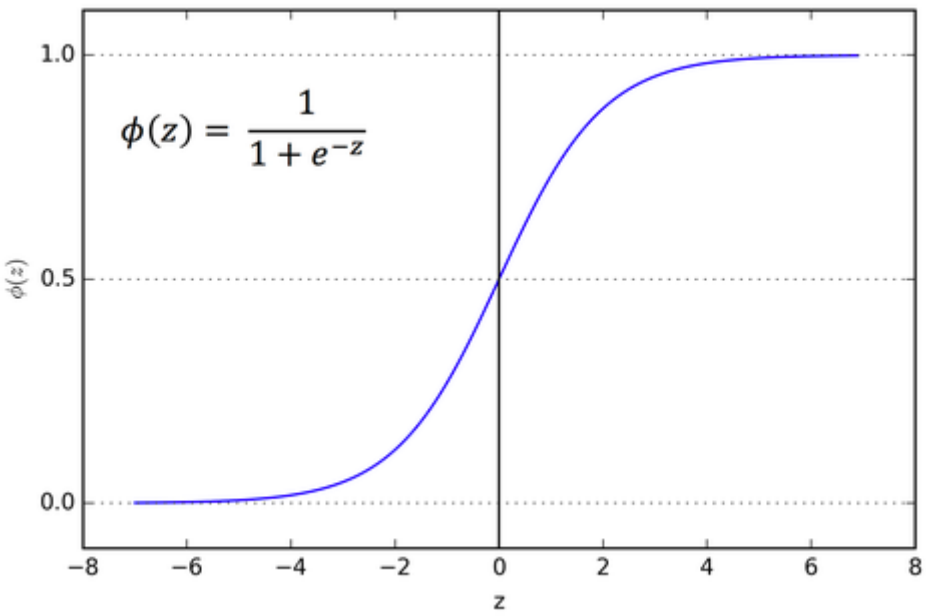
Trong đó y là biến độc lập và X_1, X_2,... X_n là các biến phụ thuộc. Giá trị của biến y sau đó sẽ được ánh xạ để trở thành xác suất bằng cách sử dụng hàm Sigmoid (Logistic). Hàm Sigmoid có thể nhận bất kỳ số có giá trị thực nào và ánh xạ nó thành một giá trị từ 0 đến 1. Hàm Sigmoid có thể được biểu diễn như sau:

$$p = \frac{1}{1 + e^{-y}}$$

Áp dụng hàm Sigmoid vào hàm hồi quy tuyến tính, ta được:

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n)}}$$

Điều này cho giá trị y cực kỳ gần 0 nếu x là giá trị âm lớn và gần bằng 1 nếu x là giá trị dương lớn. Sau khi giá trị đầu vào đã được ép về phía 0 hoặc 1, tác vụ phân lớp có thể được thực hiện.



Hình 3: Đồ thị biểu diễn hàm Sigmoid

2.3.1.3. Mô hình Random Forest

a. Giới thiệu về mô hình cây quyết định

Rừng ngẫu nhiên là một thuật toán học có giám sát. "Khu rừng" mà nó xây dựng là một tập hợp các cây quyết định, thường được huấn luyện bằng phương pháp học tập kết hợp (Ensemble Learning) Bagging. Cây quyết định là một cấu trúc cây dạng lưu đồ, trong đó mỗi nút nội bộ biểu thị một phép thử trên một thuộc tính, mỗi nhánh biểu thị một kết quả của phép thử và mỗi nút lá (hoặc nút đầu cuối) biểu diễn một nhãn lớp. Nút trên cùng trong một cây là nút gốc.

Trong cây quyết định, để dự đoán lớp của tập dữ liệu đã cho, thuật toán bắt đầu từ nút gốc của cây. Thuật toán này so sánh các giá trị của thuộc tính gốc với thuộc tính bản ghi (tập dữ liệu thực) và dựa trên sự so sánh này, đi theo nhánh và nhảy đến nút tiếp theo.

Đối với nút tiếp theo, thuật toán lại so sánh giá trị thuộc tính với các nút con khác và di chuyển xa hơn. Quá trình này được tiếp tục cho đến khi thuật toán đạt đến nút lá của cây. Quy trình hoàn chỉnh có thể được hiểu rõ hơn bằng mô tả như sau:

- Bước 1: Bắt đầu cây với nút gốc và nút này chứa tập dữ liệu hoàn chỉnh.
- Bước 2: Tìm thuộc tính tốt nhất trong tập dữ liệu bằng cách sử dụng phép đo lựa chọn thuộc tính (Atrribute Selection Measure – ASM).

- Bước 3: Chia nút gốc thành các tập con chứa các giá trị có thể có cho các thuộc tính tốt nhất.
- Bước 4: Tạo nút cây quyết định chứa thuộc tính tốt nhất.
- Bước 5: Tạo cây quyết định mới theo phương pháp đệ quy bằng cách sử dụng các tập con của tập dữ liệu đã tạo ở bước 3. Tiếp tục quá trình này cho đến khi đạt đến một giai đoạn mà không thể phân chia thêm các nút và đạt được nút cuối cùng là nút lá.

Trong khi thực hiện phân lớp dựa trên cây quyết định, vấn đề chính nảy sinh là làm thế nào để chọn thuộc tính tốt nhất cho nút gốc và cho các nút con. Một kỹ thuật được gọi là phép đo lựa chọn thuộc tính – ASM sẽ được sử dụng để giải quyết vấn đề này. Bằng phép đo này, chúng ta có thể dễ dàng chọn thuộc tính tốt nhất cho các nút của cây. Có hai kỹ thuật phổ biến cho ASM là độ lợi thông tin và chỉ số Gini.

- **Độ lợi thông tin (Information Gain)**

Độ lợi thông tin dựa trên sự giảm của hàm Entropy (thước đo tính ngẫu nhiên của thông tin đang được xử lý) khi tập dữ liệu được phân chia trên một thuộc tính. Các nút (node)/lá (leaf) có tất cả các trường hợp đều thuộc về 1 lớp duy nhất sẽ có entropy bằng 0. Trong khi đó, entropy của nút/lá mà các lớp được phân chia đều nhau sẽ là 1. Hàm Entropy được định nghĩa như sau:

$$E = -\sum_{i=1}^{n} p_i \, log_2(p_i)$$

Trong đó pi là xác suất chọn ngẫu nhiên một ví dụ trong lớp i. Để xây dựng một cây quyết định, ta phải tìm tất cả thuộc tính trả về độ lợi thông tin cao nhất:

$$information\ gain = Entropy_{parent} - Entropy_{children}$$

Entropyparent chính là mức entropy của nút cha và Entropychildren là mức entropy trung bình của các nút con được tính thông qua mức entropy của mỗi nút con được tính trọng số theo tỷ lệ của nó so với nút cha.

Sau khi tính toán được Độ lợi thông tin, ta sẽ chọn ra nút lá có Độ lợi thông tin lớn nhất để làm nút phân tách ở bước đó cho Cây quyết định. Sau khi chọn nút phân tách, ta tách nút lá đó thành 2 nút lá con dựa trên các giá trị thuộc tính của nút cha. Lặp lại

bước này cho đến khi không phân tách được nữa (entropy = 0 hoặc không còn thuộc tính nào để phân tách).

- **Chỉ số gini (Gini index)**

Chỉ số Gini tính toán xác suất bị phân lớp sai của một thuộc tính cụ thể khi được chọn ngẫu nhiên.

$$Gini\ index = 1 - \sum_{i=1}^{n} (P_i)^2$$

Giá trị tối thiểu của Gini Index là 0. Điều này xảy ra khi nút là "thuần túy", có nghĩa là tất cả các phần tử trong nút thuộc về một lớp duy nhất. Do đó, nút này sẽ không thể bị chia tách hơn nữa. Chính vì vậy, Gini Index càng thấp thì thuật toán Cây quyết định càng hiệu quả. Gini index đạt giá trị lớn nhất khi xác suất của hai lớp là như nhau.

b. Mô hình Random Forest

Bộ phân loại Rừng ngẫu nhiên (RF) phù hợp để xử lý dữ liệu chiều cao trong vấn đề phân loại chủ đề tin tức. Một mô hình RF bao gồm một tập hợp các cây quyết định, mỗi cây được huấn luyện bằng cách sử dụng các tập hợp con ngẫu nhiên của các đặc trưng trong bộ dữ liệu. Khi sử dụng mô hình RF để dự đoán chủ đề của một bản tin, dự đoán này sẽ được tạo ra thông qua biểu quyết đa số về các dự đoán của tất cả các cây trong bộ phân loại.

Mô hình RF huấn luyện các tập dữ liệu văn bản với thời gian nhanh hơn so với các kỹ thuật khác như học sâu, nhưng lại khá chậm để tạo ra các dự đoán sau khi được huấn luyện. Do đó, để đạt được cấu trúc nhanh hơn, số lượng cây trong rừng phải giảm đi, vì nhiều cây hơn trong rừng làm tăng độ phức tạp về thời gian trong bước dự đoán.

2.3.1.4. Mô hình Support Vector Machine (SVM)

SVM là một thuật toán có giám sát nhằm tìm ra một siêu phẳng (Hyperplane) trong một không gian N chiều. Siêu có thể được hiểu đơn giản là một đường thẳng có thể phân chia các lớp ra thành hai phần riêng biệt. Trong bài toán dữ liệu có thể được phân chia tuyến tính (linearly separable), SVM sẽ cố gắng tìm ra một siêu phẳng sao cho khoảng cách từ siêu phẳng này đến các điểm dữ liệu gần nhất của các lớp là lớn

nhất. Khoảng cách này được gọi là biên (Margin) và các điểm dữ liệu này được gọi là Vectơ hỗ trợ (Support Vector).

Với định nghĩa Margin được xác định ở trên, khoảng cách ngắn nhất từ siêu phẳng đến một Margin sẽ bằng với khoảng cách này đến Margin còn lại. Như vậy Margin trong không gian n chiều có thể được tính như sau:

$$Margin = \frac{2|w^T x + b|}{||w||}$$

Khi đó một siêu phẳng cần tìm có thể được viết như sau:

$W^T.X + b = 0$ (1)

Trong đó W là trọng số của các vector. Với n đặc trưng thì ta có $W = (w_1, w_2, \ldots, w_n)$. X là các tuple của tập huấn luyện, b là độ chệch (bias) và là đại lượng vô hướng.

Khi đó, mục tiêu của SVM là cần tìm giá trị margin cực đại đồng nghĩa với việc ‖w‖ đạt cực tiểu với điều kiện:

$$y_n(W^T.x_n + b) \geq 1, \forall n = 1,2,\ldots,n$$

Trong không gian hai chiều, nếu ta xem b là trọng số bổ sung thì phương trình (1) có thể được viết là:

$$w_0 + w_1x_1 + w_2x_2 = 0$$

Các điểm nằm bên trên siêu phẳng phân tách sẽ thỏa mãn:

$$w_0 + w_1x_1 + w_2x_2 > 0$$

Tương tự, các điểm nằm bên dưới siêu phẳng phân tách sẽ thỏa mãn:

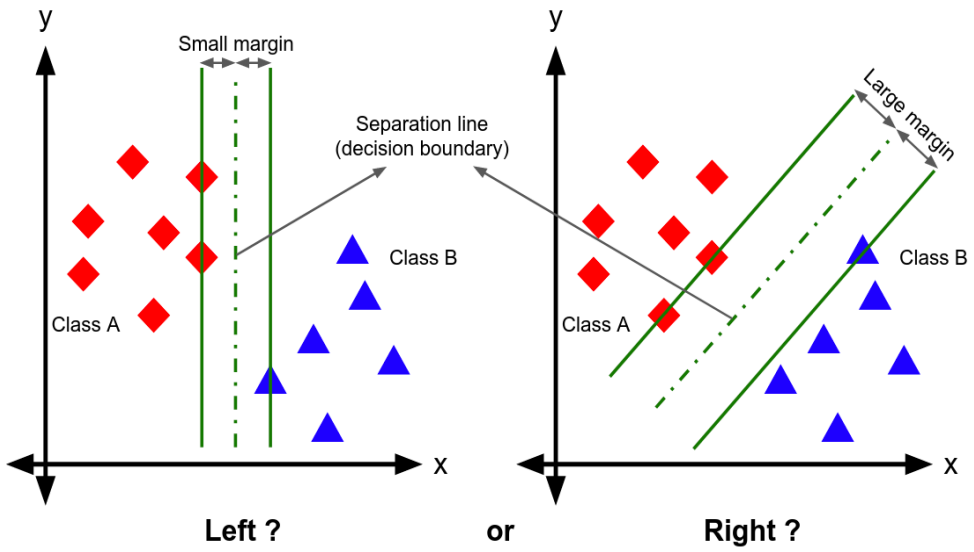
$$w_0 + w_1x_1 + w_2x_2 < 0$$

Giả sử có hai lớp A và B, mỗi lớp ứng với một giá trị là 1 và -1 tương ứng. Trọng số có thể được điều chỉnh để các siêu phẳng xác định "các cạnh" của Margin có thể được viết là:

$$H1: w_0 + w_1x_1 + w_2x_2 \geq 1$$

$$H2: w_0 + w_1x_1 + w_2x_2 \leq 1$$

Như vậy, bất kỳ điểm dữ liệu nào nằm trên hoặc phía trên H1 sẽ thuộc lớp A và bất kỳ điểm nào nằm trên hoặc phía dưới H2 sẽ thuộc về lớp B.



