TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN CUỐI KÌ MÔN

KHAI THÁC CÁC TẬP DỮ LIỆU LỚN

PHÂN LOẠI VĂN BẢN TIẾNG VIỆT SỬ DỤNG PHƯƠNG PHÁP HỌC MÁY SVM

Người hướng dẫn: **TS BÙI THANH HÙNG**

Người thực hiện: **HỒNG QUANG VINH – 186005004**

NGUYỄN ĐẠI THỊNH – 186005035

Lớp**: 18600531**

Khoá**: 2018-2020**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2019

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN CUỐI KÌ MÔN

KHAI THÁC CÁC TẬP DỮ LIỆU LỚN

PHÂN LOẠI VĂN BẢN TIẾNG VIỆT SỬ DỤNG PHƯƠNG PHÁP HỌC MÁY SVM

Người hướng dẫn: **TS BÙI THANH HÙNG**

Người thực hiện: **HỒNG QUANG VINH – 186005004**

NGUYỄN ĐẠI THỊNH – 186005035

Lớp**: 18600531**

Khoá**: 2018-2020**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2019

LỜI CẢM ƠN

Nhóm chúng em xin chân thành cảm ơn Thầy Bùi Thanh Hùng đã giúp đỡ chúng em hoàn thành đồ án. Những hướng dẫn của Thầy giúp chúng em có một nền tảng lý thuyết đủ để có thể ứng dụng và nghiên cứu phát triển đề tài này. Xin chân thành cảm ơn Thầy.

ĐỒ ÁN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của TS Bùi Thanh Hùng;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày tháng năm

Tác giả

(ký tên và ghi rõ họ tên)

Hồng Quang Vinh

Nguyễn Đại Thịnh

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

_____

_____

_____

_____

_____

_____

_____

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

Phần đánh giá của GV chấm bài

_____

_____

_____

_____

_____

_____

_____

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

TÓM TẮT

Ngày nay, với sự phát triển của cuộc Cách mạng Công nghiệp 4.0, tiên phong là sự phát triển của lĩnh vực Trí tuệ nhân tạo, con người đang tiến gần hơn tới mục đích làm cho máy tính mô phỏng được các hành vi của con người. Có thể kể đến một số lĩnh vực như nhận dạng hình ảnh, tiếng nói… Trong đó, Xử lý ngôn ngữ tự nhiên là một nhanh đang được quan tâm và có tốc độ phát triển nhanh nhất.

Xử lý ngôn ngữ tự nhiên là quá trình mô phỏng và huấn luyện cho máy tính cách mà con người tiếp nhận, xử lý ngôn ngữ. Quá trình mô phỏng này giống như cách mà con người học ngôn ngữ, xử lý văn bản chính là xử lý dựa trên mức độ cơ sở của văn bản đó, nói các khác, chúng ta cần xử lý trên mức độ từ và câu của văn bản đó. Chính vì vậy, để có thể giải quyết bài toán này, người thực hiện cũng cần có một lượng kiến thức cơ sở về ngôn ngữ học. Một số thành tựu trong lĩnh vực này có thể kể đến như: Chatbot, dịch tự động, so sánh sự tương đồng giữa hai văn bản…

Tuy nhiên, đi cùng với đó là không ít khó khăn, thách thức như: Sự khác nhau về ngôn ngữ của mỗi quốc gia, cơ sở dữ liệu của các ngôn ngữ còn thiếu… Dẫn đến hiệu quả của các mô hình xử lý ngôn ngữ không được cao.

Trong đồ án này, nhóm hướng đến các ứng dụng của xử lý ngôn ngữ tự nhiên, mục tiêu có thể phân loại chủ đề các văn bản đầu vào. Các khái niệm cũng như quá trình thực hiện sẽ được trình bày ở phần sau của đồ án này.

MỤC LỤC

LỜI CẢM ƠN i
PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iii
TÓM TẮT iv
MỤC LỤC 1
DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 3
CHƯƠNG 1 – PHÁT BIỂU BÀI TOÁN 4
1.1 Đặt vấn đề 4
1.2 Cách tiếp cận 5
1.3 Ý nghĩa khoa học và thực tiễn của bài toán 5
CHƯƠNG II – GIẢI QUYẾT BÀI TOÁN 6
2.1 Tổng quan về phương pháp giải quyết bài toán 6
2.2 Đặc trưng của giải pháp đề xuất 7
2.2.1 Word Embedding 7
2.2.2 One-hot vector 8
2.2.3 Bag of Words (BoW) 8
2.2.4 Word2vec 9
2.2.5 Support Vector Machine (SVM) 11
2.2.5.1 Giới thiệu 11
2.2.5.2 Các khái niệm cơ bản 11
2.3 Cách đánh giá 19
CHƯƠNG III – THỰC NGHIỆM 20
3.1 Dữ liệu 20
3.2 Công nghệ 22
3.2.1 Ngôn ngữ lập trình 22
3.2.2 Thư viện 22
3.3 Thực nghiệm 22

3.4 Đánh giá kết quả 24
CHƯƠNG IV – KẾT LUẬN 26
4.1 Kết quả đạt được 26
4.1.1 Kết quả 26
4.1.2 Hạn chế 26
4.2 Hướng phát triển 26
TÀI LIỆU THAM KHẢO 27

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

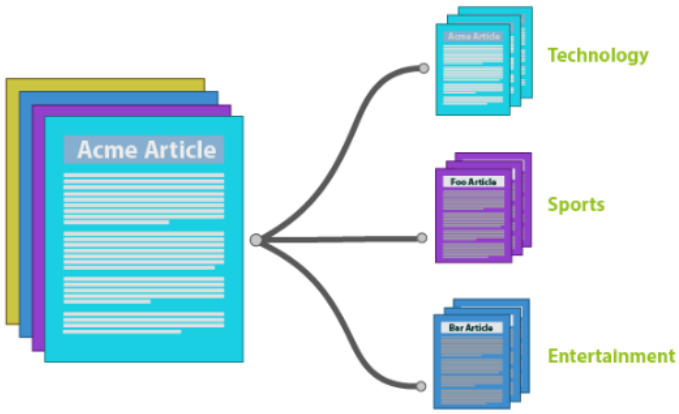
DANH MỤC HÌNH

Hình 1: Mô tả phân loại văn bản....4
Hình 2: Mô tả quy trình dự đoán nhãn dữ liệu 6
Hình 3: Vector từ trong word2vec 9
Hình 4: Mô hình CBOW và Skip-gram 10
Hình 5: Đường phân chia đối với tập dữ liệu gồm 2 thuộc tính 12
Hình 6: Một bộ dữ liệu hai chiều được phân chia tuyến tính 13
Hình 7: Siêu phẳng phân chia tuyến tính cùng với biên độ của nó 14
Hình 8: Đường biểu diễn H1 và H2 15
Hình 9: Các support vector trong SVM 16
Hình 10: Một trường hợp đơn giản trên không gian 2 chiều 18
Hình 11: Bộ dữ liệu huấn luyện trong 10-topics....21
Hình 12: Bộ dữ liệu kiểm tra trong 10-topics 21
Hình 13: Kết quả quá trình huấn luyện....24
Hình 14: Kết quả test 1 đoạn văn bản bất kỳ 25

CHƯƠNG 1 – PHÁT BIỂU BÀI TOÁN

1.1 Đặt vấn đề

Phân loại văn bản hay còn gọi là Text Classifcation hoặc là Text Categorizer là một bài toán thuộc về lĩnh vực Xử lý ngôn ngữ tự nhiên dưới dạng văn bản. Một ví dụ minh họa cho việc phân loại văn bản này là việc sắp xếp các tin tức trên báo vào các danh mục tương ứng như thể thao, giải trí, xã hội...Việc này có thể được thực hiện thủ công bởi con người tuy nhiên nó rất mất thời gian và công sức. Thay vào đó, chúng ta sẽ sử dụng một số kĩ thuật học máy để tiến hành phân loại tự động các tin tức đó.



