|  |  |
| --- | --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO** | **BỘ NÔNG THÔN VÀ PTNT** |
| **TRƯỜNG ĐẠI HỌC THỦY LỢI** | |



LÊ CÔNG MINH

**TÌM HIỂU PHƯƠNG PHÁP LIFELONG MACHINE LEARNING VÀ ỨNG DỤNG CHO BÀI TOÁN PHÁT HIỆN Ý ĐỊNH MUA BÁN**

ĐỒ ÁN TỐT NGHIỆP

HÀ NỘI, NĂM 2023

|  |  |
| --- | --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO** | **BỘ NÔNG THÔN VÀ PTNT** |
| **TRƯỜNG ĐẠI HỌC THỦY LỢI** | |

LÊ CÔNG MINH

**TÌM HIỂU PHƯƠNG PHÁP LIFELONG MACHINE LEARNING VÀ ỨNG DỤNG CHO BÀI TOÁN PHÁT HIỆN Ý ĐỊNH MUA BÁN**

|  |  |
| --- | --- |
| Ngành: | Công nghệ thông tin |
| Mã số: | 1951060862 |

|  |  |
| --- | --- |
| GIẢNG VIÊN HƯỚNG DẪN: | PGS.TS LÊ ĐỨC HẬU |

HÀ NỘI, NĂM 2023

**GÁY BÌA ĐỒ ÁN TỐT NGHIỆP, KHÓA LUẬN TỐT NGHIỆP**

**HỌ VÀ TÊN: LÊ CÔNG MINH ĐỒ ÁN/KL TỐT NGHIỆP HÀ NỘI, NĂM 2023**

**LỜI CAM ĐOAN**

Em tên là Lê Công Minh, em xin cam đoan đây là Đồ án tốt nghiệp / Khóa luận tốt nghiệp do chính bản thân em làm. Các kết quả trong Đồ án tốt nghiệp / Khóa luận tốt nghiệp này mà em làm là trung thực, không sao chép toàn bộ từ bất cứ một nguồn nào, dưới mọi hình thức nào. Việc tham khảo các nguồn tài liệu (nếu có) đã được em thực hiện trích dẫn và ghi nguồn tài liệu tham khảo một cách rõ ràng, đúng quy định.

|  |  |
| --- | --- |
|  | **Tác giả ĐATN / KLTN** |
|  |  |
|  | **Lê Công Minh** |

**LỜI CẢM ƠN**

Em xin gửi lời cảm ơn tới các thầy cô trường Đại học Thủy Lợi và những người bạn đã đồng hành cùng em trong suốt quá trình học tập và rèn luyện trong những năm tháng đại học. Em xin chân thành cảm ơn PGS.TS Lê Đức Hậu đã giúp đỡ rất nhiều để em có thể hoàn thành đồ án, giúp em bổ sung kiến thức về bài toán xử lý ngôn ngữ tự nhiên trong lĩnh vực trí tuệ nhân tạo.

Trong quá trình thực hiện đồ án tốt nghiệp do kiến thức còn hạn chế và thiếu kinh nghiệm thực tế nên không thể tránh khỏi những sai sót. Em rất mong nhận được những ý kiến đóng góp của các thầy cô để đồ án được hoàn thiện một cách tốt nhất.

Em xin chân thành cảm ơn !

|  |  |
| --- | --- |
| Description: Logo-WRU | **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập - Tự do - Hạnh phúc**  ----------★----------  **NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP** |

|  |  |
| --- | --- |
| Họ tên sinh viên: **Lê Công Minh** | Hệ đào tạo: **Đại học chính quy** |
| Lớp: **61TH3** | Ngành: **Công nghệ thông tin** |
| Khoa: **Công nghệ thông tin** |  |

1. TÊN ĐỀ TÀI:

**TÌM HIỂU PHƯƠNG PHÁP LIFELONG MACHINE LEARNING VÀ ỨNG DỤNG CHO BÀI TOÁN PHÁT HIỆN Ý ĐỊNH MUA BÁN**

1. CÁC TÀI LIỆU CƠ BẢN:
2. Z. Chen and B. Liu, "Lifelong Machine Learning," in *Synthesis Lectures on Artificial Intelligence and Machine Learning*, Morgan and Claypool, 2016.

|  |
| --- |
| 1. G. Fei et al, "Learning Cumulatively to Become More Knowledgeable," *KDD,* 2016. |
| 1. Z. Chen et al, "Lifelong Learning for Sentiment Classification," *ACL,* pp. 750-756, 2015. |
| 1. "What is Transfer learning ?," Geeksforgeeks, 29 11 2023. [Online]. Available: https://www.geeksforgeeks.org/ml-introduction-to-transfer-learning/. |
| 1. Z. Chen and B. Liu, "Lifelong and Continual Learning," [Online]. Available: https://www.cs.uic.edu/~liub/lifelong-learning.html. |
| 1. "Difference between single task learning and multitask learning," ResearchGate, [Online]. Available: https://www.researchgate.net/figure/Difference-between-single-task-learning-and-multitask-learning-a-single-task-learning\_fig2\_307622018. |
| 1. N. T. Trung and T. M. Tuấn, "Phân tích dữ liệu lớn: Phần 2," Đại học Thủy Lợi, [Online]. Available: http://tailieuso.tlu.edu.vn/handle/DHTL/8914. |
| 1. V. H. Tiệp, "Bài 32: Naive Bayes Classifier," 8 8 2017. [Online]. Available: https://machinelearningcoban.com/2017/08/08/nbc/. |
| 1. "N-Grams," deepai.org, [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/n-gram. |
| 1. N. X. Bách et al, "Cross-Domain Intention Detection in Discussion Forums," *SoICT,* pp. 173-180, 2017. |
| 1. V. H. Tiệp, "Bài 8: Gradient Descent (phần 2/2)," 16 1 2017. [Online]. Available: https://machinelearningcoban.com/2017/01/16/gradientdescent2/. |
| 1. "Các phương pháp đánh giá mô hình học máy, học sâu (Machine learning & Deep learning)," Rabiloo, [Online]. Available: https://rabiloo.com/vi/blog/cac-phuong-phap-danh-gia-mo-hinh-machine-learning-va-deep-learning. |
| 1. Z. Chen et al, "Identifying Intention Posts in Discussion Forums," *ACL,* p. 1041–1050. |
| 1. "Stanford Tokenizer," [Online]. Available: https://nlp.stanford.edu/software/tokenizer.html. |
| 1. "Java Tutorial," w3schools, [Online]. Available: https://www.w3schools.com/java/ |

1. NỘI DUNG CÁC PHẦN THUYẾT MINH VÀ TÍNH TOÁN:

|  |  |
| --- | --- |
| **Nội dung các phần** | **Tỷ lệ** |
| Chương 1. Giới thiệu | 10% |
| Chương 2. Cơ sở lý thuyết | 35% |
| Chương 3. Ứng dụng phương pháp cho bài toán | 15% |
| Chương 4. Thực nghiệm và kết quả | 25% |
| Chương 5. Kết luận | 15% |

1. NGÀY GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Ngày .... tháng …. năm 2023.

|  |  |  |
| --- | --- | --- |
| **Trưởng Bộ môn**  *(Ký và ghi rõ Họ tên)* |  | **Giáo viên hướng dẫn chính**  *(Ký và ghi rõ Họ tên)* |

Nhiệm vụ Đồ án tốt nghiệp đã được Hội đồng thi tốt nghiệp của Khoa thông qua.

|  |
| --- |
| Ngày …. tháng …. năm 2023.  **Chủ tịch Hội đồng**  *(Ký và ghi rõ Họ tên)* |

Sinh viên đã hoàn thành và nộp bản Đồ án tốt nghiệp cho Hội đồng thi.

|  |
| --- |
| Ngày .... tháng …. năm 2023.  **Sinh viên làm Đồ án tốt nghiệp**  *(Ký và ghi rõ Họ tên)* |

|  |  |
| --- | --- |
| logo | TRƯỜNG ĐẠI HỌC THUỶ LỢI  **KHOA CÔNG NGHỆ THÔNG TIN**  BẢN TÓM TẮT ĐỀ CƯƠNG ĐỒ ÁN TỐT NGHIỆP |

**Tên đề tài: Tìm hiểu phương pháp LIFELONG MACHINE LEARNING và ứng dụng cho bài toán phát hiện ý định mua bán**

*Sinh viên thực hiện*: **Lê Công Minh**

*Lớp*: **61TH3**

*Mã sinh viên:* **1951060862**

*Số điện thoại:* **0365829001**

*Email:* **1951060862@e.tlu.edu.vn**

*Giáo viên hướng dẫn*: **PGS.TS Lê Đức Hậu**

**TÓM TẮT ĐỀ TÀI**

Hiện nay các phương pháp học máy thông thường hầu hết đều là các phương pháp học có giới hạn: chúng ta đưa đầu vào là một tập dữ liệu huấn luyện, tiếp đó thực hiện giải thuật học máy trên bộ dữ liệu đó để tạo ra mô hình học máy ứng dụng cho thực tế. Những phương pháp học máy này thường bị giới hạn bởi việc không tích lũy tri thức đã học được. Trong những trường hợp cụ thể các phương pháp học máy thông thường có thể đạt được kết quả cao nhưng yêu cầu tập dữ liệu huấn luyện là rất lớn và mục tiêu cho bài toán phải được xác định cụ thể.