Hình 4: So sánh giữa Margin lớn (bên phải) và Margin nhỏ (bên trái)

Bởi vì tính thưa cũng như số lượng chiều lớn trong đặc trưng biểu diễn tài liệu, mô hình phân loại với máy vector hỗ trợ tỏ ra phù hợp đối với các tác vụ phân loại tài liệu nói chung và phân loại tin tức nói riêng (Joachims, 1998). Trong đó một số đặc trưng có thể không liên quan, nhưng chúng có xu hướng tương quan với nhau và thường được tổ chức vào các chủ đề riêng biệt có thể được phân tách tuyến tính. Mặc dù vậy, không nhất thiết phải sử dụng hàm tuyến tính cho bộ phân loại SVM. Bằng cách đi tìm một hàm biến đổi $\phi(x)$, SVM có thể ánh xạ phi tuyến dữ liệu x từ không gian đặc trưng ban đầu thành dữ liệu trong một không gian mới mà tại đó, các điểm dữ liệu có thể được phân tách tuyến tính bằng một siêu phẳng. Các hàm $\phi(x)$ thường tạo ra dữ liệu mới có số chiều cao hơn nhiều so với dữ liệu gốc, chính vì vậy việc tính toán trên các hàm này trực tiếp là vô cùng khó khăn. Một hướng tiếp cận để giải quyết giới hạn trong khả năng tính toán chính là phương pháp Kernel, trong đó thay vì tính trực tiếp tọa độ của một điểm trong không gian mới, ta sẽ tính tích vô hướng giữa 2 điểm đó thông qua một hàm Kernel.

Giả sử có 2 vector x, y cùng với ánh xạ $\varphi : R^n \rightarrow R^m$ biến đổi các vector trong không gian R^n sang không gian đặc trưng mới R^m. Lúc này tích vô hướng giữa 2 vector trong không gian mới sẽ là $\phi(x)^T \phi y$. Một Kernel sẽ là một hàm K tương ứng với tích vô hướng này:

$$K(x, y) = \phi(x)^T \phi y$$

Trong đó ϕx và ϕy là biểu diễn của x và y trong không gian đặc trưng mới.

Chẳng hạn, xem xét kernel sau:

$K(x, y) = (1 + x^T y)^2$ với $x, y \epsilon R^2$ (1)

Với 2 điểm dữ liệu (x_1, y_1) và (x_2, y_2), biểu thức (1) có thể được viết thành:

$$(1 + x_1 y_1 + x_2 y_2)^2 = 1 + x_1^2 y_1^2 + x_2^2 y_2^2 + 2x_1 y_1 + 2x_2 y_2 + 2x_1 y_1 x_2 y_2$$

Và như vậy Kernel $K(x, y) = (1 + x^T y)^2 = \phi(x)^T \phi y$ đã thực hiện tích vô hướng giữa 2 vector x, y trong không gian 6 chiều mà không cần truy cập tường minh vào không gian này.

Một số hàm Kernel thông dụng bao gồm:

- **Linear**

Đây là trường hợp đơn giản với kernel chính tích vô hướng của hai vector:

$$K(x, y) = x^T y$$

Khi huấn luyện bộ phân loại SVM với thư viện Scikit-learn, Kernel này có thể được sủ dụng bằng cách đặt kernel = 'linear'.

- **Polynomial**

Polynomial Kernel được định nghĩa như sau:

$$K(x, y) = (r + \gamma x^T y)^d$$

Với d là bậc của đa thức. Khi huấn luyện bộ phân loại SVM với thư viện Scikit-learn, Kernel này có thể được sủ dụng bằng cách đặt kernel = 'poly'.

- **Radial Basic Function**

Radial Basic Function (RBF) hay Gaussian Kernel được định nghĩa như sau:

$$K(x, y) = \exp(-\gamma \|x - y\|_2^2), \gamma > 0$$

Khi huấn luyện bộ phân loại SVM với thư viện Scikit-learn, Kernel này có thể được sủ dụng bằng cách đặt kernel = 'rbf'.

- **Sigmoid**

Polynomial Kernel được định nghĩa như sau:

$$K(x, y) = \tanh(\gamma x^T y + r)$$

Khi huấn luyện bộ phân loại SVM với thư viện Scikit-learn, Kernel này có thể được sủ dụng bằng cách đặt kernel = 'sigmoid'.

2.3.2. Nhóm các mô hình học sâu

2.3.2.1. Mạng Perceptron đa lớp

a. Giới thiệu về mạng Perceptron đa lớp

Mạng Nơ-ron truyền thẳng hoặc mạng Perceptron đa lớp (MLP) thuộc nhóm mạng thần kinh nhân tạo (ANN), là một mô hình quản lý thông tin được vận hành bằng cách mô phỏng lại hệ thống thần kinh sinh học của não bộ và cách các tế bào thần kinh hoạt động cùng nhau để hiểu đầu vào từ các giác quan của con người.

Mạng Perceptron đa lớp được xây dựng từ các Perceptron và những kết nối giữa những nút này, trong đó mỗi Perceptron sẽ đại diện cho một hàm đầu ra cụ thể còn các kết nối giữa các nút sẽ đại diện cho mức độ ảnh hưởng của nút đó lên các nút xung quanh và được gán trọng số tương ứng. Chính vì vậy, kết quả đầu ra của một mạng Nơron nhân tạo sẽ phụ thuộc vào cách mà các nút trong mạng được kết nối và trọng số tương ứng với những kết nối này. Mạng Nơron đã được ứng dụng trong nhiều lĩnh vực đa dạng như: phân loại các chữ số viết tay, nhận dạng giọng nói và dự đoán giá cổ phiếu, xếp hạng tín dụng, phân tích hành vi khách hàng, hỗ trợ ra quyết định, dự đoán tỷ giá hối đoái và lãi suất,... Ưu điểm của mạng Nơron nhân tạo là khả năng xấp xỉ các hàm phi tuyến, nhờ đó có thể phát hiện những mối liên hệ phức tạp ẩn trong dữ liệu. Tuy nhiên, hiệu năng của mạng Nơron nhân tạo lại phụ thuộc nhiều vào cấu trúc mạng mà việc tìm ra cấu trúc tối ưu thường phức tạp và tốn nhiều tài nguyên.

b. Kiến trúc mạng Perceptron truyền thẳng

Đơn vị nhỏ nhất của mạng MLP là các Perceptron. Mỗi một Perceptron có thể được xem như là một tổ hợp tuyến tính riêng biệt, bao gồm dữ liệu đầu vào, trọng số, độ chệch (hoặc ngưỡng) và đầu ra. Các Perceptron này sẽ được tổ chức thành các lớp được kết nối với nhau, nghĩa là đầu ra của lớp này sẽ trở thành đầu vào của lớp kia. Một lớp như vậy được gọi là lớp kết nối đầy đủ (fully connected layer). Như vậy, nhiệm vụ của mỗi Perceptron là nhận vào một vài giá trị đầu vào, thực hiện các tính toán và sau đó trả về một giá trị đầu ra. Quá trình tính toán này có thể được định nghĩa như sau:

$$z_j^{(l)} = w_j^{(l)T} a^{(l-1)} + b_j^{(l)}$$

Với $z_j^{(l)}$ là giá trị đầu ra thứ *j* của lớp thứ *l,* $w_j^{(l)T}$ là chuyển vị của vector trọng số thứ *j* của lớp thứ *l,* $a^{(l-1)}$ *là vector* chứa đầu ra của lớp trước đó, $\boldsymbol{b}_j^{(l)}$ là độ chệch.

Giá trị đầu ra của toàn bộ lớp thứ *l* sẽ là:

$$\boldsymbol{z}^{(l)} = \boldsymbol{W}^{(l)T}\boldsymbol{a}^{(l-1)} + b^{(l)}$$

Với $\boldsymbol{z}^{(l)}$ là vector giá trị đầu ra của lớp thứ *l,* $\boldsymbol{W}^{(l)T}$ là chuyển vị ma trận trọng số của lớp thứ *l,* $b^{(l)}$là vector độ chệch.

Nhìn chung, kiến trúc mạng Perceptron đa lớp thường sẽ bao gồm ít nhất 3 lớp: lớp đầu vào, một hoặc nhiều lớp ẩn, lớp đầu ra. Đối với bài toán phân loại chủ đề tin tức, lớp đầu vào của mạng có thể được xây dựng thông qua các phương pháp trích xuất đặc trưng đối với dữ liệu văn bản như TF-IDF, Word Embedding, … Hoặc một số phương pháp trích xuất đặc trưng khác. Lớp đầu ra sẽ bằng với số chủ đề cần được phân loại hoặc bằng 1 trong trường hợp phân loại nhị phân. Mạng MLP có thể có nhiều các lớp ẩn ở giữa. Các lớp ẩn theo thứ tự từ lớp đầu vào đến lớp đầu ra được đánh số thứ thự là *Lớp ẩn 1*, *Lớp ẩn 2*, …

c. Hàm kích hoạt

Hàm kích hoạt là một hàm được thêm vào mạng thần kinh nhân tạo để giúp mạng có thể tìm hiểu được các khuôn mẫu hay quy tắc phức tạp trong dữ liệu. Hàm kích hoạt giúp bổ sung tính phi tuyến cho mô hình, điều này là vô cùng quan trọng trong các mô hình học sâu nói chung và MLP nói riêng bởi vì nếu không có hàm kích hoạt, mô hình sẽ trở về dạng phân loại tuyến tính và khó có thể được sủ dụng trong các bài toán phức tạp hơn. Mô hình hồi quy Logistic chính là một trường hợp tiêu biểu. Với việc sủ dụng hàm kích hoạt Sigmoid, mô hình giúp ánh xạ đầu ra là một số thực với giá trị từ âm vô cùng đến dương vô cùng vào một không gian xác suất với giá trị từ 0 đến 1. Hàm kích hoạt có thể được viết như sau:

$$a^{(l)} = g(z^{(l)})$$

Cụ thể hơn, mỗi đầu ra của một Perceptron sẽ được tính là:

$$a_j^{(l)} = g(w_j^{(l)T}a^{(l-1)} + b_j^{(l)})$$

Khi hàm kích hoạt *g(.)* được áp dụng cho một ma trận (hoặc vector) trong mạng, ta hiểu rằng nó được áp dụng cho *từng thành phần của ma trận đó*. Sau đó các thành phần này được sắp xếp lại đúng theo thứ tự để được một ma trận có kích thước bằng với ma trận đầu vào. Nhờ có hàm kích hoạt, mạng MLP có thể "học" từ dữ liệu và cải thiện hiệu năng bằng cách liên tục cập nhật các tham số (trọng số và độ chệch) của các Perceptron trong mạng. Việc học hay huấn luyện mạng được thực hiện bằng cách tính toán đạo hàm có hướng/độ dốc (Gradient) của mỗi Perceptron, sau đó cập nhật các tham số dựa trên các đạo hàm này theo phương pháp lan truyền ngược.

Đối với bài toán phân loại chủ đề tin tức, hàm kích hoạt được sử dụng sẽ là hàm Softmax. Hàm Softmax được định nghĩa như sau:

$$a_i = \frac{\exp(z_i)}{\sum_{j=1}^{C} \exp(z_i)} \; \forall i = 1,2,3,\dots,C$$

Ở đây C chính là số lớp hay số chủ đề cần được phân loại. Giá trị đầu ra a_i của hàm Softmax chính là xác suất để một tài liệu thuộc vào chủ đề thứ *i*. Vì vậy a_i cần có giá trị lớn hơn 0 và tổng các a_i bằng 1. Bên cạnh đó, giá trị $z_i = w_i^T x$ càng lớn thì xác suất tài liệu rơi vào chủ đề i càng cao. Hàm $\exp(z_i)$ giúp thỏa mãn điều kiện này đồng thời đảm bảo giá trị đầu ra sẽ luôn dương.

Sau khi đã định nghĩa được hàm kích hoạt, ta có thể tính được đầu ra của mỗi điểm dữ liệu và sử dụng đầu ra này để tối ưu hóa các tham số của Perceptron nhằm cải thiện hiệu năng mô hình. Việc tối ưu hóa có thể được thực hiện bằng cách lấy đạo hàm của hàm mất mát và thực hiện cập nhật dựa trên giá trị của các đạo hàm. Vì là bài toán phân loại nên hàm mất mát của mạng sẽ là hàm Entropy chéo (Cross Entropy - CE) và được định nghĩa như sau:

$$J(\boldsymbol{W}; \boldsymbol{x}_i; \boldsymbol{y}_i) = \sum_{j=1}^{C} y_{ji} \log(a_{ji})$$

Với y_{ji} và a_{ji} lần lượt là phần tử thứ j của vector xác suất y_i – xác suất thực tế một tài liệu thuộc vào chủ đề *i* và a_i là xác suất dự đoán.

2.3.2.2. Mạng Nơron tích chập

a. Giới thiệu về mạng Nơron tích chập

Mạng thần kinh tích chập (CNN) vốn được ra đời nhằm mục đích giải quyết các tác vụ liên quan đến lĩnh vực thị giác máy tính, tuy nhiên sau này khi được áp dụng sang lĩnh vực xử lý ngôn ngữ tự nhiên và cụ thể là tác vụ phân loại văn bản, CNN đã được chứng minh là đạt được hiệu suất cao (Shaojie, J. Zico, & Vladlen, 2018), *(Nal, Edward, & Phil)*, *(Peng, et al., 2015)* kể cả đối với các tác vụ khác *(Ronan, et al.)*. Ngay cả khi mô hình chỉ bao gồm một lớp đơn giản.

b. Kiến trúc mạng Nơron tích chập

Nhìn chung, các kiến trúc mạng CNN hiện tại sẽ tiến hành phân loại văn bản theo các bước như sau (Yoav, A Primer on Neural Network Models for Natural Language Processing, 2016):

- Bước 1: Các bộ lọc tích chập 1 chiều được sử dụng làm bộ phát hiện ngram, mỗi bộ lọc chuyên về một họ ngram có liên quan chặt chẽ.
- Bước 2: Lớp gộp (Pooling layer) trích xuất các thông tin ngram liên theo thời gian quan để đưa ra quyết định.
- Bước 3: Phần còn lại của mạng phân loại văn bản dựa trên thông tin được trích xuất.