Hình 1: Mô tả phân loại văn bản

Bài toán có thể được mô tả ngắn gọn dưới dạng toán học như sau: Cho một tập gồm *n* văn bản - document đầu vào bằng các kĩ thuật xử lý nào đó chúng ta sẽ phân chúng vào một tập gồm *m* phân lớp – categories, trong đó:

- D = $\{d_1,d_2,\ldots,d_n\}$: Tập hợp các văn bản đầu vào.
- C = $\{c_1,c_2,\ldots,c_m\}$: Tập hợp các phân lớp.

Các bài toán phân loại là một dạng bài toán con của Supervised learning (Học có giám sát), trong đó ta cần có các cặp dữ liệu và nhãn biết trước, sau quá trình học, máy có thể phân loại các dữ liệu mới đưa vào thuộc nhãn nào.

1.2 Cách tiếp cận

Để giải quyết bài toán trên, ta cần phải đưa ra lời giải cho 2 câu hỏi:

- Phân loại gì?
- Phân loại như thế nào?

Mục tiêu của đề tài này là phân loại văn bản tiếng Việt theo chủ đề. Nói cách khác, để trả lời câu hỏi đầu tiên, ta cần phải thu thập được bộ dữ liệu phù hợp (đã gắn nhãn, dữ liệu đủ lớn). Chính vì sự phong phú và đa dạng của tiếng Việt, mà việc thu thập bộ dữ liệu là rất khó khăn.

Vì vậy đề tài sẽ sử dụng bộ dữ liệu có sẵn, bao gồm 10 chủ đề: Chính trị, Đời sống, Kinh doanh, Khoa học, Pháp luật, Sức khỏe, Thế giới, Thể thao, văn hóa, Vi tính. Chi tiết về bộ dữ liệu sẽ được đề cập ở phần sau bài viết.

Tìm lời giải cho câu hỏi thứ hai cũng chính là tìm quy trình hiện thực cho bài toán. Nói cách khác, ta cần xác định các phương pháp có thể áp dụng để cho ra kết quả tốt. Phương pháp học máy SVM (Support Vector Machine) thường được đề xuất là phương pháp cho ra kết quả cực kì tốt trong các bài toán phân loại. Ngoài ra, quy trình hiện thực cũng được phân ra làm hai giai đoạn nhỏ hơn là Learning (Huấn luyện) và Prediction/Validation (Kết quả dự đoán/Đánh giá kết quả).

Chi tiết về các phương pháp sẽ được trình bày ở phần sau bài viết.

1.3 Ý nghĩa khoa học và thực tiễn của bài toán

Về mặt ý nghĩa khoa học, phân loại văn bản là dạng bài toán trong lĩnh vực xử lý ngôn ngữ tự nhiên. Việc hiện thực các phương pháp giải quyết cho bài toán này sẽ giúp tiến một bước đến việc làm cho máy tính có khả năng tương tự con người, cụ thể là xử lý các thông tin dạng văn bản. Ngoài ra, việc áp dụng và đánh giá các mô hình sẽ đóng

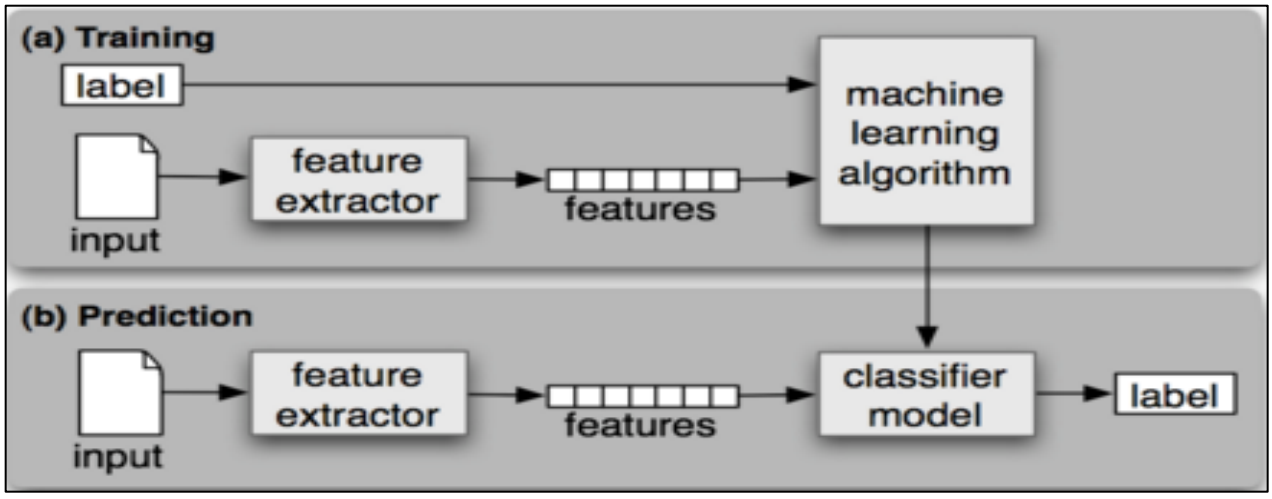
góp cho cộng đồng các kết quả thực nghiệm, từ đó có thể nghiên cứu đề xuất các mô hình, hoặc kết hợp các mô hình sẵn có để có thể giải quyết bài toán một cách tối ưu nhất.

Về mặt thực tiễn, việc áp dụng Machine Learning (ML) vào các công việc hàng ngày của con người là rất cần thiết, giúp ta tận dụng hết khả năng của máy tính, vừa giảm tải công việc của con người. Có rất nhiều lĩnh vực thực tiễn có thể áp dụng ML, như ví dụ minh họa ở trên, đó là phân loại các bài báo và sắp xếp đúng chủ đề của nó.

CHƯƠNG II – GIẢI QUYẾT BÀI TOÁN

2.1 Tổng quan về phương pháp giải quyết bài toán

Như đã đề cập ở trên, quy trình hiện thực bài toán được chia làm hai công đoạn chính: Learning (Huấn luyện) và Prediction (Dự đoán). Quy trình có thể được mô tả tóm gọn như hình dưới đây.



Hình 2: Mô tả quy trình dự đoán nhãn dữ liệu

Quá trình huấn luyện (training) (a): Các Input đầu vào ở đây là các văn bản. Tuy nhiên máy tính không thể hiểu được nguyên gốc dữ liệu dạng văn bản, cần phải áp dụng

các phương pháp rút trích đặc trưng (feature extractor). Đầu ra của các phương pháp này là các vector số - dạng dữ liệu mà máy tính hay các giải thuật học máy có thể xử lý. Các đặc trưng này sau đó được đưa vào các phương pháp học, kết hợp với các nhãn được gán trước để sinh ra model.

Quá trình dự đoán (prediction) (b): Các văn bản đầu vào cũng được rút trích đặc trưng thành các feature vector, sau đó đưa vào cho model dự đoán các nhãn của dữ liệu đầu vào.

Các phương pháp được sử dụng được tóm tắt lại:

- Tiền xử lý dữ liệu: Bao gồm tách từ và các ký hiệu đặc biệt (tokenization, segmentation), loại bỏ từ dừng (stopwords). Mục tiêu của phương pháp là giảm kích thước dữ liệu đầu vào, giúp giảm độ phức tạp cũng như thời gian hiện thực.