Phương pháp học máy LIFELONG MACHINE LEARNING (học cả đời) là một phương pháp học máy nâng cao có thể học liên tục, tích lũy những tri thức học được từ quá khứ để sử dụng cho các nhiệm vụ trong tương lai. Càng học sẽ càng tích lũy được nhiều tri thức, giống với cách học của con người chúng ta. Điều này là rất quan trọng với sự bùng nổ dữ liệu trong thời đại công nghệ hiện nay.

Đồ án này sẽ trình bày những kiến thức cơ bản về LIFELONG MACHINE LEARNING và áp dựng cùng phương pháp Naïve Bayes với cho bài toán phát hiện ý định mua bán thông qua bình luận trên các diễn đàn sử dụng bộ dữ liệu NAACL2013-Chen-Intention cho việc nghiên cứu. Bên cạnh các lý do nêu trên, việc thực hiện đồ án này còn cho phép sinh viên được tự mình tìm hiểu và thực nghiệm, tích lũy tri thức cho con đường sự nghiệp tương lai.

**CÁC MỤC TIÊU CHÍNH**

* Tìm hiểu cơ bản về phương pháp LIFELONG MACHINE LEARNING.
* Áp dụng phương pháp LIFELONG MACHINE LEARNING cho bài toán phát hiện ý định mua bán sử dụng giải thuật Naïve Bayes, Gradient Descent, trích chọn đặc trưng dựa vào độ lợi thông tin và sử dụng độ đo Precision, Recall, F1 cho các thực nghiệm.

**KẾT QUẢ DỰ KIẾN**

* Nắm được các vấn đề cơ bản về phương pháp LIFELONG MACHINE LEARNING.
* Xây dựng chương trình hoàn chỉnh.
* Đưa ra nhận xét về phương pháp LIFELONG MACHINE LEARNING cùng hướng phát triển bài toán.
* Hoàn thiện luận văn tốt nghiệp.

**TÀI LIỆU THAM KHẢO**

|  |
| --- |
| * G. Fei et al, "Learning Cumulatively to Become More Knowledgeable," *KDD,* 2016. |
| * Z. Chen et al, "Lifelong Learning for Sentiment Classification," *ACL,* pp. 750-756, 2015. |
| * "What is Transfer learning ?," Geeksforgeeks, 29 11 2023. [Online]. Available: https://www.geeksforgeeks.org/ml-introduction-to-transfer-learning/. |
| * Z. Chen and B. Liu, "Lifelong and Continual Learning," [Online]. Available: https://www.cs.uic.edu/~liub/lifelong-learning.html. |
| * "Difference between single task learning and multitask learning," ResearchGate, [Online]. Available: https://www.researchgate.net/figure/Difference-between-single-task-learning-and-multitask-learning-a-single-task-learning\_fig2\_307622018. |
| * N. T. Trung and T. M. Tuấn, "Phân tích dữ liệu lớn: Phần 2," Đại học Thủy Lợi, [Online]. Available: http://tailieuso.tlu.edu.vn/handle/DHTL/8914. |
| * V. H. Tiệp, "Bài 32: Naive Bayes Classifier," 8 8 2017. [Online]. Available: https://machinelearningcoban.com/2017/08/08/nbc/. |
| * "N-Grams," deepai.org, [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/n-gram. |
| * N. X. Bách et al, "Cross-Domain Intention Detection in Discussion Forums," *SoICT,* pp. 173-180, 2017. |
| * V. H. Tiệp, "Bài 8: Gradient Descent (phần 2/2)," 16 1 2017. [Online]. Available: https://machinelearningcoban.com/2017/01/16/gradientdescent2/. |
| * "Các phương pháp đánh giá mô hình học máy, học sâu (Machine learning & Deep learning)," Rabiloo, [Online]. Available: https://rabiloo.com/vi/blog/cac-phuong-phap-danh-gia-mo-hinh-machine-learning-va-deep-learning. |
| * Z. Chen et al, "Identifying Intention Posts in Discussion Forums," *ACL,* p. 1041–1050. |
| * "Stanford Tokenizer," [Online]. Available: https://nlp.stanford.edu/software/tokenizer.html. |
| * "Java Tutorial," w3schools, [Online]. Available: https://www.w3schools.com/java/. |

**MỤC LỤC**

[DANH MỤC CÁC HÌNH ẢNH xi](#_Toc155361513)

[DANH MỤC BẢNG BIỂU xiii](#_Toc155361514)

[DANH MỤC CÁC TỪ/TỪ VIẾT TẮT VÀ GIẢI THÍCH THUẬT NGỮ xiv](#_Toc155361515)

[CHƯƠNG 1: GIỚI THIỆU 1](#_Toc155361516)

[1.1 Lý do chọn đề tài 1](#_Toc155361517)

[1.2 Mục tiêu nghiên cứu. 1](#_Toc155361518)

[1.3 Đối tượng nghiên cứu 2](#_Toc155361519)

[1.4 Phạm vi nghiên cứu 2](#_Toc155361520)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 4](#_Toc155361521)

[2.1 Tổng quan về học máy 4](#_Toc155361522)

[2.2 Tổng quan về phương pháp Lifelong machine learning 5](#_Toc155361523)

[2.2.1 Lịch sử hình thành của phương pháp Lifelong machine learning 5](#_Toc155361524)

[2.2.2 Phương pháp Lifelong machine learning 6](#_Toc155361525)

[2.3 Giải thuật phân loại Multinomial Naive Bayes (MNB) 11](#_Toc155361526)

[2.4 Lý thuyết về N-Grams 14](#_Toc155361527)

[2.5 Phương pháp trích chọn đặc trưng Information Gain (IG) 14](#_Toc155361528)

[2.6 Phương pháp Stochastic gradient descent (SGD) 16](#_Toc155361529)

[2.7 Các thang đo đánh giá mô hình học máy 19](#_Toc155361530)

[2.8 Cách áp dụng Lifelong machine learning cho đồ án 21](#_Toc155361531)

[2.8.1 Tổng quan về cách áp dụng 21](#_Toc155361532)

[2.8.2 Chi tiết về cách áp dụng 22](#_Toc155361533)

[CHƯƠNG 3: ỨNG DỤNG PHƯƠNG PHÁP CHO BÀI TOÁN 27](#_Toc155361534)

[3.1 Mô tả bài toán 27](#_Toc155361535)

[3.2 Ngôn ngữ và thư viện dùng xây dựng 28](#_Toc155361536)

[3.2.1 Ngôn ngữ Java 28](#_Toc155361537)

[3.2.2 Thư viện Stanford NLP 31](#_Toc155361538)

[3.3 Bộ dữ liệu sử dụng 33](#_Toc155361539)

[CHƯƠNG 4: THỰC NGHIỆM VÀ KẾT QUẢ 34](#_Toc155361540)

[4.1 Các bước thực hiện 34](#_Toc155361541)

[4.1.1 Tiền xử lý dữ liệu 34](#_Toc155361542)

[4.2 Thực nghiệm 35](#_Toc155361543)

[4.2.1 Thực nghiệm 1 35](#_Toc155361544)

[4.2.2 Thực nghiệm 2 38](#_Toc155361545)

[4.2.3 Thực nghiệm 3 39](#_Toc155361546)

[4.2.4 Thực nghiệm 4 40](#_Toc155361547)

[4.3 Tổng hợp kết quả và đánh giá 43](#_Toc155361548)

[4.3.1 Tổng hợp kết quả 43](#_Toc155361549)

[4.3.2 Đánh giá kết quả 44](#_Toc155361550)

[CHƯƠNG 5: KẾT LUẬN 45](#_Toc155361551)

[TÀI LIỆU THAM KHẢO 47](#_Toc155361552)

DANH MỤC CÁC HÌNH ẢNH

[Hình 2.1 Sơ đồ một hệ thống học máy truyền thống [1] 5](#_Toc155361121)

[Hình 2.2 Kiến trúc một hệ thống Lifelong machine learning [1] 7](#_Toc155361122)

[Hình 2.3 Sơ đồ hệ thống Transfer learning [4] 9](#_Toc155361123)

[Hình 2.4 Sơ đồ hệ thống Multi-task learning [6] 9](#_Toc155361124)

[Hình 2.5 Minh họa về điểm cực tiểu [11] 17](#_Toc155361125)

[Hình 2.6 Sơ đồ áp dụng phương pháp Lifelong machine learning cho bài toán [10] 22](#_Toc155361126)

[Hình 3.1 Ngôn ngữ lập trình Java 29](#_Toc155361127)

[Hình 3.2 Thư viện Stanford NLP 32](#_Toc155361128)

[Hình 4.1 Cấu trúc một file dữ liệu không mang ý định mua bán 34](#_Toc155361129)

[Hình 4.2 Định dạng dữ liệu sau khi lấy nội dung trong thẻ Content 35](#_Toc155361130)

[Hình 4.3 Kết quả thực nghiệm 1 cho domain TV 36](#_Toc155361131)

[Hình 4.4 Kết quả thực nghiệm 1 cho domain Electronics 36](#_Toc155361132)

[Hình 4.5 Kết quả thực nghiệm 1 cho domain Camera 37](#_Toc155361133)

[Hình 4.6 Kết quả thực nghiệm 1 cho domain Cellphone 37](#_Toc155361134)

[Hình 4.7 Kết quả thực nghiệm 2 cho domain TV 38](#_Toc155361135)

[Hình 4.8 Kết quả thực nghiệm 2 cho domain Electronics 38](#_Toc155361136)

[Hình 4.9 Kết quả thực nghiệm 2 cho domain Camera 38](#_Toc155361137)

[Hình 4.10 Kết quả thực nghiệm 2 cho domain Cellphone 38](#_Toc155361138)

[Hình 4.11 Kết quả thực nghiệm 3 cho domain TV 39](#_Toc155361139)