- **Lớp tích chập văn bản 1 chiều**

Đối với bộ tài liệu đầu đầu vào chứa n từ: $w_1, w_2, w_3, \dots w_n$, những từ này sẽ được biểu diễn dưới dạng vector số thực d chiều bởi lớp nhúng tuần tự, kết quả biểu diễn là các vector nhúng từ $\boldsymbol{w_1, w_2, w_3, \dots w_n} \epsilon R^d$. Tập các vector này tạo thành ma trận biểu diễn với kích thước $d \times n$ và sẽ được đưa vào lớp tích chập và được áp dụng một bộ lọc nhân tích chập nhằm nắm bắt các liên kết cú pháp hoặc ngữ nghĩa giữa các cụm từ cách xa nhau trong văn bản. Với một chuỗi từ với mỗi từ tương ứng với một vector nhúng từ d chiều, Một tích chập 1 chiều với chiều rộng k là kết quả của việc di chuyển một cửa sổ trượt có kích thước k qua câu và áp dụng cùng một bộ lọc tích chập cho mỗi cửa sổ trong chuỗi, tức là, một tích vô hướng giữa phép nối các vectơ nhúng trong một cửa sổ đã cho và một ma trận trọng số U, sau đó thường được theo sau bởi hàm kích hoạt phi tuyến g.

Xem xét một cửa sổ của các từ $w_i, \dots w_{i+k}$, vectơ nối của cửa sổ thứ i là:

$$x_i = [w_i, w_{i+1}, \dots w_{i+k}] \epsilon R^{k \times d}$$

Các bộ lọc tích chập được áp dụng cho từng cửa sổ, kết quả tạo ra là các giá trị vô hướng r_i, mỗi giá trị cho cửa sổ thứ i:

$$r_i = g(x_i \cdot U + b) \epsilon R$$

- **Lớp gộp**

Phép gộp được sử dụng để kết hợp các vector tạo ra từ các cửa sổ tích chập khác nhau thành một vector một chiều. Điều này được thực hiện lại bằng cách lấy giá trị lớn nhất hoặc giá trị trung bình quan sát được trong vectỏ kết quả từ các tích chập. Lý tưởng nhất là vectỏ này sẽ ghi lại các đặc trưng phù hợp nhất của câu/tài liệu. Các thao tác từ lớp nhúng cho đến lớp gộp có thể được xem như là một bộ trích xuất đặc trưng cho tài liệu. Qua đó thay vì xem mỗi từ như một đặc trưng theo các phương pháp truyền thống, bộ trích xuất này sẽ chọn ra các đặc trưng tốt nhất và biểu diễn trong không gian với số chiều thấp. Từ đó các biểu diễn này có thể tiếp tục được học bởi mạng Nơron truyền thẳng mà vẫn mang lại kết quả khả quan.

2.3.2.3. Mạng bộ nhớ ngắn – dài hạn

a. Giới thiệu về mạng bộ nhớ ngắn – dài hạn

Mạng bộ nhớ dài-ngắn hạn (Long-Short Term Memory), hay còn được viết tắt là LSTM, là một dạng mạng Nơron hồi quy (Recurrent Neural Network - RNN). Mạng LSTM được thiết kế đặc biệt để có thể xử lý dữ liệu tuần tự, chẳng hạn như chuỗi thời gian, văn bản hay giọng nói. Mạng LSTM có khả năng học các phụ thuộc dài hạn trong dữ liệu tuần tự, giúp chúng phù hợp với các tác vụ như phiên dịch ngôn ngữ, nhận dạng giọng nói và dự báo giá trị chuỗi thời gian.

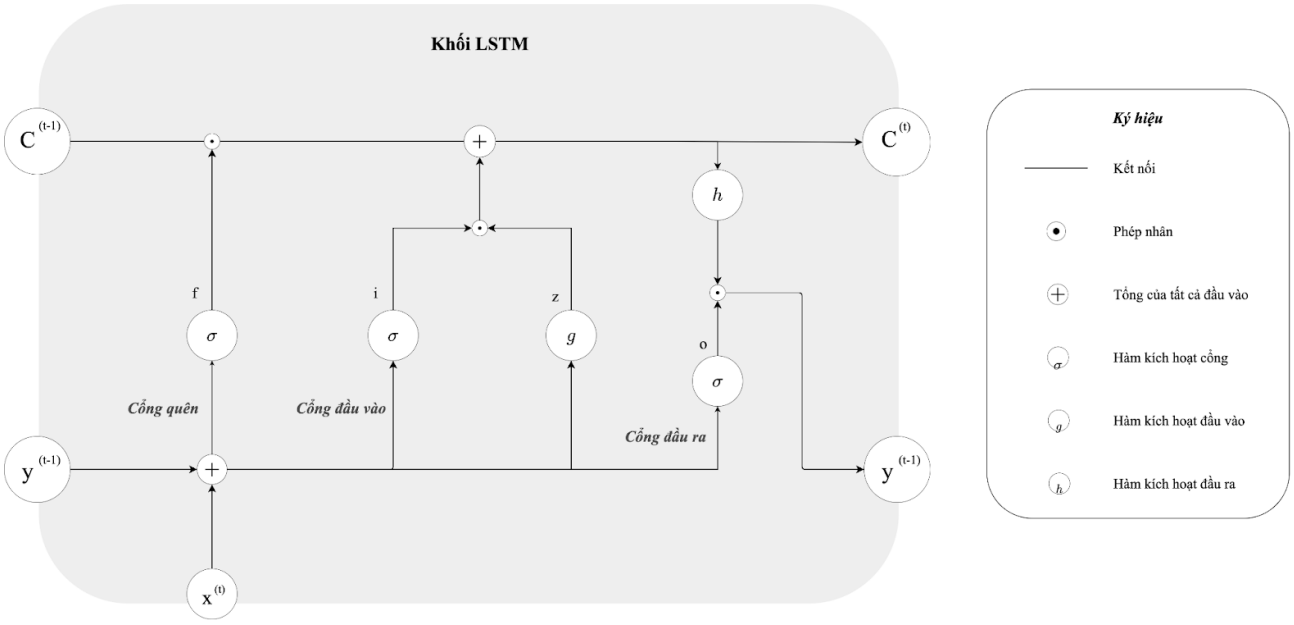
Đối với dữ liệu tuần tự, việc huấn luyện các mô hình truyền thống như mạng Nơron sâu (Deep Neural) hoặc mạng Nơron hồi quy để học các phụ thuộc dài hạn (long-term dependencies) bằng phương pháp lan truyền ngược thường gặp khó khăn và tỏ ra kém hiệu quả. Vấn đề này xuất phát từ hiện tượng triệt tiêu hoặc bùng nổ đạo hàm có hướng (Bengio et al., 1994, Hochreiter et (Hochreiter, Bengio, Frasconi, Schmidhuber, & others), 2001). Nhằm vượt qua hạn chế này khi học các phụ thuộc dài hạn, mô hình LSTM (Hochreiter and Schmidhuber 1997a) đã được giới thiệu. Hiệu năng học của mô hình LSTM đã tạo ra ảnh hưởng lên nhiều lĩnh vực khác nhau, cả về mặt lý thuyết và thực tiễn. Chính vì vậy, có thể nói mô hình LSTM là một mô

hình tân tiến và đạt được trạng thái State-of-the-art. Một số ứng dụng thành công của mô hình LSTM có thể kể đến như: hệ thống nhận diện giọng nói của Google, cải thiện hiệu năng dịch máy của Google dịch, Amazon sử dụng LSTM để cải thiện hiệu năng của trợ lý ảo Alexa, ...

b. Kiến trúc mạng bộ nhớ ngắn – dài hạn

Nhìn chung, một khối LSTM hay một đơn vị LSTM (unit) sẽ bao gồm các thành phần chính là: một ô (Cell), một cổng đầu vào (Input Gate), một cổng đầu ra (Output Gate) và cổng quên (Forget Gate). Các ô trong mạng LSTM có nhiệm vụ chính là ghi nhớ các giá trị trong khoảng thời gian tùy ý. Những giá trị này sẽ được điều chỉnh bởi các cổng dựa trên luồng thông tin liên quan đến ô.

Một cách ngắn gọn, kiến trúc mạng LSTM bao gồm một tập hợp các mạng hồi quy (Recurrent) con, các mạng con này còn được gọi là khối bộ nhớ. Ý tưởng đằng sau khối bộ nhớ là duy trì trạng thái của khối theo thời gian và điều chỉnh luồng thông tin bằng cách sử dụng các đơn vị cổng phi tuyến. Hình dưới đây minh họa kiến trúc tổng quan của mạng LSTM, bao gồm đầu vào x(t), đầu ra y(t), các hàm kích hoạt. Đầu ra của khối được kết nối hồi quy trở lại với đầu vào của khối và tất cả các cổng.



Hình 5: Kiến trúc mạng LSTM

Giả sử một mạng bao gồm N khối xử lý và M đầu vào. Quá trình chuyển tiếp dữ liệu trong mạng được thực hiện như mô tả dưới đây.

- **Đầu vào khối**

Bước này dành cho việc cập nhật thành phần đầu vào khối bằng cách kết hợp đầu vào hiện tại $x^{(t)}$ và đầu ra $y^{(t-1)}$ của khối LSTM đó trong lần lặp cuối cùng trước đó. Điều này có thể được thực hiện như mô tả dưới đây:

$$z^{(t)} = g(W_z x^{(t)} + R_z y^{(t-1)} + b_z)$$

Trong đó W_z và R_z lần lượt là các ma trận trọng số gắn liền với $x^{(t)}$ và $y^{(t-1)}$, b_z là vector chệch (bias vector). Giá trị $z^{(t)}$ có thể được hiểu là các giá trị tiềm năng có thể được giữ lại trong dài hạn.

- **Cổng đầu vào**

Tại bước này, cổng đầu vào được cập nhật bằng cách kết hợp giá trị đầu vào hiện tại là $x^{(t)}$, đầu ra $y^{(t-1)}$ và trạng thái ô $c^{(t-1)}$ của khối LSTM đó trong lần lặp cuối cùng trước đó:

$$i^{(t)} = \sigma(W_i x^{(t)} + R_i y^{(t-1)} + p_i \odot c^{(t-1)} + b_i)$$

Với $\odot$ đại diện cho tích Hadamard giữa hai vector, W_i , R_i và p_i lần lượt là các ma trạna trọng số gắn liền với $x^{(t)}$, $y^{(t-1)}$ và $c^{(t-1)}$. Giá trị kích hoạt đầu vào $i^{(t)}$ sẽ quyết định lưu trữ bao nhiêu giá trị tiềm năng $z^{(t)}$ trong dài hạn.

Bằng cách sử dụng các giá trị tiềm năng $z^{(t)}$ và giá trị kích hoạt đầu vào $i^{(t)}$, lớp LSTM có thể xác định lượng thông tin sẽ được giữ lại trong các trạng thái ô $c^{(t)}$ của mạng.

- **Cổng quên**

Tại bước này, khối LSTM xác định thông tin nào sẽ bị xóa khỏi trạng thái ô $c^{(t-1)}$ trước đó của nó. Do đó, các giá trị kích hoạt $f^{(t)}$ của các cổng quên ở bước thời gian t hiện tại sẽ được tính dựa trên giá trị đầu vào hiện tại $x^{(t)}$, đầu ra $y^{(t-1)}$ và giá trị ô $c^{(t-1)}$của các ô nhớ ở bước thời gian (t - 1) trước đó:

$$f^{(t)} = \sigma(W_f x^{(t)} + R_f y^{(t-1)} + p_f \odot c^{(t-1)} + b_f)$$

Với W_f**,** R_f và p_f lần lượt là các ma trận trọng số gắn liền với $x^{(t)}$, $y^{(t-1)}$ và $c^{(t-1)}$.

- **Giá trị ô**

Sau khi đã xác định được lượng thông tin sẽ bị xóa (hoặc "quên") và lượng thông tin sẽ được giữ lại, khối LSTM sẽ tiến hành tính toán giá trị ô hiện tại dựa trên đầu vào

khối $z^{(t)}$, giá trị kích hoạt đầu vào $i^{(t)}$, giá trị kích hoạt cổng quên $f^{(t)}$ và trạng thái ô trước đó $c^{(t-1)}$. Điều này có thể được thực hiện như sau:

$$c^{(t)} = z^{(t)} \odot i^{(t)} + c^{(t-1)} \odot f^{(t)}$$

Kết quả của phép tính $c^{(t-1)} \odot f^{(t)}$ chính là lượng thông tin sẽ bị xóa khỏi trạng thái ô, trong khi kết quả của phép tính $z^{(t)} \odot i^{(t)}$chính là lượng thông tin sẽ được giữ lại trong dài hạn.

- **Cổng đầu ra**

Tại bước này, giá trị cổng đầu ra sẽ được tính dựa trên giá trị đầu vào hiện tại $x^{(t)}$, đầu ra $y^{(t-1)}$, và giá trị ô $c^{(t-1)}$ của khối LSTM đó trong lần lặp cuối cùng trước đó:

$$o^{(t)} = \sigma(W_o x^{(t)} + R_o y^{(t-1)} + p_o \odot c^{(t-1)} + b_f)$$

Với W_o, R_o và p_o lần lượt là các ma trận trọng số gắn liền với $x^{(t)}$, $y^{(t-1)}$ và $c^{(t-1)}$.