- Rút trích đặc trưng (feature extraction) Dữ liệu đầu vào sẽ được chuyển hóa thành các vector dạng số. Quá trình này gọi chung là Word Embedding, và phương pháp rút đặc trưng được áp dụng là Bag of Words (BoW).

- Huấn luyện/Dự đoán: Phương pháp sử dụng là SVM (Support Vector Machine) nhằm xây dựng mô hình dự đoán dữ liệu dựa trên các cặp dữ liệu/nhãn được huấn luyện trước đó.

2.2 Đặc trưng của giải pháp đề xuất

2.2.1 Word Embedding

Word Embedding là tên gọi chung của các kỹ thuật xử lý từ và văn bản trong Xử lý ngôn ngữ tự nhiên (NLP), trong đó các từ, cụm từ hoặc văn bản được ánh xạ tới các vector số thực. Nó giúp xác định ngữ cảnh của một từ trong văn bản, sự tương đồng về ngữ nghĩa và cú pháp, quan hệ của một từ với các từ khác v.v…

Trong ngôn ngữ học, word embedding được thảo luận trong lĩnh vực nghiên cứu về phân phối ngữ nghĩa. Nó nhằm mục đích định lượng và phân loại sự tương đồng về

ngữ nghĩa giữa các mục ngôn ngữ dựa trên các thuộc tính phân phối của chúng trong các mẫu dữ liệu ngôn ngữ lớn.

2.2.2 One-hot vector

Phương pháp cơ bản nhất trong Word Embedding là one-hot vector, nó là vector chỉ chứa các giá trị dạng nhị phân. Phương pháp one-hot vector sẽ mã hoá (encoding) các từ trong từ điểnt hành một vector có chiều dài N (tổng số lượng các từ trong từ điển)

- $V = \{tôi_1, đang_2, đi_3, tìm_4, \ nửa_5, của_6, mình_7, đã_8, ăn_9, quả_{10}, táo_{11}\}$
- $S = 11$

Ví dụ về chuyển từ sang one-hot vector

- $tôi = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$

Ví dụ về chuyển văn bản sang one-hot vector

- tôi đang ăn quả táo $= [1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1]$

2.2.3 Bag of Words (BoW)

Mô hình túi từ (Bag of Words) là một dạng one-hot vector và bổ sung thêm đặc trưng là số lần xuất hiện của từ trong văn bản. Một ví dụ minh họa về BoW như sau:

- Vd:
  - (1) Phúc thích xem phim. Đạt cũng thích xem phim.
  - (2) Bích cũng thích xem các trận bóng đá.
  - Từ điển: [“Phúc”, “Đạt”, “Bích”, “thích”, “xem”, “phim”, “cũng”, “các”, “trận”, “bóng”, “đá”]
  - Vector đặc trưng cho 2 văn bản:
    - (1) [1 , 1 , 0 , 2 , 2 , 2 , 0 , 0 , 0 , 0 , 0]
    - (2) [0 , 0 , 1 , 1 , 1 , 0 , 1 , 1 , 1 , 1 , 1]

Một số hạn chế của phương pháp Bag-of-Words

- Độ dài của vector quá lớn, bằng với độ dài của từ điển
- Không xác định được tương quan ý nghĩa giữa các từ

Chính vì những hạn chế đó, người ta đề xuất ra mô hình word2vec, được xây dựng dựa trên cơ sở của one-hot vector.

2.2.4 Word2vec

Word2vec là một mạng neural với duy nhất 1 tầng ẩn (hidden layer), lấy đầu vào là một khối văn bản (corpus) lớn, đầu ra là một vector khoảng vài trăm chiều với mỗi từ duy nhất trong corpus được gắn với một vector tương ứng trong không gian. Các word vectors được xác định trong không gian vector sao cho những từ có chung ngữ cảnh được đặt gần nhau trong không gian.

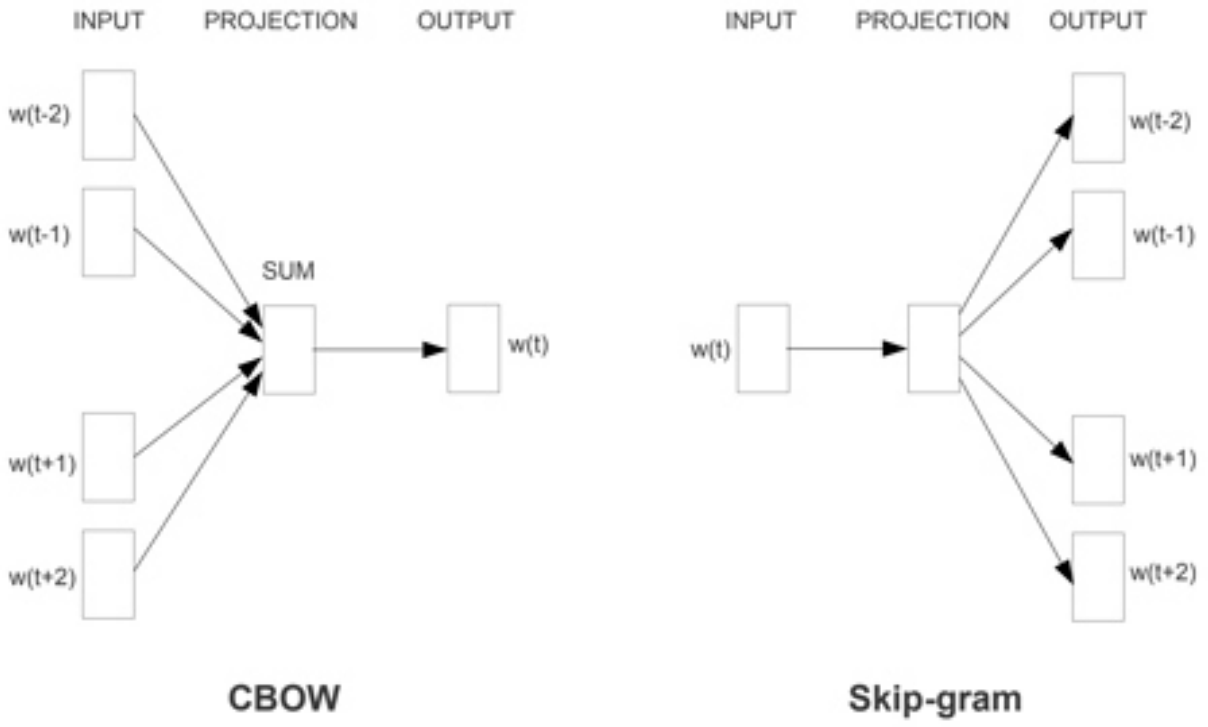
		Vua	Hoàng hậu	Phụ nữ	Công chúa
Hoàng gia		0.99	0.99	0.02	0.98
Nam tính		0.99	0.05	0.01	0.02
Nữ tính		0.05	0.93	0.999	0.94
Tuổi		0.7	0.6	0.5	0.1

Hình 3: Vector từ trong word2vec

Word2vec có thể sử dụng một trong hai kiến trúc mô hình để tạo ra một thể hiện phân tán của các từ: continous bag-of-words (cbow) hoặc skip-gram.

Trong kiến trúc continous bag-of-words, mô hình sử dụng một cửa sổ của các từ ngữ cảnh xung quanh để dự đoán từ hiện tại. Thứ tự của các từ ngữ cảnh không ảnh hưởng đến dự đoán.

Trong kiến trúc skip-gram liên tục, mô hình sử dụng từ hiện tại để dự đoán cửa sổ xung quanh của các từ ngữ cảnh. Kiến trúc skip-gram cân nhắc các từ ngữ cảnh gần đó nặng hơn các từ ngữ cảnh xa hơn.