[Hình 4.12 Kết quả thực nghiệm 3 cho domain Electronics 39](#_Toc155361140)

[Hình 4.13 Kết quả thực nghiệm 3 cho domain Camera 40](#_Toc155361141)

[Hình 4.14 Kết quả thực nghiệm 3 cho domain Cellphone 40](#_Toc155361142)

[Hình 4.15 Kết quả thực nghiệm 4 cho domain TV 41](#_Toc155361143)

[Hình 4.16 Kết quả thực nghiệm 4 cho domain Electronics 41](#_Toc155361144)

[Hình 4.17 Kết quả thực nghiệm 4 cho domain Camera 42](#_Toc155361145)

[Hình 4.18 Kết quả thực nghiệm 4 cho domain Cellphone 42](#_Toc155361146)

DANH MỤC BẢNG BIỂU

[Bảng 3.1 Thống kê về dữ liệu sử dụng [13] 33](#_Toc155361106)

[Bảng 4.1 Ý nghĩa của các thẻ trong một dữ liệu 34](#_Toc155361107)

[Bảng 4.2 Kết quả thực nghiệm với domain TV 43](#_Toc155361108)

[Bảng 4.3 Kết quả thực nghiệm với domain Electronics 43](#_Toc155361109)

[Bảng 4.4 Kết quả thực nghiệm với domain Camera 43](#_Toc155361110)

[Bảng 4.5 Kết quả thực nghiệm với domain Cellphone 44](#_Toc155361111)

DANH MỤC CÁC TỪ/TỪ VIẾT TẮT VÀ GIẢI THÍCH THUẬT NGỮ

|  |  |
| --- | --- |
| **Từ/Từ viết tắt** | **Giải thích thuật ngữ** |
| Domain | Miền dữ liệu |
| PIS | Past Information Store: Nơi lưu trữ thông tin đã học trong quá khứ |
| KB | Knowlegde Base: Cơ sở tri thức |
| KM | Knowledge Miner: Bộ khai phá tri thức |
| KBL | Knowledge-based Learner: Bộ học dựa trên cơ sở tri thức |
| NB | Naive Bayes: Giải thuật phân loại Naïve Bayes |
| MNB | Multinomial Naive Bayes |
| GD | Gradient Descent: Phương pháp giảm gradient |
| SGD | Stochastic gradient descent |
| NLP | Natural Language Processing: Xử lý ngôn ngữ tự nhiên |
| IG | Information gain: Phương pháp trích chọn đặc trưng dựa vào độ lợi thông tin |

# GIỚI THIỆU

## Lý do chọn đề tài

Các phương pháp học máy hiện nay hầu hết là các phương pháp học có giới hạn với các bước: đầu vào là một tập dữ liệu huấn luyện, thực hiện các thuật toán học máy trên bộ dữ liệu đó để tạo ra mô hình phù hợp rồi sử dụng cho mục đích cụ thể. Điều này gây ra việc không tích lũy được tri thức đã học được trong quá trình học. Bởi vậy những phương pháp học máy thông thường nếu muốn đạt được kết quả cao thường sẽ yêu cầu tập dữ liệu huấn luyện rất lớn và chỉ phù hợp cho mục đích cụ thể.

Lifelong machine learning là một phương pháp học máy nâng cao có thể học liên tục dựa trên phương pháp học của con người mà không bị giới hạn như các phương pháp học máy truyền thống. Những tri thức học được trong quả khứ sẽ được duy trì và tích lũy cho các nhiệm vụ học trong tương lai.

Từ những đặc điểm đó, em tin rằng phương pháp Lifelong machine learning sẽ giúp ích rất nhiều cho lĩnh vực trí tuệ nhân tạo trong kỷ nguyên bùng nổ công nghệ hiện nay. Một hệ thống học máy sẽ không thể thực sự thông minh nếu không có sự tích lũy tri thức trong quá trình học.

Hiện nay các bài toán về xử lý ngôn ngữ tự nhiên đang rất được quan tâm vì mỗi ngày có đến hàng nghìn đến vài chục nghìn các bài đăng trên các diễn đàn mạng xã hội. Với các doanh nghiệp thì việc quản lý các bài đăng này và tìm ra những bài có ý định mua bán để đưa ra chiến lược tiếp thị là vô cùng quan trọng. Nếu thực hiện việc đó thủ công sẽ tốn rất nhiều thời gian và công sức. Việc áp dụng phương pháp học máy này sẽ mang lại nhiều lợi ích cho các doanh nghiệp có nhu cầu.

Từ những lý do trên, đồ án này sẽ trình bày những kiến thức cơ bản về phương pháp Lifelong machine learning và áp dụng cho đề tài xử lý ngôn ngữ tự nhiên với bài toán phát hiện ý định mua bán sử dụng giải thuật Naive Bayes.

## Mục tiêu nghiên cứu.

Dưới đây là một số mục tiêu cụ thể đặt ra cho đề tài này:

* Mục tiêu 1: Tìm hiểu về phương pháp Lifelong machine learning
* Mục tiêu 2: Áp dụng phương pháp Lifelong machine learning cho đề tài
* Mục tiêu 3: Đánh giá mô hình áp dụng phương pháp này
* Mục tiêu 4: Đưa ra hướng phát triển của mô hình

## Đối tượng nghiên cứu

Đối tượng nghiên cứu của đề tài "Tìm hiểu phương pháp Lifelong Machine Learning và ứng dụng cho bài toán phát hiện ý định mua bán" gồm các đối tượng sau:

Phương pháp Lifelong machine learning: Đối tượng chính của nghiên cứu là phương pháp Lifelong machine learning với mục tiêu là giới thiệu các thành phần cơ bản của phương pháp này và một số phương pháp học máy khác có liên quan đồng thời đưa ra phương pháp đánh giá một hệ thống học máy Lifelong machine learning.

Bài toán phát hiện ý định mua bán: Đây là một bài toán cụ thể trong lĩnh vực xử lý ngôn ngữ tự nhiên. Mục tiêu là xử lý các bài đăng và đưa ra kết luận có hay không ý định mua bán hàng hóa.

Dữ liệu huấn luyện và kiểm tra: Để phát triển mô hình cho bài toán này, đối tượng nghiên cứu cũng bao gồm các bộ dữ liệu huấn luyện và kiểm tra. Đồ án sử dụng bộ dữ liệu NAACL2013-Chen-Intention để huấn luyện, đánh giá hiệu suất và độ chính xác của mô hình.

## Phạm vi nghiên cứu

Phạm vi nghiên cứu của đề tài "Tìm hiểu phương pháp Lifelong Machine Learning và ứng dụng cho bài toán phát hiện ý định mua bán" gồm các phạm vi sau:

Nghiên cứu về phương pháp Lifelong machine learning và một số phương pháp phương pháp học máy có liên quan như Transfer learning và Multi-task learning.

Nghiên cứu về các phương pháp xử lý các bài đăng mạng xã hội. Mục tiêu là tìm ra các phương pháp để tiền xử lý dữ liệu chuẩn bị cho quá trình huấn luyện.

Hiện nay có rất nhiều phương pháp trích chọn đặc trưng có thể nói đến như TF-IDF, Bag-of-words, Information Gain,… nhưng với phạm vi của đồ án nhỏ hẹp nên sẽ chỉ trình bày về phương pháp Information Gain để trích chọn đặc trưng.

Về xử lý ngôn ngữ tự nhiên cụ thể là bài toán phân lớp có nhiều phương pháp để thực hiện như SVM, Decision Tree, Naive Bayes,… nhưng với phạm vi nhỏ hẹp của đồ án nên sẽ chỉ trình bày về phương pháp Multinomial Naïve Bayes áp dụng Lifelong Machine Learning để phục vụ cho việc nghiên cứu.

Nghiên cứu về các phương pháp và chỉ số để đánh giá hiệu suất của mô hình nhận diện các bài đăng có ý định mua bán. Mục tiêu là đo lường độ chính xác, độ nhạy, độ đặc hiệu và tìm cách cải thiện mô hình thông qua tối ưu hóa các thuật toán, thay đổi các tham số hay tăng cường tập dữ liệu huấn luyện.

Phạm vi nghiên cứu có thể được điều chỉnh và mở rộng để phù hợp với khả năng và quy mô của đồ án "Tìm hiểu phương pháp Lifelong Machine Learning và ứng dụng cho bài toán phát hiện ý định mua bán".

# 

# CƠ SỞ LÝ THUYẾT

## Tổng quan về học máy

Trong những năm gần đây, lĩnh vực trí tuệ nhân tạo và cụ thể là học máy đang nổi lên rất mạnh mẽ và xuất hiện trong nhiều lĩnh vực cuộc sống của chúng ta. Học máy tập trung vào việc phát triển các thuật toán để máy tính có khả năng tự học hỏi dựa trên dữ liệu đưa vào mà không cần phải được lập trình một cách rõ ràng.