- **Đầu ra khối**

Cuối cùng, khối LSTM sẽ tính giá trị đầu ra khối dựa trên trạng thái ô hiện tại $c^{(t-1)}$ và giá trị cổng đầu ra $o^{(t)}$:

$$y^{(t)} = h(c^{(t)}) \odot o^{(t)}$$

$y^{(t)}$ chính là giá trị đầu ra cuối cùng của toàn bộ khối LSTM, đồng thời cũng là giá trị ngắn hạn mới sẽ được dùng trong việc tính toán tại các bước thời gian tiếp theo.

2.4. Đánh giá mô hình phân loại chủ đề tin tức

2.4.1. Thước đo đánh giá

Để đánh giá hiệu năng của một hệ thống phân loại chủ đề tin tức một cách toàn vẹn, việc chọn một bộ các thước đo đánh giá là cần thiết vì một thước đo đơn lẻ không thể nào phản ánh hết toàn bộ hiệu năng của hệ thống trên nhiều phương diện khác nhau. Bên cạnh đó, một hệ thống phân loại có thể được tối ưu hóa hiệu năng dựa trên một chỉ số bằng cách hi sinh những chỉ số khác và ngược lại.

Các thước đo được áp dụng để đánh giá một bộ phân loại chủ đề tin tức bao gồm: Độ phủ (Recall), độ chính xác (Precision), thước đo F (F-Measure), trung bình vi mô, trung bình vĩ mô. Những thước đo này có thể được tính toán dựa trên ma trận nhầm lẫn – một dạng bảng tổng hợp số lượng các trường hợp phân loại như sau:

+ True Positive (TP): Giá trị thực tế là Positive và giá trị dự đoán cũng là Positive.

+ True Negative (TN): Giá trị thực tế là Negative và giá trị dự đoán cũng là Negative.

+ False Positive (FP): Giá trị thực tế là Negative và giá trị dự đoán là Positive. Còn gọi là sai lầm loại 1.

+ False Negative (FN): Giá trị thực tế là Positive và giá trị dự đoán là Negative. Còn gọi là sai lầm loại 2.

- **Độ chuẩn xác (Accuracy)**

Độ chuẩn xác đề cập đến tỷ lệ số mẫu được phân lớp đúng trong toàn bộ tập dữ liệu. Thước đo này giúp ta đánh giá hiệu quả dự báo của mô hình trên một bộ dữ liệu. Độ chuẩn xác càng cao thì mô hình của chúng ta càng chuẩn xác:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} = \frac{TP + TN}{Tổng\ số\ mẫu}$$

- **Độ chính xác (Precision)**

Độ chính xác cho biết trong số m mẫu được phân vào lớp i thì có tỷ lệ bao nhiêu mẫu được phân loại đúng. Độ chính xác được định nghĩa như sau:

$$Precision = \frac{TP}{TP + FP} = \frac{TP}{Tổng\ số\ mẫu\ được\ phân\ vào\ lớp\ i}$$

- **Độ phủ (Recall)**

Độ phủ cho biết tỷ lệ mẫu được phân lớp đúng. Độ phủ được định nghĩa như sau:

$$Recall = \frac{TP}{TP + FN} = \frac{TP + TN}{Tổng\ số\ mẫu\ thực\ tế\ thuộc\ lớp\ i}$$

- **Thước đo F (F measure)**

Thước đo F là một trong những thước đo đánh giá tổng hợp phổ biến nhất được sử dụng để đánh giá hiệu năng của một bộ phân loại. Thước đo F được định nghĩa như sau:

$$F_\beta = \frac{(1 + \beta)^2\ (Precision\ \times\ Recall)}{\beta^2 \times Precision + Recall}$$

Với β là tham số được sử dụng để cân bằng giữa độ chính xác và độ phủ. Thông thường với $\beta = 1$, trọng số giữa độ chính xác và độ phủ là ngang nhau và thước đo F_1có thể được viết là:

$$F_1 = \frac{(1+1)^2\,(Precision \times Recall)}{1^2 \times Precision + Recall} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$
$$= \frac{2TP}{2TP + FP + FN}$$

2.4.2. Trung bình vi mô, trung bình vĩ mô

Trong trường hợp các bộ phân loại nhị phân được sử dụng để phân loại bộ dữ liệu đa lớp, ta sẽ cần một hướng tiếp cận khác so với phân loại nhị phân truyền thống để có thể xây dựng được bộ phân loại cho nhiều lớp và đánh giá các bộ phân loại đó. Một trong những phương pháp phổ biến là phân loại One-vs-Rest. Bằng cách xây dựng C bộ phân loại tương ứng với C lớp trong bộ dữ liệu, chẳng hạn nếu dữ liệu có 10 lớp khác nhau thì ta sẽ xây dựng 10 bộ phân loại, với bộ phân loại của lớp thứ nhất giúp phân biệt các điểm dữ liệu thuộc lớp thứ nhất so với các lớp còn lại, bộ phân loại của lớp thứ hai giúp phân biệt các điểm dữ liệu thuộc lớp thứ hai so với các lớp còn lại,… Cứ thế đến bộ dữ liệu của lớp thứ 10.

Như vậy với mỗi lớp, ta coi dữ liệu thuộc lớp đó có nhãn là *positive*, tất cả các dữ liệu còn lại có nhãn là *negative*. Sau đó, các giá trị Precision, Recall, thước đó F có thể được áp dụng lên từng lớp. Với mỗi lớp, ta sẽ nhận được một cặp giá trị Precision và Recall tương ứng. Sau khi đã có các cặp giá trị Precision và Recall cho các lớp, ta có thể tính trung bình vi mô cho mỗi thước đo như sau:

$$micro - average\ Precision = \frac{\sum_{c=1}^{C} TP_c}{\sum_{c=1}^{C}(TP_c + FP_c)}$$

$$micro - average\ Recall = \frac{\sum_{c=1}^{C} TP_c}{\sum_{c=1}^{C}(TP_c + FN_c)}$$

Trung bình vĩ mô sẽ là trung bình cộng của các thước đo theo lớp. Hệ quả là điểm trung bình vi mô gán các trọng số bằng nhau cho mọi tài liệu và nó được coi là điểm trung bình trên mỗi tài liệu. Mặt khác, điểm trung bình vĩ mô gán trọng số bằng nhau cho từng chủ đề mà không tính đến tần suất và do đó, nó là điểm trung bình cho mỗi chủ đề.

Chương 3. Cài đặt, thử nghiệm mô hình

3.1. Môi trường và công cụ

3.1.1. Thư viện Numpy

Numpy, viết tắt của Numerical Python, là một dự án mã nguồn mở được tạo ra vào năm 2005 nhằm phục vụ cho các mục đích tính toán số học trên Python và đặc biệt phù hợp với các đối tượng là mảng n chiều. Một số chức năng chính của Numpy có thể kể đến như:

+ Ndarray: một cấu trúc dữ liệu được dùng để lưu trữ các mảng n chiều. Đồng thời, ndarray cũng cung cấp khả năng thực hiện các phép toán số học được vectơ hóa và các phép toán broadcasting trên các mảng này.

+ Các hàm toán học tiêu chuẩn cho các phép toán nhanh trên toàn bộ mảng dữ liệu mà không cần phải sử dụng đến vòng lặp.

+ Các công cụ để đọc/ghi dữ liệu mảng vào đĩa và làm việc với các tệp được ánh xạ từ bộ nhớ.

+ Các chức năng liên quan đến đại số tuyến tính, khởi tạo số ngẫu nhiên và biến đổi Fourier.

+ Các công cụ tích hợp mã được viết bằng ngôn ngữ C, C++ hoặc Fortran.

3.1.2. Thư viện Pandas

Pandas là một thư viện mã nguồn mở giúp thực hiện các tác vụ phân tích dữ liệu một cách dễ dàng và trực quan. Được phát triển bởi Wes McKinney vào năm 2008 do nhu cầu về một công cụ phân tích định lượng mạnh mẽ và linh hoạt, Pandas đã phát triển thành một trong những thư viện Python phổ biến nhất hiện nay với một cộng đồng đóng góp cực kỳ tích cực. Với Pandas, người dùng có thể thực hiện những tác vụ như:

+ Phân tích dữ liệu chuỗi thời gian.
+ Lọc dữ liệu theo điều kiện.
+ Trực quan hóa dữ liệu.
+ Xử lý dữ liệu bị thiếu.
+ Tổng hợp dữ liệu.

3.1.3. Thư viện Scikit-Learn

Scikit - Learn là một thư viện Python mã nguồn mở với khả năng triển khai một loạt các thuật toán học máy, tiền xử lý, xác thực chéo và trực quan hóa dữ liệu dưới một giao diện đồng nhất. Với Scikit-Learn, người dùng có thể thực hiện những tác vụ như:

+ Tiền xử lý dữ liệu để trích xuất và chuẩn hóa đặc trưng trong quá trình phân tích dữ liệu.
+ Xây dựng các mô hình phân lớp, hồi quy dữ liệu.
+ Xây dựng các mô hình phân cụm dữ liệu.
+ Khả năng cung cấp các công cụ so sánh, xác thực và lựa chọn các tham số tối ưu để có thể lựa chọn và sử dụng mô hình thích hợp.
+ Trực quan hóa dành cho máy học cho phép vẽ biểu đồ và điều chỉnh hình ảnh nhanh chóng.

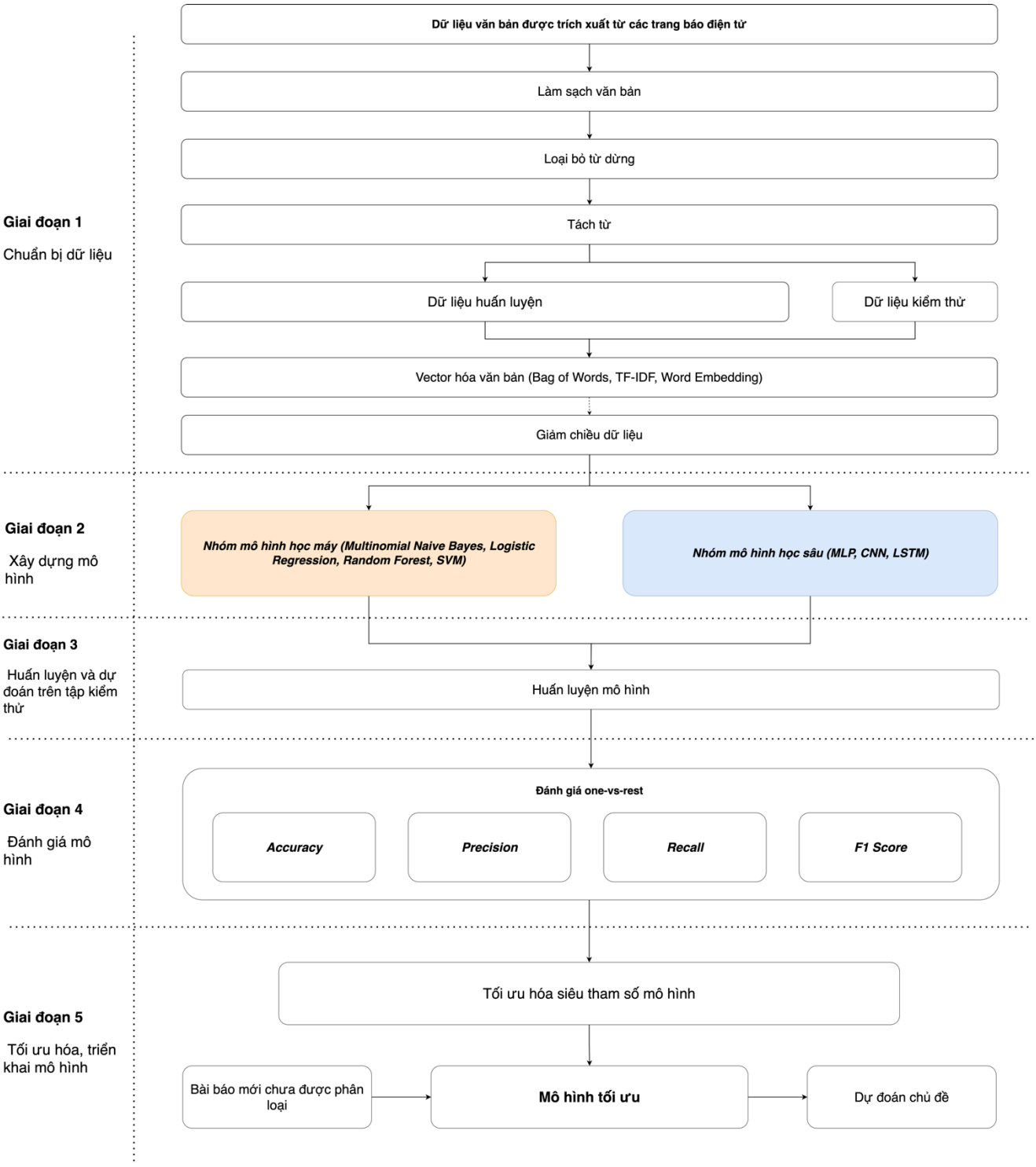
3.1.4. Thư viện Keras

Keras là một API cấp cao được phát triển bởi Google và được thiết kế cho Python để triển khai mạng nơ-ron dễ dàng hơn. **Keras** có thể chạy trên các thư viện và khung công tác như TensorFlow, Theano, PlaidML, MXNet, CNTK. Keras rất thân thiện với người mới bắt đầu vì cấu trúc tối thiểu của nó cung cấp cách tạo ra các mô hình học sâu một cách dễ dàng và gọn ghẽ dựa trên TensorFlow hoặc Theano. **Keras đã được TensorFlow thông qua làm API cấp cao chính thức** của mình. Khi được nhúng vào TensorFlow, nó cung cấp các mô-đun có sẵn cho tất cả các tính toán mạng nơ-ron và do đó có thể thực hiện học sâu rất nhanh. TensorFlow rất linh hoạt và lợi ích chính là tính toán phân tán. Bạn có thể linh hoạt và có thể kiểm soát ứng dụng của mình, thực hiện ý tưởng của bạn trong thời gian ngắn, sử dụng Keras, trong khi tính toán liên quan đến tensors, đồ thị tính toán, phiên, v.v. có thể được tùy chỉnh bằng cách sử dụng Tensorflow Core API.