Hình 4: Mô hình CBOW và Skip-gram

2.2.5 Support Vector Machine (SVM)

2.2.5.1 Giới thiệu

SVM là một phương pháp trong việc phân loại dữ liệu tuyến tính và không tuyến tính. Bài báo đầu tiên về Support Vector Machine được giới thiệu vào năm 1992 bởi Vladimir Vapnik và hai đồng sự Bernhard Boser và Isabelle Guyon, mặc dù nền móng cơ bản của SVM đã có từ năm 1960 (bao gồm các công việc được thực hiện rất sớm bởi Vapnik và Alexei Chervonenkis trong lý thuyết học thống kê).

Trước khi đi vào tìm hiểu phương pháp SVM, ta phải biết được các khái niệm về siêu phẳng phân chia tuyến tính, support vector…

2.2.5.2 Các khái niệm cơ bản

a. Siêu phẳng phân cách

Cho trước tập dữ liệu *D* gồm $(X_1, y_1), (X_2, y_2), ..., (X_{|D|}, y_{|D|})$.

Trong đó X_i là một tập các bộ huấn luyện tương ứng với nhãn lớp y_i. Mỗi y_i sẽ nhận một trong hai giá trị hoặc là +1 hoặc là -1 $(y_i \in \{+1, -1\})$.

Phương pháp phân lớp SVM sẽ tìm ra đường phân lớp "tốt nhất" để phân chia tập dữ liệu này thành từng *lớp* tách biệt ra với nhau. Theo tài liệu "Robust Real-time Object Detection" của Paul Viola và Michael Jones, phương trình tổng quát của một đường phân chia như vậy được biểu diễn dưới dạng sau:

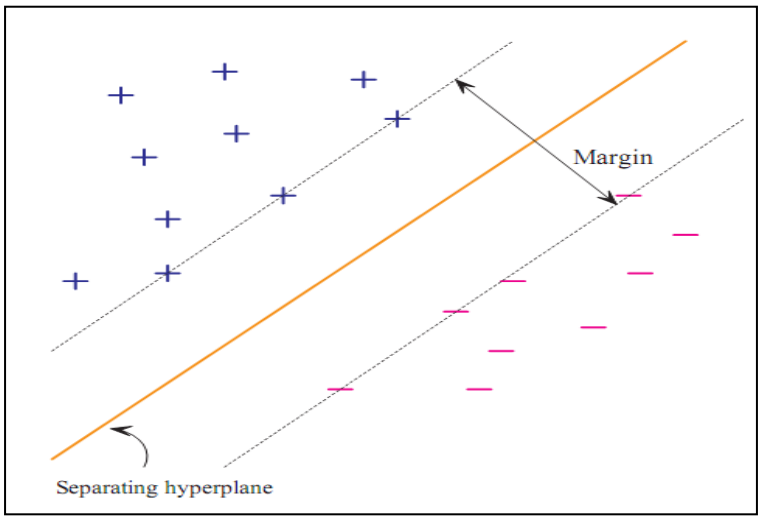
$$W.X + b = 0 \qquad (1)$$

Trong đó:

- *W*: Vector trọng số, $W = \{w_1, w_2, ..., w_n\}$.
- *n*: Số thuộc tính (hay còn gọi là số chiều của dữ liệu).
- *b*: Một đại lượng vô hướng, thường được xem như là một độ nghiêng (bias).

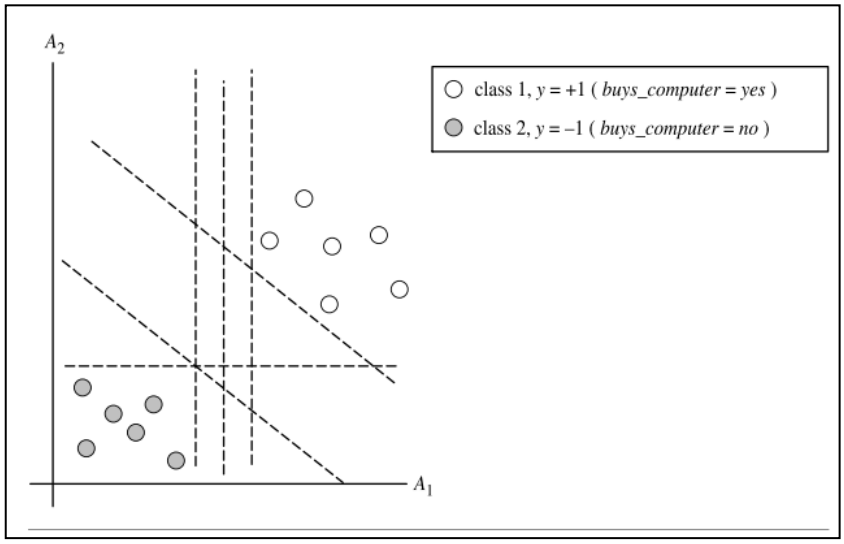
Đối với trường hợp dữ liệu hai chiều (hai thuộc tính) thì phương trình trên biểu diễn của đường thẳng phân chia. Nếu dữ liệu là ba chiều thì đường phân chia giữa hai

tập sẽ là một mặt phẳng phân cách. Tổng quát cho dữ liệu *n* chiều thì sẽ được phân cách bởi một siêu phẳng. Trong bài toán sẽ sử dụng thuật ngữ "siêu phẳng" (hyperplane) để chỉ đến ranh giới quyết định mà muốn tìm kiếm bất chấp số lượng thuộc tính.



Hình 5: Đường phân chia đối với tập dữ liệu gồm 2 thuộc tính

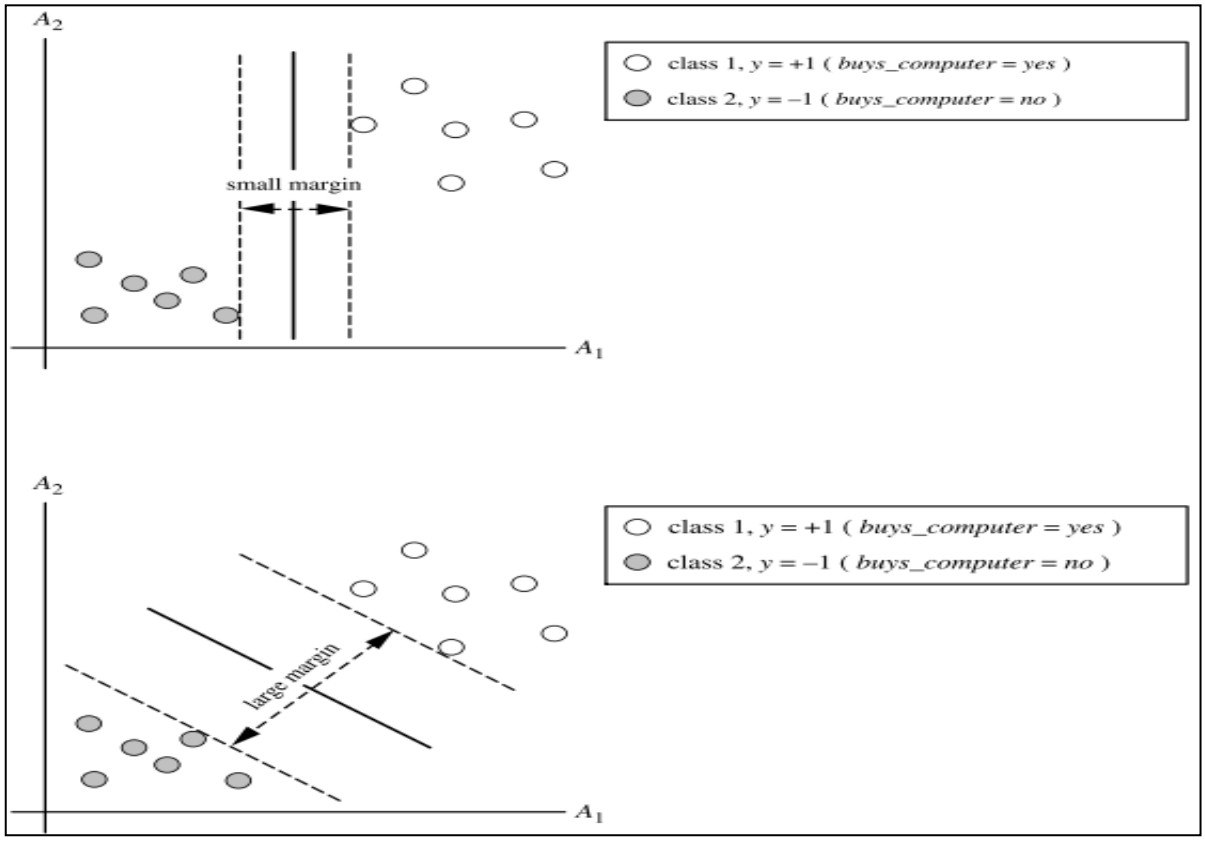
Tuy nhiên trong thực tế ta có thể tìm được vô số những siêu phẳng phân chia trên cùng một tập dữ liệu. Và muốn tìm đường thẳng phân chia sao cho tốt nhất, có nghĩa là có sai sót phân loại bé nhất trên bộ dữ liệu.