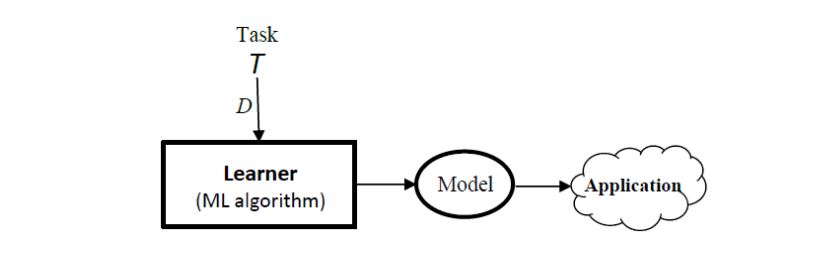
Các phương pháp học máy chủ yếu được chia làm hai loại chính: học có giám sát (Supervised Learning), học không giám sát (Unsupervised learning). Với học có giám sát, máy tính sẽ được huấn luyện từ các dữ liệu đầu vào đã được gán nhãn và dự đoán đầu ra của một dữ liệu mới tương ứng. Học có giám sát còn được tiếp tục chia nhỏ ra thành hai bài toán chính là phân lớp (Classification) và hồi quy (Regression). Một bài toán được xem là phân lớp nếu các nhãn của dữ liệu đầu vào được chia thành một số hữu hạn nhóm, ví dụ như bài toán xác định xem liệu một email có phải là spam hay không. Còn nếu nhãn của các dữ liệu đầu vào không được chia thành các nhóm mà là một giá trị thực cụ thể thì sẽ là bài toán hồi quy, ví dụ như bài toán dự đoán giá nhà.

Trong khi đó với học không giám sát thì chúng ta không biết được đầu ra hay nhãn mà chỉ có dữ liệu đầu vào. Mục tiêu của học không giám sát là dựa vào cấu trúc của dữ liệu để thực hiện một công việc nào đó, ví dụ như phân cụm (Clustering) hoặc giảm số chiều của dữ liệu (Dimension Reduction) để thuận tiện trong việc lưu trữ và tính toán.

Ngày nay các ứng dụng của học máy đã đạt được nhiều thành công rực rỡ trong các lĩnh vực như xử lý ngôn ngữ tự nhiên, nhận dạng giọng nói, thị giác máy tính và nhiều lĩnh vực khác. Học máy đang đóng vai trò quan trọng trong cuộc sống, mang lại tiềm năng lớn cho việc tự động hóa và tối ưu hóa các nhiệm vụ phức tạp.

Để áp dụng học máy cho một vấn đề cụ thể yêu cầu dữ liệu phải đủ lớn, phải hiểu biết về việc lựa chọn thuật toán, tinh chỉnh tham số, khả năng đánh giá và giải thích kết quả mô hình. Sự thành công của học máy cũng phụ thuộc rất nhiều vào độ chính xác, độ tin cậy và mức độ phức tạp của mô hình.

Tuy nhiên, việc các phương pháp học máy theo kiểu truyền thống chỉ đơn thuần sử dụng một tập dữ liệu huấn luyện có sẵn để sinh ra mô hình học máy thì sẽ không có sự tích lũy tri thức để phục vụ cho các bài toán trong tương lai. Mỗi nhiệm vụ cần học yêu cầu chúng ta phải chuẩn bị tập dữ liệu rất lớn để đảm bảo cho sự chính xác của mô hình, gây ra rất nhiều giới hạn và thường cho ra độ chính xác không đảm bảo. Cho dù thời điểm hiện nay đã có rất nhiều những phương pháp để cải tiến độ chính xác nhưng các phương pháp đó vẫn tập trung thực hiện một nhiệm vụ duy nhất, vẫn gặp nhiều vấn đề vì không tích lũy được tri thức đã học.



Hình 2.1 Sơ đồ một hệ thống học máy truyền thống [1]

Những năm gần đây học máy đã phát triển vượt bậc nhờ vào lĩnh vực mới là học sâu (Deep Learning) mang lại rất nhiều lợi ích trong xử lý những bài toán phức tạp. Nhưng do để phù hợp với nội dung của đồ án, em xin phép không trình bày về phần này.

## Tổng quan về phương pháp Lifelong machine learning

### Lịch sử hình thành của phương pháp Lifelong machine learning

Phương pháp Lifelong machine learning hay Lifelong learning được hai nhà nghiên cứu Thrun và Mitchell đề xuất lần đầu tiên vào năm 1995 để khắc phục nhược điểm của những phương pháp học máy đã nêu ra trong phần 2.1 ở phía trên. Sau đó phương pháp này được tiến hành nghiên cứu trong bốn lĩnh vực chính: Lifelong Supervised Learning, Lifelong Unsupervised Learning, Lifelong Semi-Supervised Learning và Lifelong Reinforcement Learning. Sau đây là phần trình bày về một vài nghiên cứu cụ thể của từng lĩnh vực.

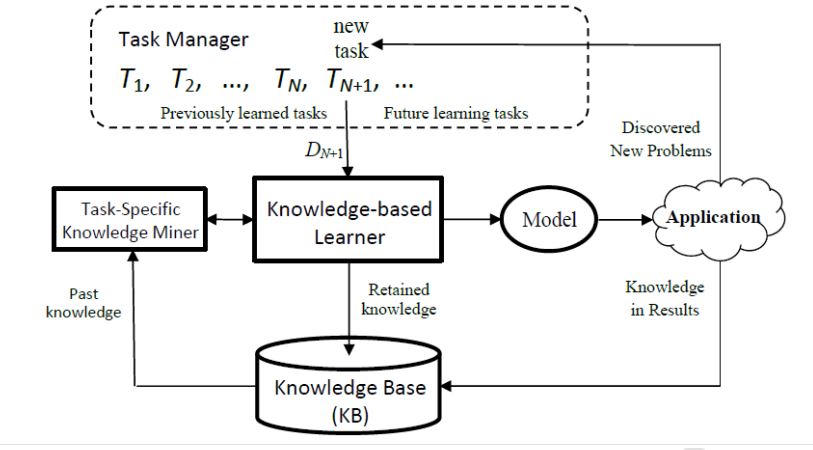
* **Lifelong Supervised Learning**: Vào năm 1996, Thrun lần đầu tiên nghiên cứu phương pháp Lifelong machine learning khi từng nhiệm vụ trong quá khứ trước đó hay nhiệm vụ mới đều nhằm nhận diện một khái niệm hoặc một lớp cụ thể bằng cách sử dụng phân lớp nhị phân (binary classification). Năm 2016, Fei et al [2] đã mở rộng cấu trúc của Lifelong machine learning với mục đích khiến phương pháp này có thể tích lũy kiến thức trong quá trình học, tức là khi gặp một lớp hay một nhiệm vụ mới thì sẽ xây dựng một bộ phân loại mới mà có thể phân loại cả những lớp cũ và mới bằng cách cập nhật lại bộ phân loại cũ dựa vào lớp mới. Năm 2015, Chen et al [3] đã đưa ra phương pháp Lifelong machine learning sử dụng giải thuật phân loại Naïve Bayesian. Năm 2014, lý thuyết về Lifelong machine learning được hoàn thiện bới Pentina và Lampert trong PAC-learning framework. [1]
* **Lifelong Unsupervised Learning**: Những bài báo về lĩnh vực này chủ yếu là về mô hình hóa chủ đề (Topic Modeling) và trích xuất thông tin trong Lifelong machine learning. Trong những năm 2014 và 2016, Zhiyuan Chen, Bing Liu và Shuai Wang đã cung cấp rất nhiều phương pháp về mô hình hóa chủ đề mà kho tri thức được xây dựng từ những nhiệm vụ trong quá khứ và sử dụng để tạo ra những chủ đề tốt hơn trong những nhiệm vụ mới. Cũng trong năm 2016, Lei Shu đã cung cấp phương pháp lifelong relaxation để giải quyết một bài toán phân loại trong chủ đề học không giám sát. [1]
* **Lifelong Semi-Supervised Learning**: Những nghiên cứu trong lĩnh vực này được biểu diễn bởi hệ thống NELL (Never-Ending Language Learner). Hệ thống này liên tục đọc dữ liệu trên Web phục vụ cho công việc trích xuất thông tin từ những năm 2010, hiện tại đã đạt tới hàng triệu thực thể. [1]
* **Lifelong Reinforcement Learning**: Năm 1995, Thurn và Mitchell đã đề xuất một vài phương pháp Lifelong machine learning về việc học cho robot. Năm 1998, Mark B Ring đã cung cấp một phương pháp học liên tục nhằm giải quyết những nhiệm vụ khó bằng cách xử lý những nhiệm vụ dễ trước. Năm 2007, Aaron Wilson cùng các nhà nghiên cứu khác đã cung cấp phương pháp quy trình quyết định Markov (Markov Decision Process). Từ đó đến nay đã có rất nhiều nghiên cứu về lĩnh vực này. [1]

### Phương pháp Lifelong machine learning

#### Định nghĩa và các thành phần cơ bản

Định nghĩa về Lifelong machine learning được Thrun đề cập vào năm 1996 như sau: Một hệ thống đã thực hiện việc học qua các nhiệm vụ từ 1 đến N. Khi đối mặt với việc học nhiệm vụ thứ N + 1, hệ thống sẽ sử dụng tri thức đã được tích lũy từ việc học N nhiệm vụ trước đó để phục vụ cho việc học nhiệm vụ thứ N + 1.