Các tiện ích và khả năng của Keras có thể kể đến như:

- Hỗ trợ nhiều nền tảng, phụ trợ, mô hình mạng nơ-ron.
- Mạng nơ-ron Keras được viết bằng Python.
- Chạy trơn tru trên cả CPU và GPU.
- Hỗ trợ cộng đồng rộng lớn với nhiều người dùng và tài liệu, sẵn sàng trợ giúp hơn với các khuôn khổ học sâu khác.
- Keras là sự lựa chọn của nhiều tên tuổi lớn như Netflix, Uber, Square, Yelp, v.v. cho các sản phẩm của họ trong phạm vi công cộng.

3.2. Quy trình thực nghiệm

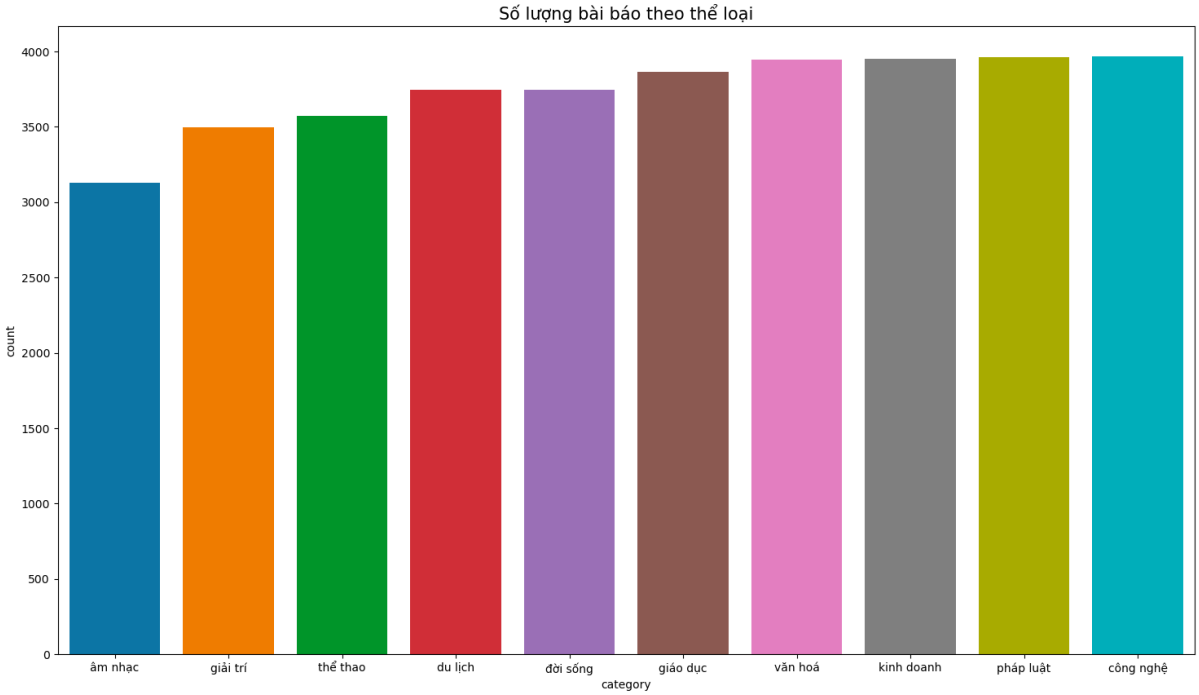


Hình 6: Quy tình thực nghiệm xây dựng hệ thống phân loại chủ đề tin tức

3.2.1. Chuẩn bị dữ liệu

3.2.1.1. Mô tả bộ dữ liệu

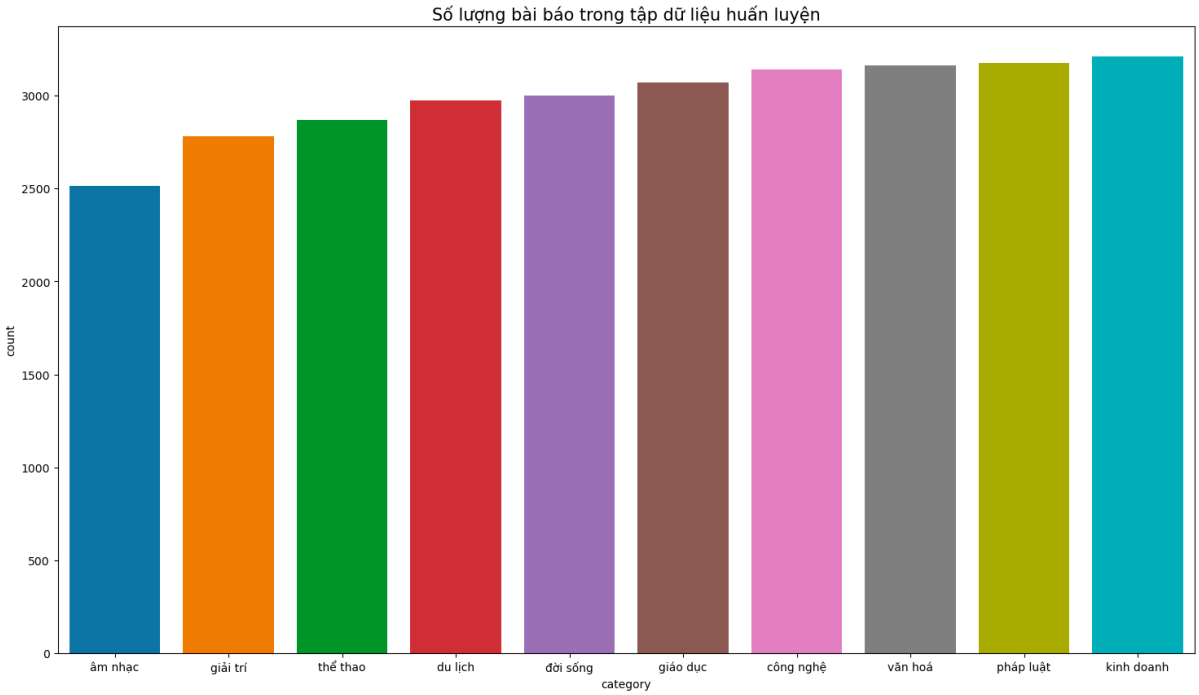
Bộ dữ liệu bao gồm 41276 bài báo điện tử được thu thập từ các trang báo trực tuyến, bao gồm: VnExpress, Báo Thanh Niên, Báo Nhân Dân, Báo điện tử Tiền Phong, … Được chia thành 10 chủ đề khác nhau: Âm nhạc, Du lịch, Công nghệ, Văn hóa, Giáo dục, Đời sống, Pháp luật, Kinh doanh, Giải trí, Thể thao.



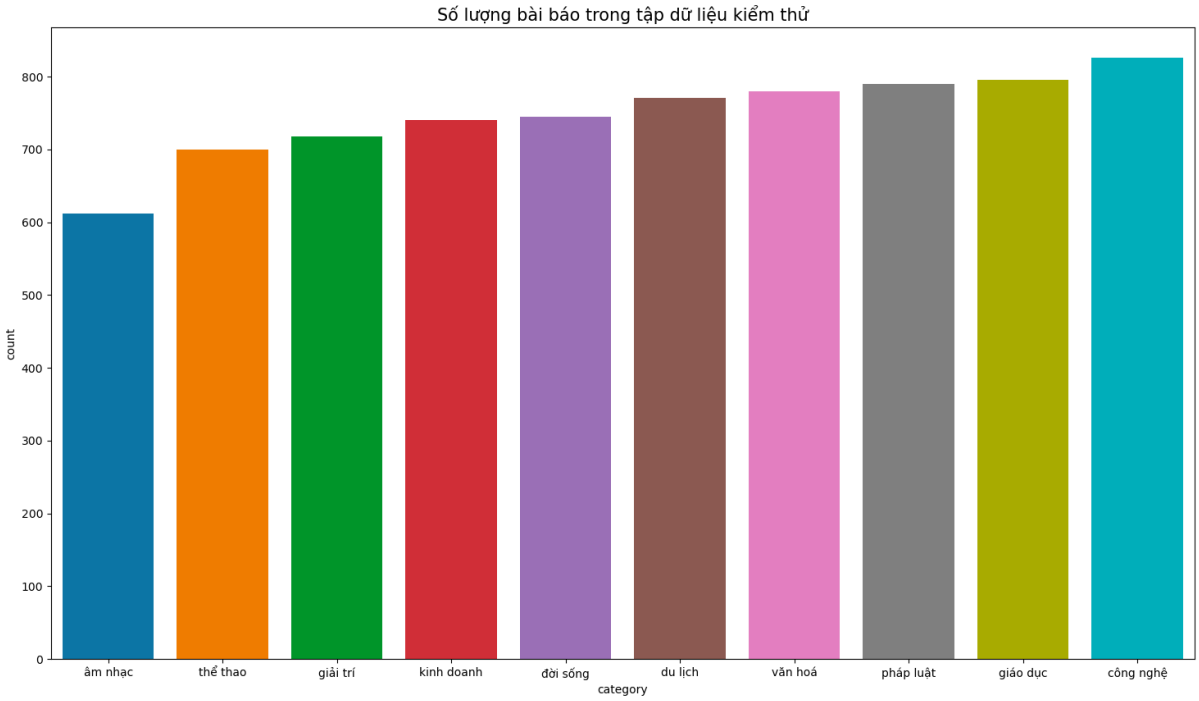
Hình 7: Số lượng bài báo theo chủ đề

3.2.1.2. Chuẩn bị dữ liệu huấn luyện và kiểm thử

Từ bộ dữ liệu ban đầu, 2 tập dữ liệu là tập huấn luyện (training set) và tập kiểm thử (testing set) sẽ được tạo ra với mục đích tương ứng là xây dựng mô hình phân lớp dựa trên dữ liệu và đánh giá hiệu năng của mô hình phân lớp đó. Cụ thể, tập huấn luyện được tạo ra bằng cách lấy ngẫu nhiên 80% điểm dữ liệu (29906 bài báo) trong bộ dữ liệu ban đầu, 20% còn lại sẽ là tập kiểm thử (7477 bài báo).



Hình 8: Số lượng bài báo trong tập dữ liệu huấn luyện



Hình 9: Số lượng bài báo trong tập dữ liệu kiểm thử

Sau đó, cả tập dữ liệu huấn luyện và kiểm thử sẽ được tách từ dựa trên 2 phương pháp là tách từ đơn (1 từ chỉ có 1 tiếng duy nhất) và tách từ ghép (1 từ có thể có 1 hoặc nhiều tiếng):

+ Với phương pháp tách từ đơn, số lượng từ vựng thu được từ tập dữ liệu là 95167 từ. Như vậy ma trận biểu diễn văn bản sẽ có 95167 đặc trưng.

+ Với phương pháp tách từ ghép, số lượng từ vựng thu được từ tập dữ liệu là 99362 từ. Như vậy ma trận biểu diễn văn bản sẽ có 99362 đặc trưng.

Tương ứng với mỗi phương pháp tách từ, văn bản trong hai tập dữ liệu sẽ tiếp tục được vector hóa với kỹ thuật Túi Từ và TF-IDF nhằm mục đích thực nghiệm và so sánh.

3.2.2. Xây dựng, huấn luyện và đánh giá mô hình

Dữ liệu huấn luyện sau khi được vector hóa sẽ được sử dụng để lần lượt "học" 4 mô hình là Hồi Quy Logistic, Rừng Ngẫu Nhiên, Naïve Bayes Đa thức và Máy Vector Hỗ Trợ. Cả 4 mô hình sẽ được khởi tạo với bộ siêu tham số mặc định. Sau đó lần lượt mỗi mô hình sẽ được thực nghiệm trên tập dữ liệu huấn luyện và kiểm thử với các trường hợp sau:

+ Văn bản được tách từ đơn và vector hóa với kỹ thuật Túi Từ.
+ Văn bản được tách từ đơn và vector hóa với kỹ thuật TF-IDF.
+ Văn bản được tách từ ghép và vector hóa với kỹ thuật Túi Từ.
+ Văn bản được tách từ ghép và vector hóa với kỹ thuật TF-IDF.

Mỗi mô hình sau khi đã được huấn luyện trên tập dữ liệu huấn luyện sẽ được đánh giá hiệu năng trên tập dữ liệu kiểm thử. Các bài báo trong tập dữ liệu kiểm thử sẽ được xem là chưa được phân loại chủ đề và mô hình có nhiệm vụ dự đoán chủ đề đó. Chủ đề dự đoán sẽ được so sánh với chủ đề ban đầu của bài báo để tính toán các điểm số đánh giá cần thiết.

3.2.3. Tối ưu hóa mô hình

Mô hình với bộ điểm số đánh giá tốt nhất sẽ được chọn để tiếp tục tối ưu hóa các siêu tham số. Một bộ siêu tham số được tối ưu hóa đồng nghĩa với một chiến lược huấn luyện mà với nó, mô hình có thể mang lại kết quả đánh giá tốt nhất.