Hình 6: Một bộ dữ liệu hai chiều được phân chia tuyến tính

Do đó mục tiêu của phương pháp phân lớp SVM là tìm một siêu phẳng phân cách giữa hai lớp sao cho khoảng cách lề (margin) giữa hai lớp đạt cực đại.

Siêu phẳng có biên độ lớn nhất (maximum marginal hyperplane) sẽ được chọn như là siêu phẳng phân chia tập dữ liệu một cách tốt nhất. Trong hình bên dưới, ta thấy có hai siêu phẳng có thể phân chia được và những biên độ của nó. Trước khi đi vào định nghĩa của biên độ (margin), hãy nhìn vào hình trên một cách trực quan. Cả hai siêu phẳng đều phân tách tất cả những bộ dữ liệu cho trước. Một cách trực quan, siêu phẳng với biên độ lớn hơn sẽ chính xác hơn trong việc phân loại các bộ dữ liệu trong tương lai so với siêu phẳng có biên độ nhỏ hơn. Điều này là lý do tại sao (trong suốt giai đoạn học hay huấn luyện), SVM tìm những siêu phẳng có biên độ lớn nhất, gọi là MMH (maximum marginal hyperlane). Siêu phẳng có biên độ lớn nhất là siêu phẳng có khoảng cách từ nó tới hai mặt bên của nó thì bằng nhau (mặt bên song song với siêu phẳng). Khoảng cách đó thật ra là khoảng cách ngắn nhất từ MMH tới bộ dữ liệu huấn luyện gần nhất của mỗi lớp. Siêu phẳng có biên độ lớn nhất này cho một sự phân loại tốt nhất giữa các lớp.



Hình 7: Siêu phẳng phân chia tuyến tính cùng với biên độ của nó

Siêu phẳng phân cách có vai trò quan trọng trong việc phân lớp, nó quyết định xem một bộ dữ liệu sẽ thuộc về lớp nào. Để thực hiện việc phân lớp, SVM chỉ cần xác định xem một bộ dữ liệu nằm về phía nào của siêu phẳng phân cách:

$$D(x) = sign(W.X + b) \quad (2)$$

- *D(x) < 0*: bộ dữ liệu sẽ nằm phía dưới siêu phẳng phân cách
- *D(x) = 0*: bộ dữ liệu sẽ nằm trên siêu phẳng phân cách
- *D(x) > 0*: bộ dữ liệu sẽ nằm phía trên siêu phẳng phân cách

b. Support Vector

Ta có phương trình tổng quát của siêu phẳng:

$$W.X + b = 0$$

Ta xét trên ví dụ sau:

Với bộ dữ liệu huấn luyện có hai thuộc tính A_1 và A_2: $X=\{x_1, x_2\}$, với x_1, x_2 là giá trị của thuộc tính A_1, A_2. $W = \{w_1, w_2\}$. Phương trình siêu phẳng có thể viết lại:

$$w_0 + w_1x_1 + w_2x_2 = 0$$

Trong đó:

- w_0 tương đương với hằng số b trong phương trình tổng quát của siêu phẳng

Vì vậy mỗi điểm nằm trên siêu phẳng phân cách thỏa mãn:

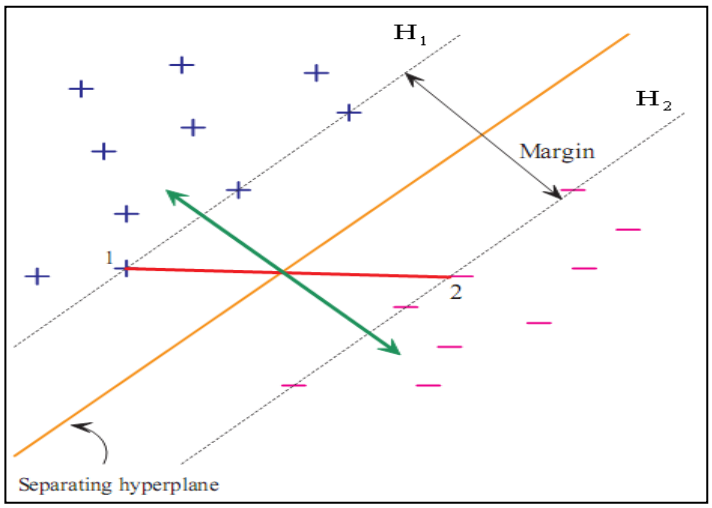
$$w_0 + w_1x_1 + w_2x_2 > 0$$

Tương tự, những điểm nằm dưới siêu phẳng phân cách phải thỏa mãn:

$$w_0 + w_1x_1 + w_2x_2 < 0$$

Bằng cách điều chỉnh trọng số w_0 ta có:

$$H_1: w_0 + w_1x_1 + w_2x_2 \geq 1 \; với \; y_i = +1$$
$$H_2: w_0 + w_1x_1 + w_2x_2 \leq -1 \; với \; y_i = -1$$



Hình 8: Đường biểu diễn H1 và H2

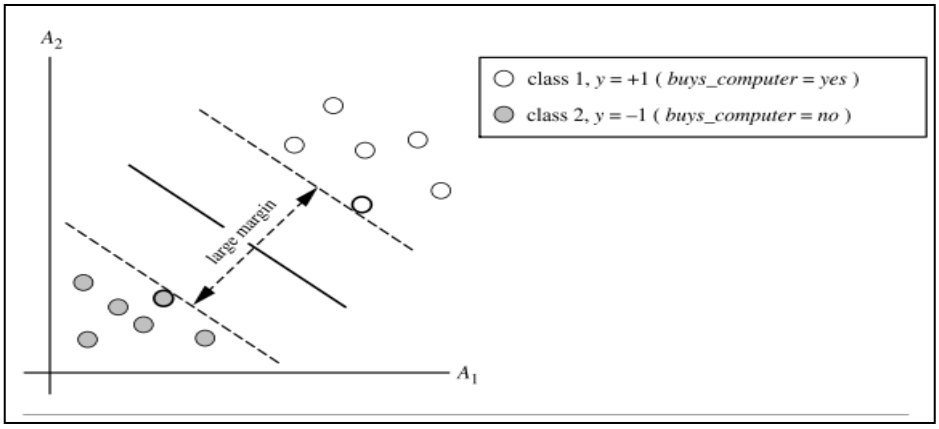
Đường màu đỏ là khoảng cách Euclidean của hai điểm 1 và 2. Đường màu xanh là khoảng cách Euclidean nhỏ nhất.