Dựa vào định nghĩa đó, chúng ta có thể nhận ra được ba đặc điểm chính của phương pháp Lifelong machine learning: (1) học một cách liên tục, (2) các tri thức được khai phá từ các nhiệm vụ trong quá khứ sẽ được lưu trữ trong cơ sở tri thức (Knowledge Base), (3) khả năng sử dụng những tri thức đã được tích lũy đó để giúp đỡ các nhiệm vụ cần học trong tương lai. Từ đó, phương pháp Lifelong machine learning sẽ học một chuỗi các nhiệm vụ, có thể sẽ không bao giờ ngừng lại quá trình học và từ đó hệ thống sẽ trở nên ngày càng có nhiều tri thức, giúp việc học trở nên tốt hơn. Những đặc điểm đó khiến cho phương pháp Lifelong machine learning trở nên khác biệt với những phương pháp có vẻ tương đồng như Transfer learning hay Multi-task learning khi nhưng phương pháp đó không có một hay những đặc điểm trên. [1]



Hình 2.2 Kiến trúc một hệ thống Lifelong machine learning [1]

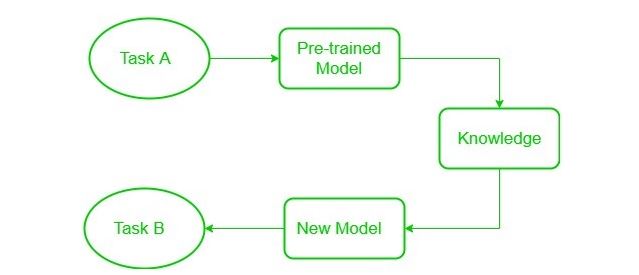
Một hệ thống Lifelong machine learning thường có bốn thành phần chính sau:

1. **Nơi lưu trữ thông tin quá khứ (Past Information Store: PIS):** Đây là nơi lưu trữ những thông tin có được từ những nhiệm vụ học trước đó. Nơi này có thể có nhiều phần nhỏ để lưu trữ những thông tin như dữ liệu gốc, các kết quả trung gian hay các mô hình đã được xây dựng trong quá khứ. [3]
2. **Cơ sở tri thức (Knowledge Base: KB):** Đây là nơi lưu trữ tri thức tích lũy được từ việc khai phá trong PIS. Trong một hệ thống cụ thể, người dùng cần quyết định một cách biểu diễn tri thức sao cho phù hợp với bài toán cần dùng. [3]
3. **Bộ khai phá tri thức (Knowledge Miner: KM)**: Đây là nơi sẽ khai phá tri thức trong PIS và những tri thức có được sẽ được chuyển tới lưu trữ trong KB. [3]
4. **Bộ học dựa trên cơ sở tri thức (Knowledge-Based Learner: KBL):** Dựa vào những tri thức tích lũy được trong KB và những thông tin trong PIS, bộ học này tiến hành việc học những nhiệm vụ mới. [3]

Từ kiến trúc được mô tả như hình Hình 2.2, ta có thể phát biểu cách thức hoạt động của phương pháp Lifelong machine learning như sau: Khi gặp một nhiệm vụ mới TN+1 với bộ dữ liệu DN+1, Task Manager (TM) sẽ đưa nhiệm vụ đó tới Knowledge-based Learner (KBL) . KBL sẽ thực hiện việc học nhiệm vụ mới TN+1 với sự trợ giúp từ tri thức đã được tích lũy từ các nhiệm vụ T1 đến TN trước đó trong Knowledge Base (KB) sẽ tạo đầu ra là một Model để sử dụng cho các ứng dụng thực tế. Đồng thời các tri thức mới vừa được học sẽ được chuyển tới KB để phục vụ cho các nhiệm vụ khác sau này. Ứng dụng được áp dụng Model đó có thể tìm ra các nhiệm vụ mới và đồng thời tiếp tục cung cấp những tri thức được khai phá cho KB để nâng cao chất lượng Model.

#### Các phương pháp học máy tương tự khác

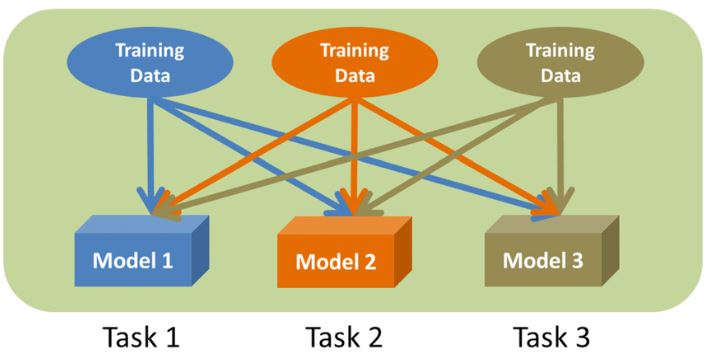
Như đã đề cập phía trên, một hệ thống Lifelong machine learning gồm ba đặc điểm chính là khả năng học liên tục, tri thức được tích lũy, khả năng áp dụng những tri thức đó cho các nhiệm vụ trong tương lai. Hiện nay có khá nhiều những phương pháp học máy khác có những đặc điểm tương tự như vậy. Sau đây em xin trình bày về hai phương pháp có nét tương đồng nhất là Transfer learning và Multi-task learning.



Hình 2.3 Sơ đồ hệ thống Transfer learning [4]

Transfer learning là một chủ đề nổi bật trong lĩnh vực học máy và khai phá dữ liệu trong những năm gần đây. Phương pháp này còn được biết đến như là thích ứng miền (Domain adaptation) trong lĩnh vực xử lý ngôn ngữ tự nhiên. [1]

Phương pháp Transfer learning sử dụng source domain đã được gán nhãn để phục vụ cho quá trình học target domain. Không giống như Lifelong machine learning, Transfer learning thì học không liên tục và không có sự lưu giữ tri thức đã được học. Không chỉ vậy, source domain phải có sự tương tự như target domain thì mới áp dụng được Transfer learning. Luồng hoạt động chỉ là một chiều vì chỉ có source domain hỗ trợ cho target domain và không có chiều ngược lại. [5]



Hình 2.4 Sơ đồ hệ thống Multi-task learning [6]

Multi-task learning là phương pháp có thể học nhiều nhiệm vụ cùng một thời điểm. Mặc dù phương pháp này có khả năng học liên tục nhưng lại không có sự tích lũy tri thức trong quá trình học. Điều này gây ra sự khó khăn khi gặp phải nhiệm vụ mới nếu số lượng nhiệm vụ là rất lớn vì không có tri thức tích lũy được. [5]

Với những so sánh đó, chúng ta có thể nhận thấy rằng phương pháp Lifelong machine learning mang những đặc điểm tốt hơn so với hai phương pháp kể trên. Hiện này còn có rất nhiều những phương pháp khác mang tính chất tương tự nhưng để phù hợp với nội dung của đồ án, em xin không trình bày thêm về phần này.

#### Phương pháp đánh giá hệ thống

Khi tiến hành đánh giá những phương pháp học máy truyền thống, chúng ta đánh giá dựa vào quá trình huấn luyện và kiểm thử dữ liệu được lấy ở cùng một miền. Với Lifelong machine learning, chúng ta cần sử dụng một phương pháp đánh giá khác bởi nó thực hiện việc hoc trên nhiều miền khác nhau và cần phải đánh giá kết quả của cách học đó. Thông thường các bước đánh giá phương pháp Lifelong machine learning được thực hiện như sau:

1. ***Chạy trên dữ liệu từ những nhiệm vụ trong quá khứ:*** Đầu tiên chúng ta tiến hành chạy giải thuật học máy trên tập dữ liệu từ một tập các nhiệm vụ đã được học từ trước và tri thức thu được sẽ được lưu trữ trong KB. Tùy theo từng giải thuật khác nhau chúng ta sẽ có các phương pháp lưu trữ tri thức khác nhau. [1]
2. ***Chạy trên dữ liệu từ nhiệm vụ mới:*** Tiếp theo chúng ta tiến hành chạy giải thuật học máy trên dữ liệu từ nhiệm vụ mới bằng cách sử dụng tri thức tích lũy được trong KB. [1]
3. ***Chạy giải thuật thực nghiệm:*** Để tiến hành so sánh, chúng ta sẽ tiến hành chạy một vài giải thuật thực nghiệm. Thông thường sẽ có hai loại thực nghiệm. Loại đầu tiên là sử dụng thuật toán theo phương pháp truyền thống trên dữ liệu mới mà không áp dụng những tri thức tích lũy được trong quá khứ. Thực nghiệm thứ hai là sử dụng giải thuật Lifelong machine learning mà chúng ta đã tạo ra. [1]
4. ***Phân tích kết quả:*** Chúng ta tiến hành so sánh kết quả các thực nghiệm. [1]

#### Những khó khăn của phương pháp

Lifelong machine learning là phương pháp mang lại nhiều thách thức như cần phải lựa chọn tri thức nào phù hợp để giữ lại, làm sao để sử dụng những tri thức đã tích lũy được cũng như làm sao để duy trì KB. Sau đầy là 2 thách thức tiềm ẩn nhưng rất quan trọng trong việc sử dụng phương pháp này:

* *Tính chính xác của tri thức*: Việc sử dụng tri thức không chính xác sẽ gây ra những hậu quả nghiêm trọng cho việc học các nhiệm vụ mới. Những tri thức được tích lũy không chính xác từ nhiệm vụ mới sẽ càng khiến cho hệ thống gặp nhiều lỗi hơn. [1]
* Khả năng ứng dụng tri thức: Có thể một phần của tri thức áp dụng chính xác cho một vài nhiệm vụ trong quá khứ nhưng không có nghĩa nó sẽ tương thích với các nhiệm vụ trong tương lai. Nếu nhiệm vụ mới được áp dụng những tri thức không phù hợp thì cũng sẽ gây ra hậu quả như trường hợp kể trên. [1]

## Giải thuật phân loại Multinomial Naive Bayes (MNB)

Naive Bayes là một phương pháp đơn giản phân loại dựa vào xác suất. Phương pháp này được sử dụng rộng rãi trong lĩnh vực học máy, phổ biến trong nhiều lĩnh vực như các công cụ tìm kiếm, các bộ lọc mail nói riêng và phân loại văn bản nói chung. Ý tưởng cơ bản của cách tiếp cận Naive Bayes là sử dụng xác suất có điều kiện giữa từ và chủ đề để dự đoán xác suất chủ đề của một văn bản cần phân loại. Ưu điểm của giải thuật này là dễ cài đặt và thời gian huấn luyện mô hình nhanh.