3.3. Cài đặt mô hình

3.3.1. Tách từ đơn

- **Count Vectorizer**

Công cụ Count Vectorizer của thư viện Scikit-Learn được sử dụng để vector hóa văn bản đã được tách từ đơn theo phương pháp Túi Từ. Tiến hành huấn luyện các bộ phân

loại trên tập huấn luyện và đánh giá hiệu năng với tập kiểm thử, thu được kết quả đánh giá như sau:

Mô hình	**Precision**	**Recall**	**F1 Score**	**% Accuracy**
Multinomial Naive Bayes	0.78	0.75	0.75	76.06
Logistic Regression	0.86	0.86	0.86	86.38
Random Forest	0.87	0.87	0.87	87.59
Support Vector Machine	0.86	0.86	0.86	86.47

Bảng 1: Kết quả kiểm thử mô hình với phương pháp tách từ đơn và Count Vectorizer

- **TF-IDF Vectorizer**

Công cụ TF-IDF Vevtorizer của thư viện Scikit-Learn với phương pháp lấy trọng số từ TF-IDF được sử dụng để vector hóa văn bản đã được tách từ đơn. Tiến hành huấn luyện các bộ phân loại trên tập huấn luyện và đánh giá hiệu năng với tập kiểm thử, thu được kết quả đánh giá như sau:

Mô hình	**Precision**	**Recall**	**F1 Score**	**% Accuracy**
Multinomial Naive Bayes	0.81	0.81	0.80	81.06
Logistic Regression	0.84	0.84	0.84	84.93
Random Forest	0.87	0.87	0.87	87.79
Support Vector Machine	0.87	0.88	0.87	88.0

Bảng 2: Kết quả kiểm thử mô hình với phương pháp tách từ đơn và TF-IDF

3.3.2. Tách từ đa âm tiết

- **Count Vectorizer**

Văn bản được tách từ ghép và vector hóa với Count Vectorizer. Tiến hành huấn luyện các bộ phân loại trên tập huấn luyện và đánh giá hiệu năng với tập kiểm thử, thu được kết quả đánh giá như sau:

Mô hình	Precision	Recall	F1 Score	% Accuracy
Multinomial Naive Bayes	0.79	0.76	0.76	76.65
Logistic Regression	0.87	0.87	0.87	87.24
Random Forest	0.88	0.88	0.87	88.14
Support Vector Machine	086	0.86	0.86	86.87

Bảng 3: Kết quả kiểm thử mô hình với phương pháp tách từ đa âm tiết và Count Vectorizer

- **TF-IDF Vectorizer**

Văn bản được tách từ ghép và vector hóa với TF-IDF Vectorizer. Tiến hành huấn luyện các bộ phân loại trên tập huấn luyện và đánh giá hiệu năng với tập kiểm thử, thu được kết quả đánh giá như sau:

Mô hình	Precision	Recall	F1 Score	% Accuracy
Multinomial Naive Bayes	0.82	0.82	0.82	82.71
Logistic Regression	0.85	0.85	0.85	85.66
Random Forest	0.87	0.88	0.87	88.08
Support Vector Machine	0.88	0.88	0.88	88.69

Bảng 4: Kết quả kiểm thử với phương pháp tách từ đa âm tiết và TF-IDF

3.3.3. Tinh chỉnh siêu tham số mô hình học máy

Với các kết quả như trên, hai mô hình với hiệu suất tốt nhất là mô hình hồi quy Logistic và mô hình SVM đã được lựa chọn để tiếp tục tinh chỉnh siêu tham số. Kết quả sau khi tinh chỉnh thu được như sau:

Siêu tham số	Lựa chọn
C	10
Penalty	l2
Solver	newton-cg

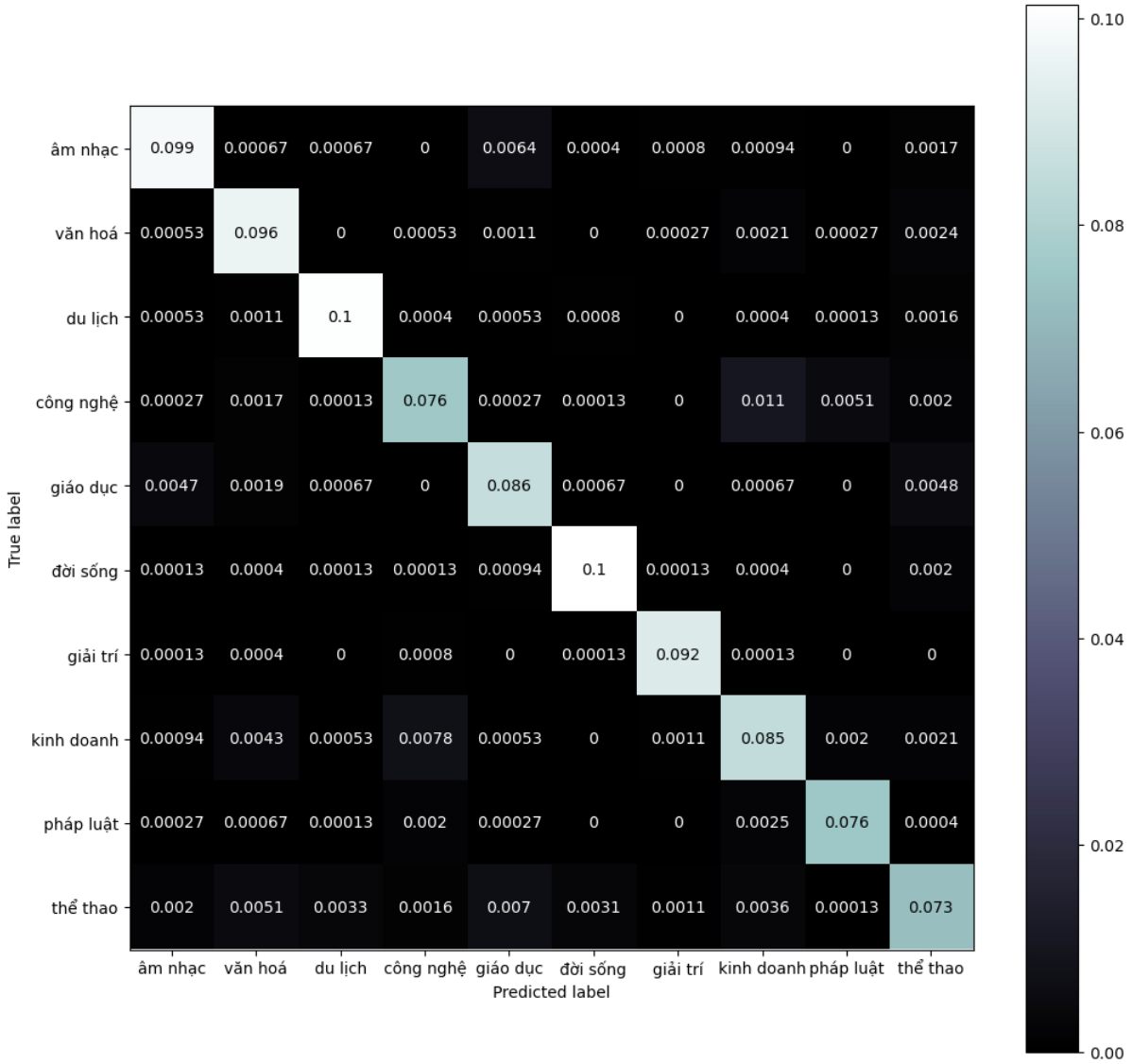
Bảng 5: Tham số tối ưu cho mô hình Hồi quy Logistic

Siêu tham số	Lựa chọn
C	1
Degree	10
Gamma	Auto
Kernel	Linear

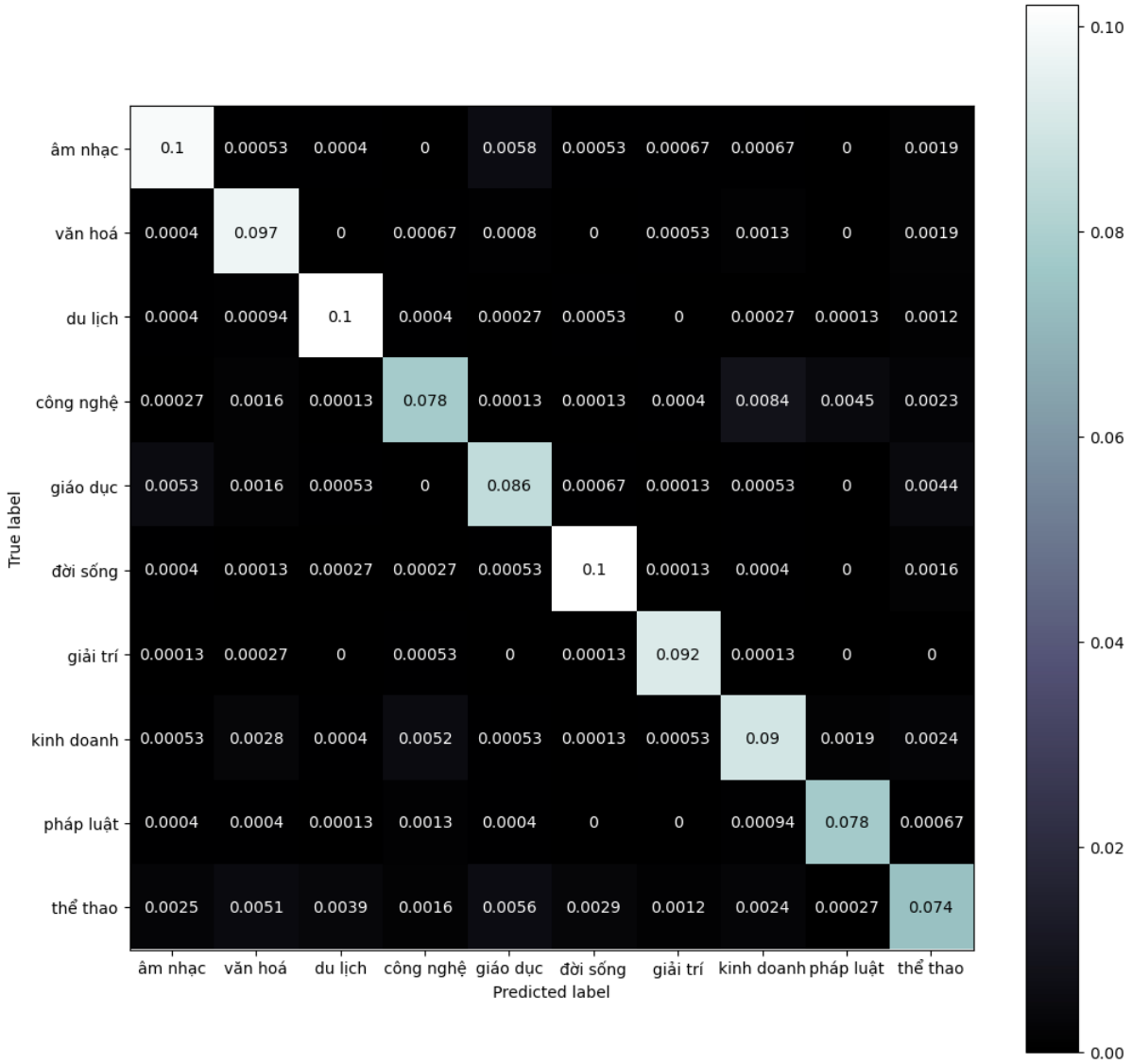
Bảng 6: Tham số tối ưu cho mô hình SVM

Mô hình	Precision	Recall	F1 Score	% Accuracy
Logistic Regression	0.82	0.76	0.78	89.0
Support Vector Machine	0.90	0.90	0.90	90.0