Điều này có nghĩa là nếu bất kì bộ nào nằm tại hoặc trên H_1 đều thuộc về lớp +1, và bất kì bộ nào nằm tại hoặc dưới H_2 đều thuộc về lớp -1.Kết hợp 2 bất đẳng thức trên ta có:

$$y_i(w_0 + w_1x_1 + w_2x_2) \geq 1, \forall i$$

Mỗi bộ huấn luyện nằm tại các mặt biên H_1 hay H_2 thỏa mãn phương trình trên được gọi là support vectors. Support vectors là những bộ gần với siêu phẳng phân chia tuyến tính (MMH) nhất.

Trong hình bên dưới, support vectors là hình tròn có viền dày hơn. Ta thấy rằng các support vectors là những bộ khó phân lớp nhất và cung cấp nhiều thông tin nhất cho việc phân lớp.



Hình 9: Các support vector trong SVM

c. Biên độ Margin

Từ các điều trên có thể đưa ra một công thức cho việc tính biên độ lớn nhất. Khoảng cách từ siêu phẳng phân chia đến mọi điểm tại H1 là $\frac{1}{||W||}$

Trong đó:

- $||W||$ là khoảng cách Euclidean chuẩn của W là $\sqrt{W * W}$. Với W = {w_1, w_2} khi đó $\sqrt{W * W} = \sqrt{w_1^2 + w_2^2}$.

Theo định nghĩa, khoảng cách từ siêu phẳng đến H_1 bằng với khoảng cách từ mọi điểm tại H_2 đến siêu phẳng. Vì vậy, kích thước của biên độ cực đại là $\frac{2}{||W||}$.

d. Phân lớp dữ liệu

Trường hợp dữ liệu có thể phân chia tuyến tính được:

Việc huấn luyện SVM với mục đích là để tìm ra các support vectors và MMH. MMH là ranh giới phân chia tuyến tính giữa các lớp và vì thế SVM tương ứng có thể được sử dụng để phân lớp dữ liệu mà dữ liệu đó có thể phân chia tuyến tính. Bài toán xem SVM được huấn luyện là SVM tuyến tính.

Sau khi huấn luyện SVM, bài toán sẽ phân loại các bộ mới. Dựa trên công thức Lagrangian ta có:

$$d(X^T) = \sum_{i=1}^{l} y_i \alpha_i X^T + b_0$$

Trong đó:

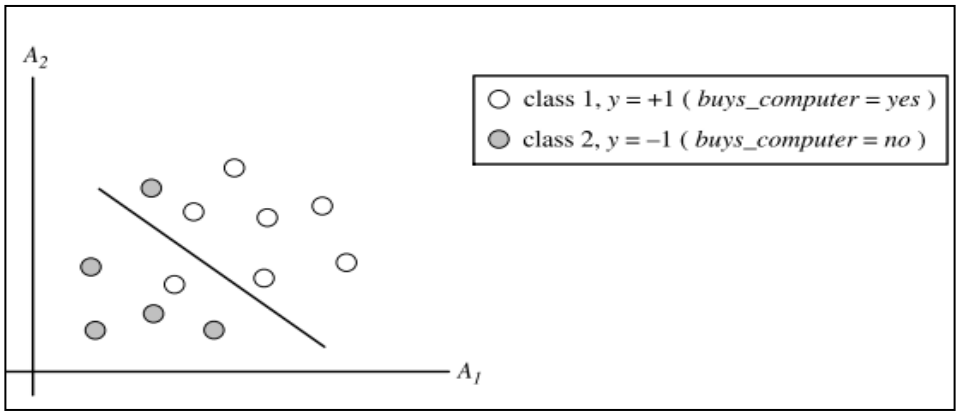
- y_i là nhãn lớp của support vector X_i
- X^T là một bộ test
- α (nhân tử Lagrangian)
- b_0 là biến số được xác định bởi sự tối ưu hóa hay các thuật toán SVM

- l là số lượng các support vectors.

MMH có thể được xem như "ranh giới quyết định" trong việc quyết định xem một bộ test bất kỳ sẽ thuộc vào lớp nào. Cho một bộ test X^T, gắn nó vào phương trình trên, và sau đó kiểm tra dấu của kết quả. Từ đó ta sẽ biết được bộ test sẽ rơi vào mặt nào của siêu phẳng. Nếu dấu là dương, thì X^T rơi vào phía trên của MMH, và SVM đoán rằng X^T thuộc về lớp +1. Nếu dấu là âm, thì X^T nằm tại hoặc dưới MMH và nhãn lớp được đoán là -1.

Trường hợp dữ liệu không thể chia tuyến tính được :

Trong phần trên đề cập đến trường hợp SVM phân lớp những dữ liệu có thế phân chia tuyến tính, nhưng nếu dữ liệu không thể phân chia tuyến tính thì sao? Trong trường hợp này không có đường thẳng nào có thể vẽ được để phân chia các lớp này. SVM tuyến tính mà đã học thì không đem lại lời giải khả thi trong trường hợp này.



Hình 10: Một trường hợp đơn giản trên không gian 2 chiều

Tuy nhiên hướng tiếp cận của SVM tuyến tính có thể được mở rộng để tạo ra SVM không tuyến tính cho việc phân lớp các dữ liệu không thể phân chia tuyến tính (hay gọi tắt là dữ liệu không tuyến tính). Những SVM như vậy có khả năng tìm những

ranh giới quyết định không tuyến tính (những mặt không tuyến tính) trong không gian đầu vào.

"Làm thế nào mở rộng tiếp cận tuyến tính?". Thu được SVM phi tuyến bằng cách mở rộng SVM tuyến tính như sau. Có hai bước chính:

Bước 1: Chuyển dữ liệu nguồn lên một không gian nhiều chiều hơn bằng cách sử dụng ánh xạ phi tuyến. Một vài ánh xạ phi tuyến thông thường có thể được sử dụng để thực hiện bước này

Bước 2: Tìm những siêu phẳng trong không gian mới này. Cuối cùng là lại quay lại vấn đề tối ưu bình phương đã được giải quyết sử dụng công thức SVM tuyến tính. Siêu phẳng có biên độ lớn nhất được tìm thấy trong không gian mới tương ứng với siêu bề mặt phân chia không tuyến tính trong không gian ban đầu.

2.3 Cách đánh giá

Độ chính xác (Accuracy) và độ đo F (F-measure) được sử dụng để đánh giá hiệu suất của các mô hình được đề xuất.

Các chỉ số được tính toán thông qua 2 chỉ số Presision và Recall, thông qua True Positive (TP), False Positive (FP), True Negative (TN) và False Negative (FN).