Giải thuật này sử dụng tới định lý Bayes nên tiếp theo đây sẽ nêu ra ngắn gọn công thức của định lý Bayes. Công thức đó được phát biểu như sau:

|  |  |
| --- | --- |
|  | (2‑1) |

Trong đó:

* P(Y|X): Xác suất có điều kiện sự kiện Y xảy ra nếu biết rằng sự kiện X đã xảy ra. [7]
* P(X|Y): Xác suất có điều kiện sự kiện X xảy ra nếu biết rằng sự kiện Y đã xảy ra. [7]
* P(Y): Xác suất sự kiện Y xảy ra. [7]
* P(X): Xác suất sự kiện X xảy ra. [7]

Xét bài toán phân lớp có C lớp 1,2,…,C. Giả sử có một điểm dữ liệu x được biểu diễn dưới dạng một vector có d chiều (x Rd). Khi đó xác suất để điểm dữ liệu này rơi vào lớp c được tính bằng: [8]

|  |  |
| --- | --- |
|  | (2‑2) |

Biểu thức này sẽ giúp chúng ta xác định được xác suất để điểm dữ liệu rơi vào mỗi class. Từ đó có thể giúp xác định class của điểm dữ liệu đó bằng cách chọn ra class có xác suất cao nhất: [8]

|  |  |
| --- | --- |
|  | (2‑3) |

Biểu thức (2‑3) thường khó tính được trực tiếp, thay vào đó quy tắc Bayes (2‑1) thường được sử dụng nên công thức này sẽ được thay bằng: [8]

|  |  |
| --- | --- |
|  | (2‑4) |

Vì mẫu số P(x) trong công thức (2‑4) không phụ thuộc vào c nên ta có thể rút gọn công thức đó thành: [8]

|  |  |
| --- | --- |
|  | (2‑5) |

Lúc này, chúng ta cần tính hai giá trị là P(x|c) và P(c). Với P(c) chúng ta có thể tính theo công thức:

|  |  |
| --- | --- |
|  | (2‑6) |

Trong đó Nc là số lượng dữ liệu trong tập dữ liệu huấn luyện có nhãn là c và N là tổng số dữ liệu trong tập dữ liệu huấn luyện.

Với P(x|c) tức phân phối của các điểm dữ liệu trong class c thường rất khó tính toán vì x là một biến ngẫu nhiên nhiều chiều, cần rất rất nhiều dữ liệu training để có thể xây dựng được phân phối đó. Để giúp cho việc tính toán được đơn giản, người ta thường giả sử một cách đơn giản nhất rằng các thành phần của biến ngẫu nhiên x là [độc lập với nhau](https://machinelearningcoban.com/2017/07/09/prob/#-independence), nếu biết c. Tức là: [8]

|  |  |
| --- | --- |
|  | (2‑7) |

Sau khi xác định được P(c) và P(x|c) thì lúc này công thức (2‑4) sẽ được phát biểu thành:

|  |  |
| --- | --- |
|  | (2‑8) |

Lúc này chúng ta cần xác định P(xi|c) để tìm ra điểm dữ liệu đầu vào sẽ thuộc vào lớp nào. Việc tính toán P(xi|c) phụ thuộc vào kiểu dữ liệu và có ba loại được sử dụng phổ biến là Gaussian Naive Bayes, Multinomial Naive Bayes và Bernoulli Naive Bayes. Mô hình Gaussian Naive Bayes được sử dụng chủ yếu trong loại dữ liệu mà các thành phần là các biến liên tục, Multinomial Naive Bayes được sử dụng chủ yếu trong phân loại văn bản còn Bernoulli Naive Bayes được áp dụng cho các dữ liệu mà mỗi thành phần là một giá trị binary (bằng 0 hoặc bằng 1). [8] Do đó để phù hợp với nội dung của đồ án là phân loại những bài đăng mạng xã hội thì em sử dụng Multinomail Naive Bayes. Khi đó P(xi|c) sẽ được tính như sau:

|  |  |
| --- | --- |
|  | (2‑9) |

Trong đó Nci là tổng số lần xuất hiện đặc trưng i trong các văn bản của lớp c và Nc là tổng số từ xuất hiện trong lớp c. Cách tính này có một hạn chế là nếu gặp một từ chưa bao giờ xuất hiện trong lớp c bất kì dẫn đến P(xi|c) = 0. Kéo theo kết quả cần tìm của biểu thức (2‑8) sẽ luôn có giá trị bằng 0, điều này làm cho kết quả là không chính xác. Để giải quyết trường hợp này một kĩ thuật được gọi là Laplace smoothing được áp dụng như sau:

|  |  |
| --- | --- |
|  | (2‑10) |

Trong đó k là hằng số được thêm vào để tránh xác suất bằng 0 và thường được gán bằng 1. |V| là số lượng từ vựng khác nhau có trong từ điển.

Lúc này vẫn còn một vấn đề với biểu thức (2‑8) khi d lớn thì vế phải của biểu thức sẽ là một số rất nhỏ và gần với giá trị 0 khiến cho việc tính toán sẽ gặp sai số. Để giải quyết vấn đề này thì biểu thức được tính lại bằng cách lấy log của vế phải, điều này không gây ảnh hưởng tới kết quả bởi log là một hàm đồng biến trên tập các số dương. Lúc này biểu thức sẽ được phát biểu thành: [8]

|  |  |
| --- | --- |
|  | (2‑11) |

## Lý thuyết về N-Grams

N-Grams được hiểu là các chuỗi liền kề của N từ trong một văn bản nhất định. N-grams được sử dụng trong rất nhiều lĩnh vực khác nhau và là một phương pháp đơn giản mà hiệu quả trong NLP.

Xét một ví dụ cụ thể nếu chúng ta có một câu “The quick brown fox jumps over the lazy dog”. Lúc này với N =1, chúng ta sẽ có các từ đứng độc lập với nhau và mỗi từ đó sẽ được gọi là unigram : "The", "quick", "brown", "fox", "jumps", "over", "the", "lazy", "dog". [9]

Với N = 2, chúng ta sẽ có các chuỗi bao gồm 2 từ đứng cạnh nhau, mỗi chuỗi như vậy được gọi là bigram: "The quick", "quick brown", "brown fox", "fox jumps", "jumps over", "over the", "the lazy", "lazy dog". [9]

Với N = 3, chúng ta sẽ có các chuỗi bao gồm 3 từ đứng cạnh nhau, mỗi chuỗi như vậy được gọi là trigram: "The quick brown", "quick brown fox", "brown fox jumps", "fox jumps over", "jumps over the", "over the lazy", "the lazy dog". [9]

Tương tự như vậy chúng ta có thể xét đến số lượng N tùy theo nhu cầu sử dụng cho các bài toán khác nhau. Tuy nhiên nếu N càng lớn thì sẽ càng phức tạp cho việc xử lý.

## Phương pháp trích chọn đặc trưng Information Gain (IG)

Trong thời điểm hiện nay dữ liệu mà chúng ta thu thập thường ở dạng thô và đến từ nhiều nguồn khác nhau trên mạng internet. Khi đó thì dữ liệu chỉ có một số lượng đặc trưng nhất định là có ích cho bài toán còn phần lớn là thông tin nhiễu gây ra những ảnh hưởng không tốt tới bài toán. Dữ liệu này cần phải được xử lý để chọn ra những đặc trưng tốt thì những phương pháp học máy chúng ta xây dựng mới đạt được hiệu quả cao. Với dữ liệu là văn bản như trong đồ án, có thể tồn tại nhiều dạng khác nhau như chữ cái thường, in hoa, dấu câu hay những kí tự đặc biệt.

Có rất nhiều những phương pháp trích chọn đặc trưng như Bag-of-words, TF-IDF, Word2vec, Information Gain hay Skip-gram được sử dụng trong xử lý văn bản. Với phạm vi của đồ án, bài toán phát hiện ý định trong văng bản thường sẽ chỉ có một hoặc hai câu thể hiện ý định còn phần lớn sẽ không liên quan trực tiếp đến việc thể hiện ý định. Bởi vậy trích chọn đặc trưng là rất quan trọng với đồ án này để giảm đi những thông tin nhiễu. Để phù hợp với phạm vi nhỏ hẹp của đồ án này, em sẽ sử dụng phương pháp trích chọn đặc trưng IG để tìm ra độ lợi thông tin.

Với mỗi đặc trưng f thì độ lợi thông tin sẽ được tính như sau: [10]

|  |  |
| --- | --- |
|  | (2‑12) |

Nếu giá trị IG(f) càng lớn thì càng chứng tỏ đặc trưng f càng mang nhiều thông tin.