Bảng 7: Kết quả kiểm thử với mô hình đã được tối ưu



Hình 10: Ma trận nhầm lẫn của mô hình Hồi quy Logistic

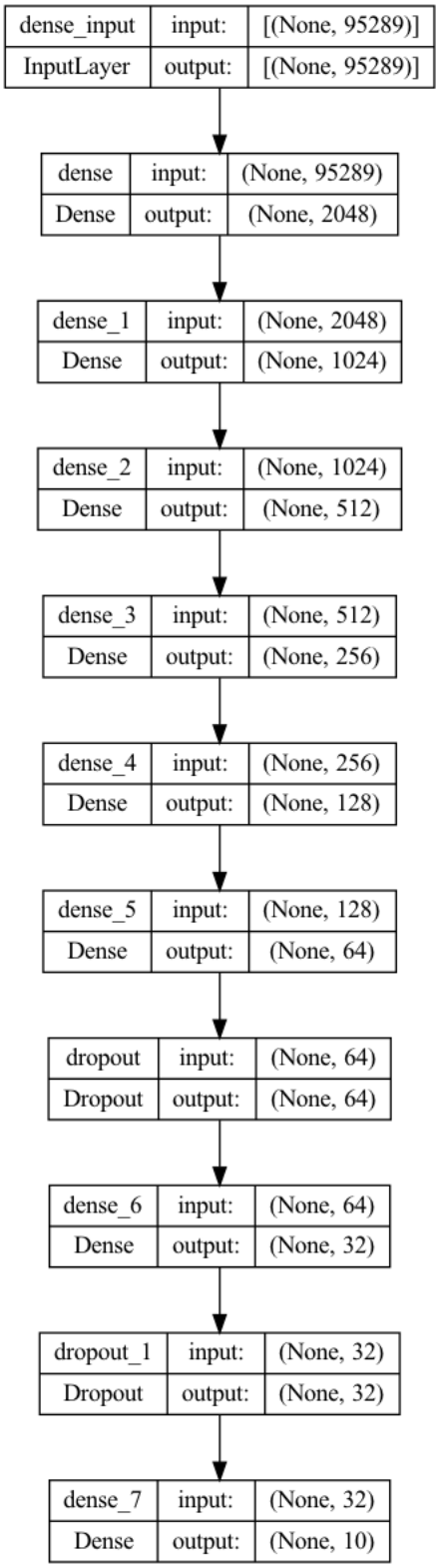


Hình 11: Ma trận nhầm lẫn của mô hình Máy Vector Hỗ Trợ

3.3.4. Mô hình học sâu

3.3.4.1. Perceptron đa lớp

- **Kiến trúc mô hình**

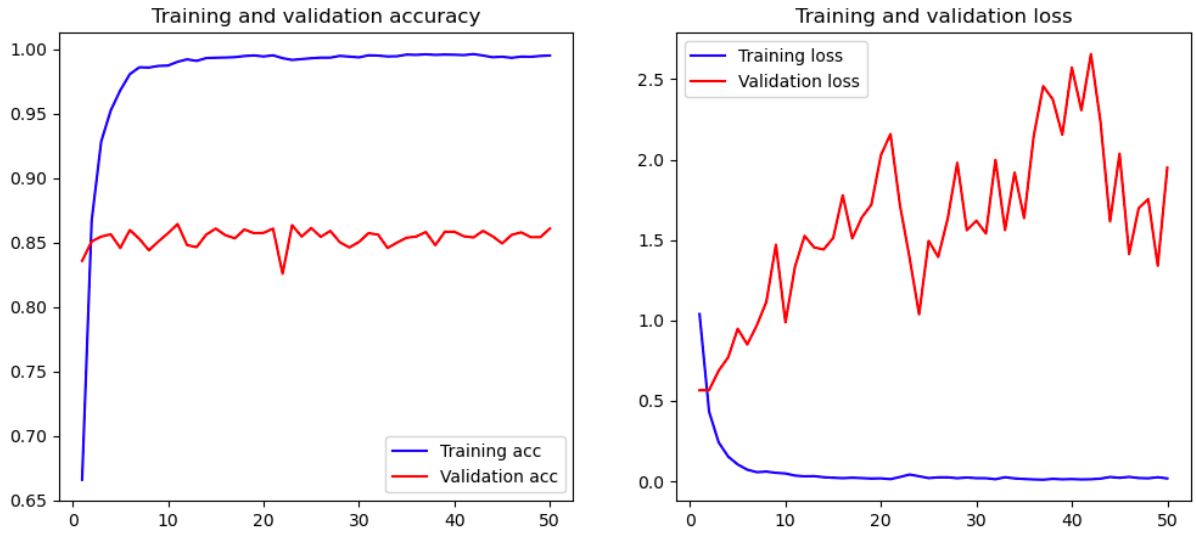


Hình 12: Kiến trúc mô hình MLP

Lớp đầu tiên của mạng MLP sẽ nhận đầu vào là các bài báo trong tập huấn luyện đã được tách từ đa âm tiết và được vector hóa với kỹ thuật TF-IDF dưới dạng một ma trận thưa với 95289 đặc trưng. Các lớp tiếp theo sẽ là lớp kết nối đầy đủ (Dense Layer) với số lượng Nơron

trong mỗi lớp lần lượt là 2048, 2048, 1024, 512, 256, 128, 64, 32 và lớp cuối cùng chứa 10 Nơron tương ứng với số chủ đề cần phân loại. Tổng tham số có thể huấn luyện được của mô hình là 153,458,986.

- **Kết quả huấn luyện**

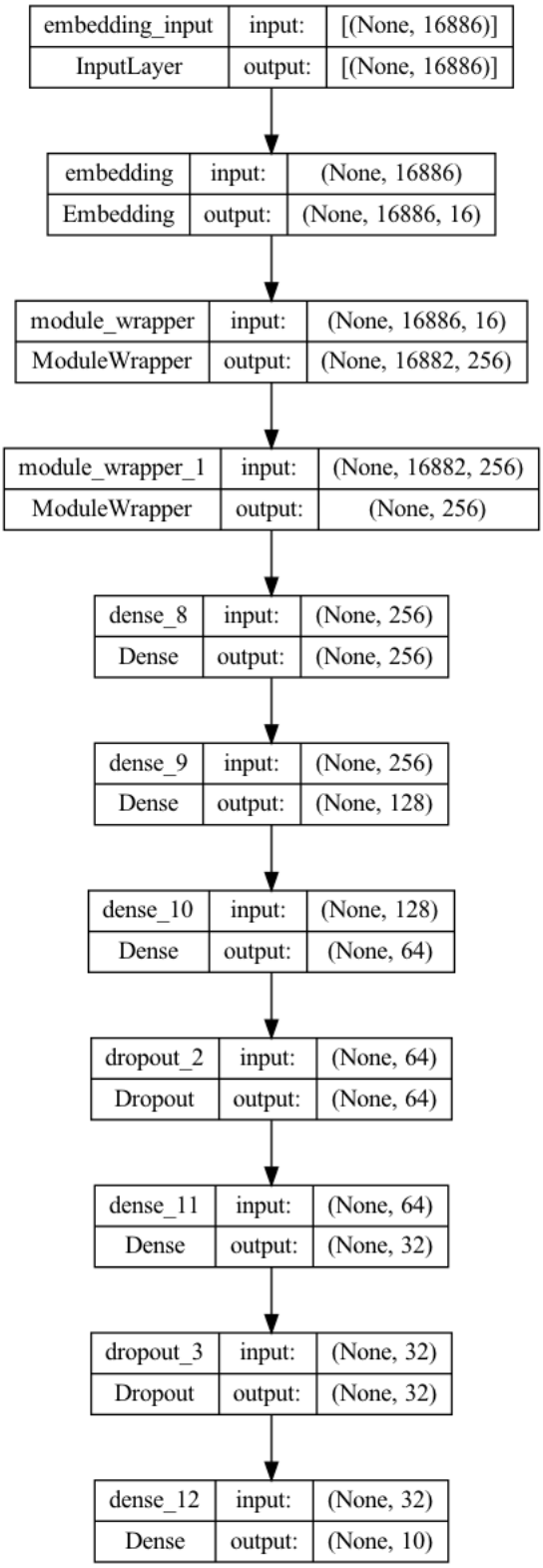


Hình 13: Kết quả huấn luyện mô hình MLP

- Sau khi huấn luyện với 50 Epoch, mô hình đạt được độ chính xác cao nhất trên tập dữ liệu thẩm định là 86,428%. Độ chính xác trên tập dữ liệu kiểm thử là 86,01%.

3.3.4.2. Mạng Nơron tích chập

- **Kiến trúc mô hình**



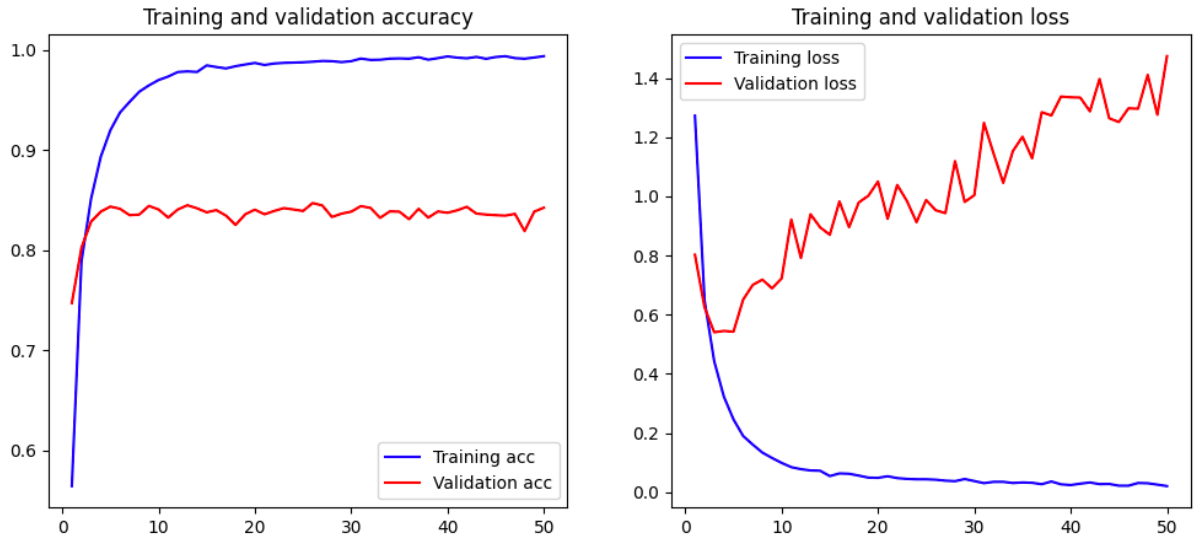
Hình 14: Kiến trúc mô hình CNN

Lớp đầu tiên của mạng CNN là lớp nhúng (Embedding) với nhiệm vụ biểu diễn các từ trong tập dữ liệu huấn luyện dưới dạng một vector số thực n chiều. Lớp này yêu cầu dữ liệu đầu vào cần được mã hóa bởi một số nguyên, như vậy mỗi từ trong bộ ngữ liệu huấn luyện sẽ được đại diện bởi một số nguyên duy nhất. Lớp Nhúng sau đó sẽ được khởi tạo với các trọng

số ngẫu nhiên và sẽ học cách nhúng cho tất cả các từ trong tập dữ liệu huấn luyện. Sau đó lớp nhúng sẽ được kết nối với lớp tích chập để thực hiện phép tích chập và tiếp tục được giảm chiều dữ liệu.

Các lớp tiếp theo sẽ là lớp kết nối đầy đủ (Dense Layer) với số lượng Nơron trong mỗi lớp lần lượt là 256, 128, 64, 32 và lớp cuối cùng chứa 10 Nơron tương ứng với số chủ đề cần phân loại. Tổng tham số có thể huấn luyện được của mô hình là 1,346,010.

- **Kết quả huấn luyện**

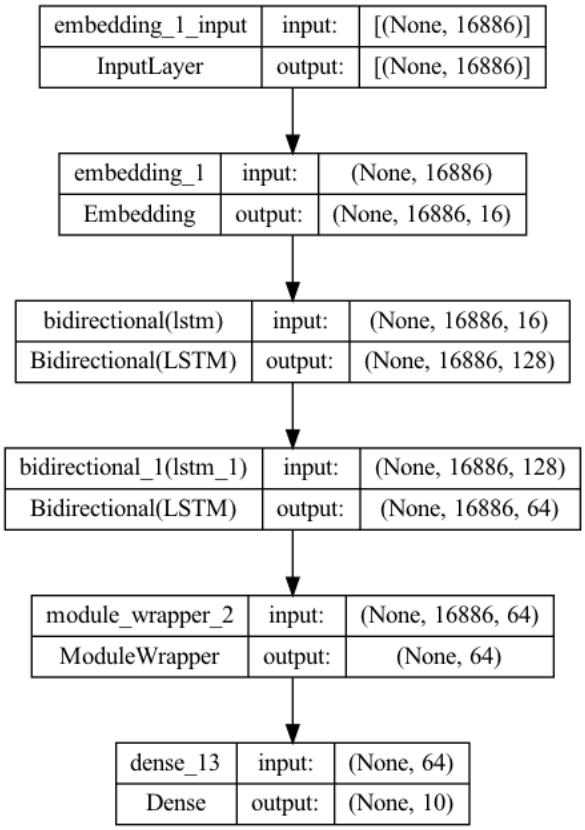


Hình 15: Kết quả huấn luyện mô hình CNN

- Sau khi huấn luyện với 50 Epoch, mô hình đạt được độ chính xác cao nhất trên tập dữ liệu thẩm định là 84,726%. Độ chính xác trên tập dữ liệu kiểm thử là 83,96%.

3.3.4.3. Mạng trí nhớ ngắn hạn, dài hạn

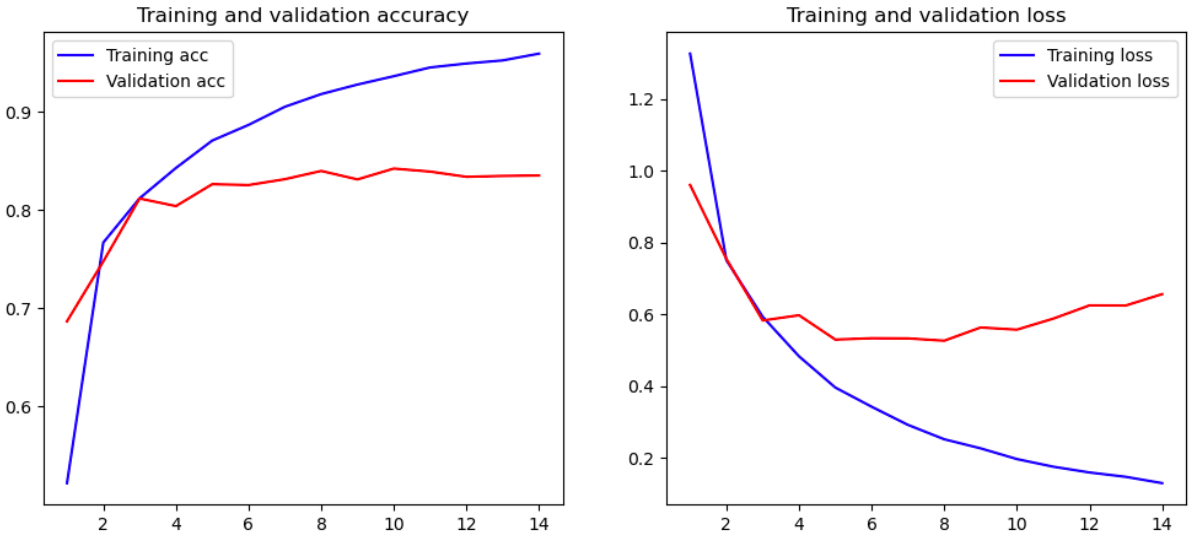
- **Kiến trúc mô hình**



Hình 16: Kiến trúc mô hình LSTM

Lớp đầu tiên của mạng LSTM là lớp nhúng tương tự như mạng CNN. Sau đó lớp nhúng sẽ được kết nối với cặp LSTM 2 chiều (BiLSTM) với mỗi chiều là một lớp: một lớp sẽ lấy đầu vào theo chiều thuận và lớp kia theo hướng ngược lại. BiLSTM tăng hiệu quả lượng thông tin có sẵn cho mạng, cải thiện bối cảnh có sẵn cho thuật toán. Kết quả đầu ra của 2 lớp BiLSTM sau đó sẽ được tổng hợp thành vector 1 chiều bởi lớp gộp và được kết nối với lớp cuối cùng để tiến hành phân loại.