- $Precision_{positive} = \frac{TP}{TP+FP}$
- $Precision_{negative} = \frac{TN}{TN+FN}$
- $Recall_{positive} = \frac{TP}{TP+FN}$
- $Recall_{positive} = \frac{TN}{TN+FP}$

Độ chính xác (Accuracy) và độ đo F (F-measure) được tính toán theo công thức:

- $Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$
- $F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$

CHƯƠNG III – THỰC NGHIỆM

3.1 Dữ liệu

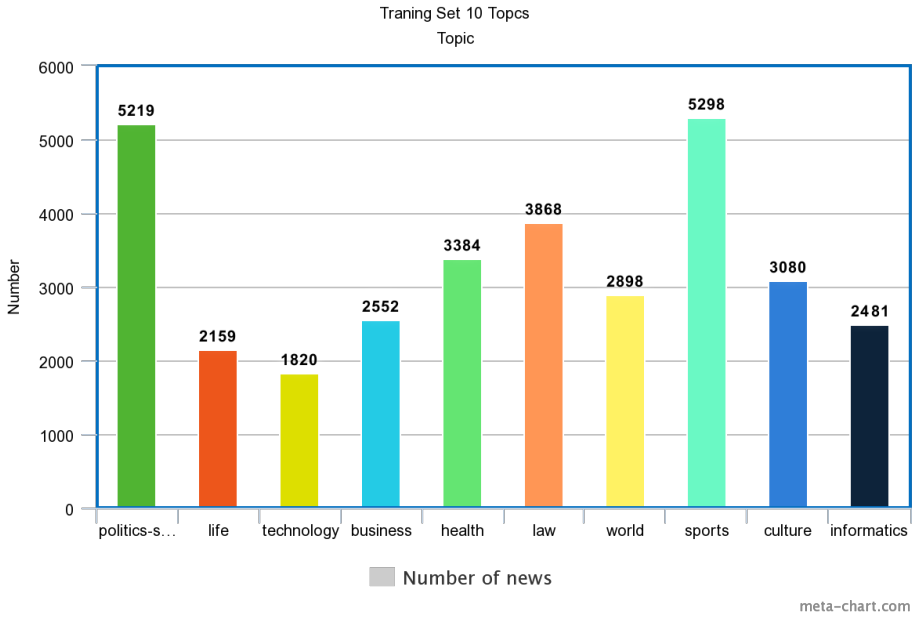
Đề tài sử dụng bộ dữ liệu "A Large-scale Vietnamese News Text Classification Corpus" được sử dụng trong bài báo: A Comparative Study on Vietnamese Text Classification Methods - Cong Duy Vu Hoang, Dien Dinh, Le Nguyen Nguyen, Quoc Hung Ngo.

Bộ dữ liệu bao gồm 2 loại: 10-topics và 27-topics. Tuy nhiên về một số giới hạn về thời gian và tài nguyên, nên đề tài sử dụng bộ dữ liệu 10-topics bao gồm 10 chủ đề: Chính trị, Đời sống, Kinh doanh, Khoa học, Pháp luật, Sức khỏe, Thế giới, Thể thao, văn hóa, Vi tính.

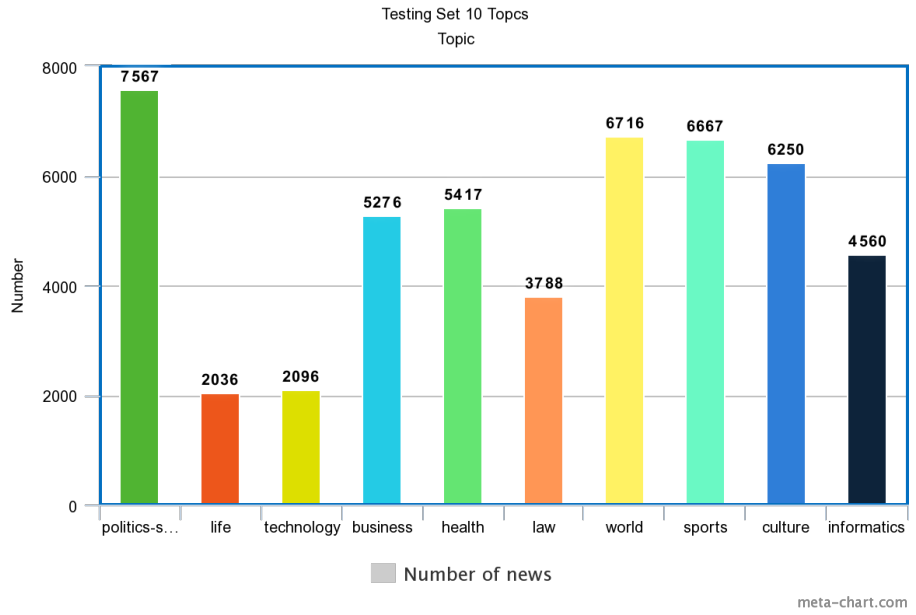
Bộ dữ liệu có tổng cộng 84,132 văn bản, trong đó được chia ra làm 2 loại:

- Huấn luyện: 33,759 văn bản
- Kiểm tra: 50,373 văn bản

Hình dưới minh họa về bộ dữ liệu 10-topics.



Hình 11: Bộ dữ liệu huấn luyện trong 10-topics



Hình 12: Bộ dữ liệu kiểm tra trong 10-topics

3.2 Công nghệ

3.2.1 Ngôn ngữ lập trình

Toàn bộ chương trình demo được viết bằng ngôn ngữ Python, vì tính nhanh gọn và các thư viện hỗ trợ cực kì tốt, đặc biệt là trong lĩnh vực Khoa học dữ liệu (Data Science).

Để chạy được demo, máy tính cần cài đặt Python phiên bản 3.7 hoặc cao hơn, có thể tham khảo đường dẫn cài đặt Python ở trang web: https://www.python.org/

3.2.2 Thư viện

Một số thư viện được áp dụng trong chương trình demo:

- ViTokenizer: Sử dụng cho tiền xử lý văn bản như tách từ, v.v…
- gensim: Thư viện cho xử lý ngôn ngữ tự nhiên, sử dụng để xây dựng từ điển và rút các đặc trưng BoW của văn bản.
- sklearn: Sử dụng LinearSVC cho mô hình phân lớp.
- pickle: Sử dụng để save và load model đã được huấn luyện.

3.3 Thực nghiệm

Vì thiếu thời gian cũng như tài nguyên phần cứng, nhóm chỉ tiến hành traning một phần của bộ dữ liệu. Chủ đề về "Công nghệ" có ít dữ liệu nhất (1820 văn bản), chính vì thế nhóm tiến hành chạy 1801 văn bản cho mỗi chủ đề.

Các bước thực hiện đối với huấn luyện mô hình:

- Load data từ bộ dữ liệu 10-topics và lưu dưới dạng JSON với format như sau:

```
{
        'category': 'Nhãn của văn bản'
        'content': 'Nội dung văn bản'
}
```

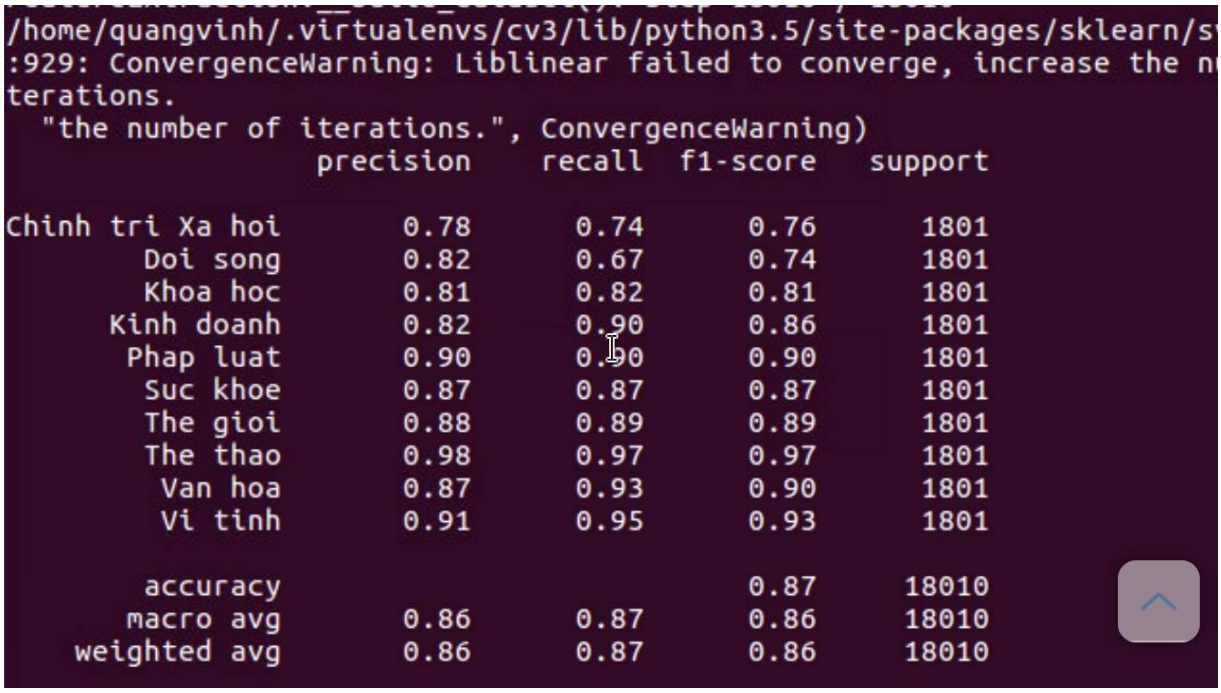
- Sử dụng ViTokenizer tách các từ trong văn bản, loại từ dừng và lưu thành từ điển.
- Sử dụng gensim rút đặc trưng Bag-of-Words, chuyển văn bản thành dạng vector số thực để xử lý.
- Sử dụng các feature vector của tập train và tập test đưa vào bộ phân lớp SVM để tiến hành huấn luyện và validation.
- Sau khi huấn luyện lưu mô hình xuống file **'linear_svc_model.pk'**
- In bảng report quá trình validation.