Trong đó không phụ thuộc vào đặc trưng f nên công thức (2‑12) sẽ được rút ngọn như sau: [10]

|  |  |
| --- | --- |
|  | (2‑13) |

Với bài toán chỉ có hai nhãn là positive (+) và negative (-), công thức (2‑13) sẽ được biến đổi thành: [10]

|  |  |
| --- | --- |
|  | (2‑14) |

Trong đó:

* : Xác suất của một mẫu chứa đặc trưng f. [10]
* : Xác suất của một mẫu không chứa đặc trưng f. [10]
* và : Xác suất của một mẫu mà có nhãn là positive (+) và negative (-) khi có chứa đặc trưng f. [10]
* và : Xác suất của một mẫu mà có nhãn là positive (+) và negative (-) khi không chứa đặc trưng f. [10]

Để dễ dàng cho việc thực hiện tính toán, em gọi một số giá trị như sau:

* Df : Số lượng mẫu chứa đặc trưng f trong bộ dữ liệu huấn luyện
* D: Tổng số mẫu của tập giá trị huấn luyện
* và : Số lượng mẫu có nhãn là positive (+) và negative (-) mà chứa đặc trưng f trong tập dữ liệu huấn luyện
* và : Số lượng mẫu có nhãn là positive (+) và negative (-) mà không chứa đặc trưng f trong tập dữ liệu huấn luyện

Khi đó ta sẽ có những công thức sau:

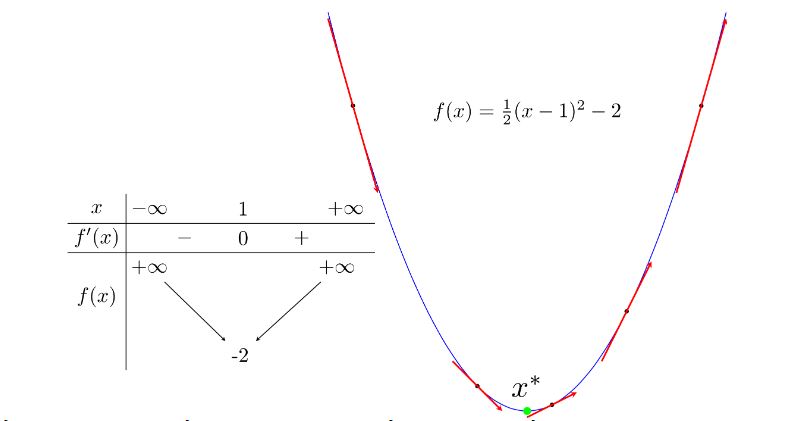
|  |  |
| --- | --- |
|  | (2‑15) |
|  | (2‑16) |
|  | (2‑17) |
|  | (2‑18) |
|  | (2‑19) |
|  | (2‑20) | |

Lúc này khi đã tính được IG cho từng đặc trưng của bộ dữ liệu huấn luyện, ta tiến hành sắp xếp các giá trị IG đó theo thứ tự giảm dần và chọn những đặc trưng có IG cao nhất mà chúng ta cần thiết.

## Phương pháp Stochastic gradient descent (SGD)

Trong học máy nói riêng và toán tối ưu nói chung, chúng ta thường xuyên phải tìm giá trị nhỏ nhất (hoặc đôi khi là lớn nhất) của một hàm số nào đó. Nhìn chung, việc tìm giá trị nhỏ nhất (global minimum) của các hàm mất mát trong học máy là rất phức tạp nên thay vào đó người ta thường cố gắng tìm các điểm cực tiểu địa phương (local minimum) và ở một mức độ nào đó, coi đó là nghiệm cần tìm của bài toán. [11] Khi cực tiểu địa phương chính là giá trị nhỏ nhất của hàm số thì đó là trường hợp tốt nhất mà chúng ta hướng đến.

Các điểm cực tiểu địa phương là nghiệm của phương trình đạo hàm bằng 0. Nếu bằng một cách nào đó có thể tìm được toàn bộ (hữu hạn) các điểm cực tiểu, ta chỉ cần thay từng điểm cực tiểu địa phương đó vào hàm số rồi tìm điểm làm cho hàm có giá trị nhỏ nhất. Tuy nhiên, trong hầu hết các trường hợp, việc giải phương trình đạo hàm bằng 0 là bất khả thi. Nguyên nhân có thể đến từ sự phức tạp của dạng của đạo hàm, từ việc các điểm dữ liệu có số chiều lớn, hoặc từ việc có quá nhiều điểm dữ liệu. [11]



Hình 2.5 Minh họa về điểm cực tiểu [11]

Hướng tiếp cận phổ biến nhất là xuất phát từ một điểm mà chúng ta coi là gần với nghiệm của bài toán, sau đó dùng một phép toán lặp để tiến dần đến điểm cần tìm, tức đến khi đạo hàm gần với 0. Gradient Descent (GD) và các biến thể của nó là một trong những phương pháp được dùng nhiều nhất. [11]

Trước tiên chúng ta sẽ xem xét trường hợp hàm số chỉ phụ thuộc vào một biến duy nhất với quan sát hình Hình 2.5. Giả sử xt là điểm ta tìm được sau vòng lặp thứ t. Ta cần tìm một thuật toán để đưa xt về càng gần x\* càng tốt. [11]

Chúng ta có thể quan sát được như sau:

Nếu đạo hàm của hàm số tại xt: f′(xt) > 0 thì xt nằm về bên phải so với x\* (và ngược lại). Để điểm tiếp theo xt+1 gần với x\* hơn, chúng ta cần di chuyển xt về phía bên trái, tức về phía âm. Nói các khác, **chúng ta cần di chuyển ngược dấu với đạo hàm**: [11]

|  |  |
| --- | --- |
|  | (2‑21) |

Trong đó là một đại lượng ngược dấu với đạo hàm f′(xt). [11]

Nếu xt càng xa x\* về phía bên phải thì f′(xt) càng lớn hơn 0 (và ngược lại). Vậy, lượng di chuyển Δ, một cách trực quan nhất, là tỉ lệ thuận với −f′(xt). [11]

Với những nhận xét đó chúng ta có thể viết lại công thức (2‑21) như sau:

|  |  |
| --- | --- |
|  | (2‑22) |

Trong đó  là một số dương được gọi là tốc độ học (learning rate). Dấu trừ thể hiện việc chúng ta phải đi ngược với đạo hàm (Đây cũng chính là lý do phương pháp này được gọi là Gradient Descent - descent nghĩa là đi ngược). Các quan sát đơn giản phía trên, mặc dù không phải đúng cho tất cả các bài toán, là nền tảng cho rất nhiều phương pháp tối ưu nói chung và thuật toán Machine Learning nói riêng. [11]

Việc lựa chọn tốc độ học là rất quan trọng trong các bài toán thực tế. Việc lựa chọn giá trị này phụ thuộc vào từng bài toán cụ thể và phải làm một vài thí nghiệm để chọn ra giá trị tốt nhất. Nếu tốc độ học quá nhỏ thì tốc độ hội tụ sẽ chậm, ảnh hưởng tới tốc độ của phương pháp học máy và thậm chí không thể tới được đích. Ngược lại nếu tốc độ học quá lớn thì có thể thuật toán sẽ nhanh hơn nhưng cũng có thể không hội tụ được nếu bước nhảy là quá lớn gây ra sự quẩn quanh ở đích. [11]

Với trường hợp sử dụng GD cho hàm nhiều biến, chúng ta có thể tổng quát như sau đây. Giả sử ta cần tìm cực tiểu địa phương cho hàm f() trong đó (theta) là một vector, thường được dùng để ký hiệu tập hợp các tham số của một mô hình cần tối ưu. Đạo hàm của hàm số đó tại một điểm  bất kỳ được ký hiệu là . Tương tự như hàm một biến, thuật toán GD cho hàm nhiều biến cũng bắt đầu bằng một điểm dự đoán , sau đó, ở vòng lặp thứ t, quy tắc cập nhật là: [11]

|  |  |
| --- | --- |
|  | (2‑23) |

Trong đó là đạo hàm của hàm số tại .

Thuật toán GD có rất nhiều biến thể giúp gia tăng tốc độ chạy hay sự hội tụ của thuật toán của thuật toán như Batch Gradient Descent, Mini-batch Gradient Descent hay Stochastic gradient descent. Ở đây em sử dụng Stochastic gradient descent (SGD) vì sự đơn giản áp dụng cho đồ án.

Khác với 2 thuật toán biến thể của GD nêu ở trên thì SGD chỉ thực hiện tính đạo hàm của hàm mất mát dựa trên chỉ một điểm dữ liệu xi rồi cập nhật θ dựa trên đạo hàm này. Việc này được thực hiện với từng điểm trên toàn bộ dữ liệu, sau đó lặp lại quá trình trên. Thuật toán rất đơn giản này trên thực tế lại làm việc rất hiệu quả. [11]

Mỗi lần duyệt một lượt qua tất cả các điểm trên toàn bộ dữ liệu được gọi là một epoch. Với GD thông thường thì mỗi epoch ứng với 1 lần cập nhật θ còn với SGD thì mỗi epoch ứng với N lần cập nhật θ với N là số điểm dữ liệu. Nhìn vào một mặt, việc cập nhật từng điểm một như thế này có thể làm giảm đi tốc độ thực hiện 1 epoch. Nhưng nhìn vào một mặt khác, SGD chỉ yêu cầu một lượng epoch rất nhỏ (thường là 10 cho lần đầu tiên, sau đó khi có dữ liệu mới thì chỉ cần chạy dưới một epoch là đã có nghiệm tốt). Vì vậy SGD phù hợp với các bài toán có lượng cơ sở dữ liệu lớn và cần tốc độ cao. [11]

Một điểm cần lưu ý đó là: sau mỗi epoch, chúng ta cần shuffle (xáo trộn) thứ tự của các dữ liệu để đảm bảo tính ngẫu nhiên. Việc này cũng ảnh hưởng tới hiệu năng của SGD. [11]

Một cách toán học, quy tắc cập nhật của SGD là:

|  |  |
| --- | --- |
|  | (2‑24) |

Trong đó là đạo hàm với chỉ 1 điểm dữ liệu (input,label) là (xi;yi). [11]

Điều kiện dừng của SGD có thể được áp dụng theo nhiều cách như giới hạn số vòng lặp, so sánh giá trị đạo hàm giữa hai lần liên tiếp nếu độ chênh lệch ở một ngưỡng chấp nhận được hay so sánh giá trị hàm mất mát giữa hai lần liên tiếp cũng tương tự như vậy.