- **Kết quả huấn luyện**



Hình 17: Kết quả huấn luyện mô hình LSTM

- Sau khi huấn luyện với 14 Epoch, mô hình đạt được độ chính xác cao nhất trên tập dữ liệu thẩm định là 84,24%. Độ chính xác trên tập dữ liệu kiểm thử là 84,92%.

Chương 4. Đánh giá kết quả thực nghiệm

4.1. Phân loại dữ liệu mới

4.1.1. Các mô hình học máy

Thể loại bài báo	Đường dẫn	Kết quả mô hình Hồi quy Logistic	Kết quả mô hình SVM
Kinh doanh	https://vnexpress.net/gia-vang-tuan-toi-co-the-tang-4629911.html	Kinh doanh	Kinh doanh
Thể thao	https://vnexpress.net/alcaraz-toi-tung-nghi-khong-the-danh-bai-djokovic-4630198.html	Thể thao	Thể thao
Đời sống	https://thanhnien.vn/nhung-nguoi-hy-sinh-giac-ngu-chi-doi-lai-nu-cuoi-185230716141530768.htm	Đời sống	Đời sống
Công nghệ	https://thanhnien.vn/sony-quay-lai-thi-truong-viet-nam-bang-hai-mau-smartphone-xperia-moi-185230717101107123.htm	Công nghệ	Công nghệ
Văn hóa	https://thanhnien.vn/cai-luong-tap-the-mot-thoi-vang-bong-so-phan-cua-doan-sai-gon-1-2-3-185230716233249471.htm	**Giải trí**	Giải trí
Âm nhạc	https://vtv.vn/van-hoa-giai-tri/tai-phat-hanh-speak-now-taylor-swift-lap-hang-loat-ki-luc-moi-tren-billboard-20230717100010913.htm	Âm nhạc	**Văn hóa**
Du lịch	https://vnexpress.net/so-sanh-du-	**Kinh doanh**	Du lịch

	lich-viet-voi-cac-nuoc-dong-nam-a-4628811.html		
Giáo dục	https://vtv.vn/giao-duc/nhieu-truong-dai-hoc-cong-bo-diem-san-xet-tuyen-bang-diem-thi-tot-nghiep-trung-hoc-pho-thong-20230720161701174.htm	Giáo dục	Giáo dục
Pháp luật	https://vnexpress.net/cuu-pho-chu-tich-ha-noi-toi-tro-thanh-toi-do-cua-thanh-pho-4630796.html	Pháp luật	Pháp luật
Giải trí	https://tuoitre.vn/dem-nhac-dac-biet-chao-mung-325-nam-hinh-thanh-sai-gon-cho-lon-gia-dinh-tp-hcm-20230702205148203.htm	**Văn hóa**	**Văn hóa**

Bảng 8: Kết quả phân loại dữ liệu mới của nhóm mô hình học máy

4.1.2. Các mô hình học sâu

Thể loại bài báo	Đường dẫn	Kết quả mô hình MLP	Kết quả mô hình CNN	Kết quả mô hình BiLSTM
Kinh doanh	https://vnexpress.net/gia-vang-tuan-toi-co-the-tang-4629911.html	Kinh doanh	Kinh doanh	Kinh doanh
Thể thao	https://vnexpress.net/alcaraz-toi-tung-nghi-khong-the-danh-bai-djokovic-4630198.html	Thể thao	Thể thao	Thể thao
Đời sống	https://thanhnien.vn/nhung-nguoi-hy-sinh-giac-ngu-chi-doi-lai-nu-cuoi-185230716141530768.htm	Đời sống	Đời sống	Đời sống
Công nghệ	https://thanhnien.vn/sony-quay-lai-thi-truong-viet-nam-bang-hai-mau-smartphone-xperia-moi-185230717101107123.htm	Công nghệ	Công nghệ	Công nghệ
Văn hóa	https://thanhnien.vn/cai-luong-tap-the-mot-thoi-vang-bong-so-phan-cua-doan-sai-gon-1-2-3-185230716233249471.htm	**Giải trí**	**Giải trí**	Văn hóa
Âm nhạc	https://vtv.vn/van-hoa-giai-tri/tai-phat-hanh-speak-now-taylor-swift-lap-hang-loat-ki-luc-moi-tren-billboard-20230717100010913.htm	Âm nhạc	**Giải trí**	**Giải trí**
Du lịch	https://vnexpress.net/so-sanh-du-lich-viet-voi-cac-nuoc-dong-nam-a-4628811.html	Du lịch	Du lịch	Du lịch
Giáo dục	https://vtv.vn/giao-duc/nhieu-truong-dai-hoc-cong-bo-diem-san-xet-tuyen-bang-diem-thi-tot-nghiep-trung-hoc-pho-	Giáo dục	Giáo dục	Giáo dục

	thong-20230720161701174.htm			
Pháp luật	https://vnexpress.net/cuu-pho-chu-tich-ha-noi-toi-tro-thanh-toi-do-cua-thanh-pho-4630796.html	Pháp luật	Pháp luật	Pháp luật
Giải trí	https://tuoitre.vn/dem-nhac-dac-biet-chao-mung-325-nam-hinh-thanh-sai-gon-cho-lon-gia-dinh-tp-hcm-20230702205148203.htm	Giải trí	Giải trí	Giải trí

Bảng 9: Kết quả phân loại dữ liệu mới của nhóm mô hình học sâu

4.2. Đánh giá kết quả

Có thể thấy được đối với các thể loại đặc trưng như kinh doanh, thể thao, công nghệ, du lịch, giáo dục, đời sống và pháp luật thì các mô hình vẫn có thể thực hiện phân loại một cách chính xác. Tuy nhiên đối với các thể loại ít có sự phân tách tường minh hơn như âm nhạc, giải trí, văn hóa thì các mô hình có gặp khó khăn trong quá trình phân loại. Đồng thời đối với những thể loại này thì chúng ta cũng khó kết luận bài báo thuộc hoàn toàn vào thể loại nào, vì các bài báo này đều chứa khả năng đồng thời thuộc vào các thể loại khác nhau.

Về mặt thực nghiệm, mô hình học máy Hồi quy Logistic và Máy Vector Hỗ trợ với kỹ thuật tách từ LM và vector hóa TF-IDF đã đạt được kết quả tốt nhất trên tập dữ liệu kiểm thử với độ chính xác 89%. Các mô hình học sâu tuy đạt được điểm số ấn tượng trên tập dữ liệu huấn luyện (gần như tuyệt đối), thì lại gặp hiện tượng quá khớp khi cố gắng đưa ra kết quả phân loại mới trên tập dữ liệu kiểm thử. Mặc dù vậy, các mô hình mạng CNN và LSTM với kỹ thuật nhúng từ lại yêu cầu ít đơn vị tính toán hơn các phương pháp truyền thống như Túi Từ và TFIDF, do đó sẽ phù hợp hơn khi cần phải phân tích và xử lý với một lượng lớn dữ liệu. Tăng lượng dữ liệu huấn luyện cũng là một cách thức để nâng cao hiệu năng học của các mô hình học sâu.

KẾT LUẬN VÀ ĐỀ XUẤT

1. Kết quả đạt được

Về mặt khái niệm và lý thuyết, luận văn đã tiến hành khảo sát và hệ thống hóa các định nghĩa, các hướng tiếp cận liên quan đến quy trình xây dựng hệ thống phân loại chủ đề tin tức tiếng việt tự động, bao gồm: thu thập dữ liệu, tiền xử lý dữ liệu tin tức, trích xuất đặc trưng, giảm chiều dữ liệu, mô hình hóa dữ liệu và đánh giá hiệu năng

mô hình. Từ đó có thể tìm ra các cách thức, kỹ thuật phù hợp để tiến hành thực nghiệm trên dữ liệu thực.

Về mặt thực nghiệm, các bài báo điện tử thuộc 10 chủ đề khác nhau đã được thu thập từ các trang báo điện tử phổ biến. Các bài báo này sau đó đã được áp dụng các phương pháp tiền xử lý là làm sạch văn bản và loại bỏ từ dừng. Dữ liệu sau khi được tiền xử lý đã được sử dụng để xây dựng bộ từ điển theo 2 phương pháp tách từ khác nhau và được vector hóa theo 2 cách thức khác nhau và thực nghiệm trên các mô hình nhằm tìm ra hướng tiếp cận tối ưu nhất. Đồng thời, các hướng tiếp cận phổ biến trong thời gian gần đây như kỹ thuật nhúng từ và các mô hình học sâu cũng được áp dụng và so sánh kết quả, đưa ra đánh giá.

2. Hạn chế và hướng phát triển đề tài

Bên cạnh những kết quả đạt được, luận văn vẫn còn những hạn chế sau:

+ ***Hạn chế trong các phương pháp giảm chiều dữ liệu***: Các kỹ thuật vector hóa truyền thống thường tạo ra bộ dữ liệu với số chiều hay số lượng đặc trưng vô cùng lớn, vậy nên việc áp dụng các kỹ thuật giảm chiều dữ liệu là cần thiết để có thể tiết kiệm tài nguyên tính toán mà vẫn đạt được hiệu năng cần thiết.
+ ***Vấn đề quá khớp đối với các mô hình học sâu:*** Mặc dù được huấn luyện trên một lượng lớn dữ liệu, các mô hình học sâu vẫn không tránh khỏi tình trạng quá khớp.

Hướng phát triển của đề tài:

+ Tiến hành thu thập nhiều dữ liệu hơn, từ đó có thể tạo ra được kết quả kiểm thử tốt hơn.
+ Áp dụng phương pháp phân loại đa nhãn thay vì phân loại đơn lớp vì một bài báo có thể thuộc vào nhiều chủ đề khác nhau.
+ Thực nghiệm với các kỹ thuật trích xuất đặc trưng và giảm chiều dữ liệu khác nhằm tìm ra hướng tiếp cận phù hợp hơn cho mục đích phân tích bộ dữ liệu.

TÀI LIỆU THAM KHẢO

Gerard, S., & Christopher, B. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management, 24*(5), 513-523.

Yoav, G., & Omer, L. (2014). word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global Vectors for Word Representation. *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP*, *14.*

Cam, T. N., Trung, K. N., Xuan, H. P., Le, M. N., & Quang, T. H. (2006). Vietnamese word segmentation with CRFs and SVMs: An investigation. *20th Pacific Asia Conference on Language, information and Computation (PACLIC).*

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (n.d.). *Efficient estimation of word representations in vector space.* arXiv.

Thorsten, J. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. *European Conference on Machine Learning*, (pp. 137–142).

Dien, D., Kiem, H., & Van, T. N. (2001). Vietnamese Word Segmentation. *6th Natural Language Processing Pacific Rim Symposium (NLPRS)*, (pp. 749–756).

Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features. *ECML Conference.*

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space.* arXiv .

Shaojie, B., J. Zico, K., & Vladlen, K. (2018). *An empirical evaluation of generic convolutional and recurrent networks for sequence modeling.* ArXiv.

Nal, K., Edward, G., & Phil, B. (n.d.). A convolutional neural network for modelling sentences. *52nd Annual Meeting of the Association for Computational Linguistics* (pp. 655-665). ACL 2014.

Peng, W., Jiaming, X., Bo, X., Cheng-Lin, L., Heng, Z., Fangyuan, W., & Hongwei, H. (2015). Semantic clustering and convolutional neural network for short text categorization. *53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing* (pp. 352–357). ACL 2015.

Ronan, C., Jason, W., Leon, B., Michael, ´. K., Koray, K., & Pavel, P. K. (n.d.). Natural language processing (almost) from scratch. *Machine Learning Research*.

Yoav, G. (2016). A Primer on Neural Network Models for Natural Language Processing. *Journal of Artificial Intelligence Research 57*, 345–420.

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 157–166.

Hochreiter, S., Bengio, Y., Frasconi, P., Schmidhuber, J., & others. (n.d.). *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.*

PHỤ LỤC

- **Bản kiểm tra đạo văn**

https://drive.google.com/file/d/1cLj3aLZ8dL5hFKcXNi0lCuy-TuijGkF2/view?usp=share_link

- **Mã nguồn**

Mã nguồn của luận văn cùng những dữ liệu liên quan đã được tải lên địa chỉ sau: https://drive.google.com/drive/folders/1VQuLAtoH6E7LQL6HUIRaCv2Ub-9AreoR?usp=sharing