Các bước thực hiện test 1 đoạn văn bản bất kỳ:

- Sử dụng ViTokenizer tách các từ trong văn bản, loại từ dừng và lưu thành từ điển.
- Sử dụng gensim rút đặc trưng Bag-of-Words, chuyển văn bản thành dạng vector số thực để xử lý.
- Load model được lưu trong **'linear_svc_model.pk'** thực hiện dự đoán kết quả và in ra nhãn được dự đoán.

3.4 Đánh giá kết quả

Kết quả quá trình huấn luyện được thể hiện trong hình dưới



Hình 13: Kết quả quá trình huấn luyện

Độ chính xác trung bình: 87%

Thực hiện test thử 1 đoạn bất kỳ, nội dung của đoạn văn bản được cắt từ báo tuổi trẻ, chuyên mục ‘Văn hóa’, đường dẫn bài báo: https://tuoitre.vn/chibooks-dua-sach-viet-vao-thi-truong-trung-quoc-20191025134047412.htm

Nội dung đoạn văn bản được cắt ra như sau:

“Đây là lần thứ ba sách Việt Nam được triển lãm tại Trung Quốc, đều do nỗ lực của dịch giả Nguyễn Lệ Chi và thương hiệu sách Chibooks "tự thân vận động": 2006 (Hội chợ triển lãm Trung Quốc - ASEAN (CAEXPO), 2016 (Hội sách quốc tế Bắc Kinh) và 2019 (Triển lãm sách Quảng Tây). "Mục tiêu của Chibooks là nỗ lực đưa sách Việt ra với thị trường thế giới" - dịch giả Nguyễn Lệ Chi chia sẻ. Dịp này, các sách tiếng Việt được giới thiệu tại "Quảng Tây thư triển" gồm: Vắt qua những ngàn mây (của Đỗ Quang Tuấn Hoàng), From Zero to Hero (Ray Đoàn Huy -Toàn Juno), Dành cả thanh xuân để chạy theo idol (Hồng Trân); các sách dịch như Ếch (Mạc Ngôn), Cây thạch lựu bói trái anh đào (Lý Nhĩ - tác giả vừa đoạt giải Mao Thuẫn 2019), Mộng đổi đời (Đông Tây)...”

Kết quả: ‘Văn hóa’

```
(cv) (base) TMAs-Mac-mini:source_code tma$ python Main.py
Đây là lần thứ ba sách Việt Nam được triển lãm tại Trung Quốc, đều do nỗ lực của dịch giả Nguyễn Lệ Ch
i và thương hiệu sách Chibooks "tự thân vận động": 2006 (Hội chợ triển lãm Trung Quốc - ASEAN (CAEXPO)
, 2016 (Hội sách quốc tế Bắc Kinh) và 2019 (Triển lãm sách Quảng Tây). "Mục tiêu của Chibooks là nỗ lự
c đưa sách Việt ra với thị trường thế giới" - dịch giả Nguyễn Lệ Chi chia sẻ. Dịp này, các sách tiếng
Việt được giới thiệu tại "Quảng Tây thư triển" gồm: Vắt qua những ngàn mây (của Đỗ Quang Tuấn Hoàng),
From Zero to Hero (Ray Đoàn Huy -Toàn Juno), Dành cả thanh xuân để chạy theo idol (Hồng Trân); các sác
h dịch như Ếch (Mạc Ngôn), Cây thạch lựu bói trái anh đào (Lý Nhĩ - tác giả vừa đoạt giải Mao Thuẫn 20
19), Mộng đổi đời (Đông Tây)...

['Van hoa']
(cv) (base) TMAs-Mac-mini:source_code tma$
```

Hình 14: Kết quả test 1 đoạn văn bản bất kỳ

CHƯƠNG IV – KẾT LUẬN

4.1 Kết quả đạt được

4.1.1 Kết quả

Một số kết quả đạt được:

- Tìm hiểu và áp dụng các phương pháp xử lý văn bản.
- Tìm hiểu và áp dụng các phương pháp vector hóa văn bản, rút trích đặc trưng dựa trên số lượng, tần suất từ trong văn bản.
- Tìm hiểu và áp dụng phương pháp học máy SVM để phân loại dữ liệu văn bản theo chủ đề.
- Độ chính xác của mô hình phân loại tương đối cao, khoảng 87%.

4.1.2 Hạn chế

Một số điểm hạn chế cần phát triển thêm:

- Sử dụng dữ liệu có sẵn, chưa xây dựng được bộ dữ liệu để sử dụng.
- Chưa đánh giá trên toàn bộ dữ liệu do thiếu tài nguyên phần cứng.
- Độ chính xác cần được cải thiện.
- Mô hình BoW chỉ quan tâm tới số lần xuất hiện của từ, không quan tâm tới ngữ nghĩa từ trong văn bản. Cần tìm hiểu áp dụng thêm các phương pháp khác như Word2vec, v.v…

4.2 Hướng phát triển

Một số hướng phát triển cho đề tài:

- Tìm hiểu và áp dụng các phương pháp xử lý văn bản để giảm tối đa độ phức tạp cũng như thời gian xử lý của thuật toán.
- Nghiên cứu các phương pháp rút đặc trưng văn bản dựa trên ngữ nghĩa để tăng độ chính xác.
- Áp dụng thêm các giải thuật học sâu Deep Learning và so sánh kết quả với mô hình hiện tại.

TÀI LIỆU THAM KHẢO

Tiếng Anh

[1] JVnTextPro: A Java-based Vietnamese Text Processing Tool - http://jvntextpro.sourceforge.net/

[2] https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

[3] Introduction to Word Embedding and Word2Vec - https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa

Tiếng Việt

[1] Phân loại văn bản Tiếng Việt tự động - Phần 1 - https://viblo.asia/p/phan-loai-van-ban-tieng-viet-tu-dong-phan-1-yMnKM3bal7P

[2] Underthesea Wiki - https://github.com/undertheseanlp/underthesea/wiki

[3] Xây dựng mô hình không gian vector cho Tiếng Việt - https://viblo.asia/p/xay-dung-mo-hinh-khong-gian-vector-cho-tieng-viet-GrLZDXr2Zk0