## Các thang đo đánh giá mô hình học máy

Khi chúng ta đã xây dựng một mô hình học máy và huấn luyện nó trên một tập dữ liệu, điều tiếp theo cần làm là đánh giá hiệu năng của mô hình trên tập dữ liệu mới. Việc đánh giá mô hình giúp chúng ta trả lời những câu hỏi như liệu mô hình đã được huấn luyện thành công hay chưa và mức độ thành công của mô hình tốt đến đâu. [12]

Sau đây em xin trình bày về 4 thang đo được sử dụng rỗng rãi trong lĩnh vực học máy cụ thể là trong bài toán phân lớp là Accuracy, Precision, Recall và F1.

Khi thực hiện bài toán phân loại, giữa dự đoán và thực tế có thể xảy ra 4 trường hợp:

* True Positive (TP):  Dự đoán là positive và kết quả cũng là positive (dự đoán đúng).
* True Negative (TN): Dự đoán là negative và kết quả cũng là negative (dự đoán đúng)
* False Positive (FP): Dự đoán là positive nhưng kết quả là negative (dự đoán sai).
* False Negative (FN): Dự đoán là negative nhưng kết quả là positive (dự đoán sai).

Thang đo Accuracy được định nghĩa là tỷ lệ phần trăm dự đoán đúng cho dữ liệu thử nghiệm. Nó có thể được tính bằng cách chia tổng số dự đoán chính xác cho tổng số dự đoán đã thực hiện. [12]

|  |  |
| --- | --- |
|  | (2‑25) |

Tuy nhiên thực tế không phải lúc nào thang đo accuracy cũng có thể đại diện cho việc mô hình có chính xác hay không. Xét một trường hợp cụ thể như dự đoán xem ai đó có mắc bệnh hay không. Giả sử, số người mắc bệnh là 1 người, 99 người không mắc bệnh và công cụ dự đoán đưa ra kết quả là tất cả mọi người đều không mắc bệnh. Vậy công cụ vẫn đạt tỷ lệ chính xác là 99%. Tuy nhiên, chúng ta sẽ không tìm được ai là người mắc bệnh. Vậy mô hình không có giá trị với trường hợp này. Ta có thể kết luận thang đo accuracy không áp dụng được cho dữ liệu có phân phối không đều. Từ đó chúng ta có 2 thang đo là precision và recall.

Thang đo Precision thể hiện tối thiểu hóa với FP. Một trường hợp cụ thể có thể sử dụng thang đo này là phân loại spam email, nều FP lớn thì sẽ khiến mail quan trọng bị rơi vào mục spam, điều này là không chấp nhận được. Nếu FN là lớn thì vẫn có thể chấp nhận được vì không lỡ mail quan trọng. Tỉ lệ này được tính bằng số dự đoán positive là đúng (TP) chia cho tổng số positive mà ta dự đoán (TP+FP)

|  |  |
| --- | --- |
|  | (2‑26) |

Thang đo Recall thể hiện sự tối thiểu hóa với FN. Một trường hợp cụ thể có thể sử dụng thang đo này là xác định nhiễm bệnh, nều FP lớn thì có thể chấp nhận được vì không gây nguy hiểm đến tính mạng. Nếu FN là lớn thì không chấp nhận được vì gây ảnh hưởng đến tính mạng. Tỉ lệ này được tính bằng số dự đoán positive là đúng (TP) chia cho tổng số positive trong thực tế (TP+FN)

|  |  |
| --- | --- |
|  | (2‑27) |

Chúng ta có thể thấy Precision và Recall là 2 giá trị đối ngược với nhau. Precision sẽ coi trọng giá trị FP hơn còn Recall coi trọng giá trị FN hơn. Để hài hòa được 2 giá trị này chúng ta có công thức tính F1 kết hợp giữa Precision và Recall được thể hiện như sau:

|  |  |
| --- | --- |
|  | (2‑28) |

Cả 4 giá trị này đều có thể sử dụng trong những trường hợp cụ thể nhưng thông thường trong mô hình học máy chúng ta thường sử dụng thang đo F1 bởi nó thể hiện được sự hài hòa giữa Recall và Precision, không thiên vị giá trị nào cả cũng như có thể sử dụng cho cả những bài toán có phân phối dữ liệu đều và không đều.

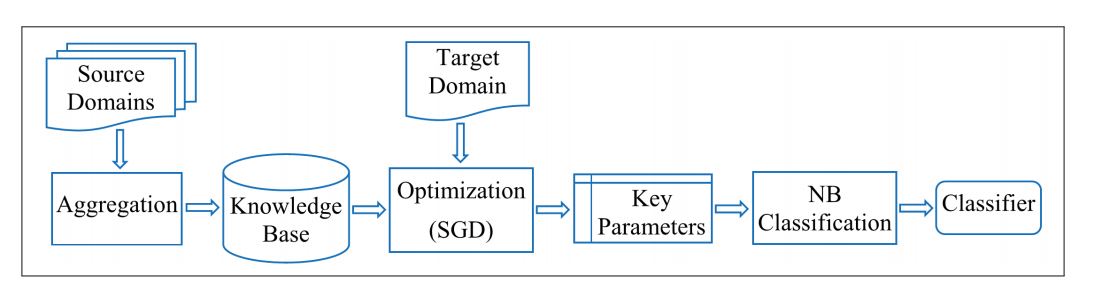
Thang đo F1 thường có 2 tính chất:

* F1 sẽ có xu hướng nhận giá trị gần với giá trị nào nhỏ hơn giữa precision và recall. [12]
* F1 sẽ có giá trị lớn nếu cả 2 giá trị precision và recall đều lớn. [12]

## Cách áp dụng Lifelong machine learning cho đồ án

### Tổng quan về cách áp dụng

Với bài toán phát hiện ý định mua bán được giới thiệu chung tại chương 1, để phù hợp với mục tiêu để ra thì đồ án này sẽ cung cấp phương pháp Lifelong machine learning áp dụng cùng giải thuật phân loại Naive Bayes với mục đích tìm ra những bài đăng có mang ý định mua bán hay không. Với ý tưởng sử dụng những tri thức đã được tích lũy trong quá khứ để giúp đỡ cho các nhiệm vụ học sau này của phương pháp Lifelong machine learning, các bước áp dụng cho bài toán sẽ được trình bày như sau:



Hình 2.6 Sơ đồ áp dụng phương pháp Lifelong machine learning cho bài toán [10]

Phương pháp này được chia làm ba phần chính:

* **Trộn dữ liệu:** Trong phần này sẽ tiến hành trộn dữ liệu và trích xuất tri thức từ nhiều source domain rồi lưu trữ trong KB. [10]
* **Tối ưu hóa:** Phần này sẽ sử dụng tri thức tích lũy được trong KB để tối ưu hóa những tham số sẽ sử dụng trong mô hình phân loại. [10]
* **Phân loại:** Phần này sẽ sử dụng những tham số đã được tối ưu để xây dựng giải thuật phân loại Naive Bayes. [10]

Từ sơ đồ áp dụng hình Hình 2.6, chúng ta có thể phát biểu cụ thể các bước tiến hành như sau: Với các dữ liệu từ các source domain, chúng ta tiến hành trộn các dữ liệu đó và thực hiện việc trích chọn tri thức để lưu vào KB. Bước tiếp theo từ dữ liệu target domain của nhiệm vụ học mới, ta kết hợp cùng những tri thức đã được tích lũy trong KB rồi thực hiện tối ưu hóa sử dụng Stochastic Gradient Descent (SGD). Chúng ta sử dụng SGD bởi khi tiến hành trộn dữ liệu có thể xảy ra việc nhiễu dữ liệu. Tiếp đó chúng ta tiến hành chọn ra các đặc trưng quan trọng sau khi đã tối ưu bằng SGD để phục vụ cho việc phân loại sử dụng giải thuật Naive Bayes tạo ra mô hình.

### Chi tiết về cách áp dụng

#### Các thành phần trong bài toán

Dựa vào những thông tin đã được đề cập trong phần 2.2.2.1, các thành phần để áp dụng phương pháp Lifelong machine learning cho đồ án sẽ được phát biểu như sau:

* **PIS**: Với từng nhiệm vụ đã học trong quá khứ, chúng ta tiến hành lưu trữ 2 phần:

1. và cho mỗi từ w. [3]
2. Số lần xuất hiện của từ w trong các dữ liệu positive (+) và số lần xuất hiện của từ w trong các dữ liệu negative (-) . trong mỗi nhiệm vụ . [3]

* **KB**: Cơ sở tri thức sẽ lưu trữ số lần từ w xuất hiện trong các dữ liệu mang nhãn positive (+) và negative (-) của tất cả các nhiệm vụ trước đó: [3]

|  |  |  |
| --- | --- | --- |
|  | (2‑29) | |
|  | | (2‑30) |

* **KM:** Bộ khai phá tri thức ở đây sẽ tiến hành tổng hợp số lượng những từ w được đề cập phía trên theo 2 công thức (2‑29), (2‑30) và được lưu trữ trong KB. [3]
* **KBL**: Bộ học dựa trên cơ sở tri thức sẽ tiến hành thực hiện việc học những nhiệm vụ tiếp theo nhờ những tri thức và thông tin thu được. [3]

#### Thực hiện tối ưu hóa hàm mục tiêu

Với bài toán phân loại trong học máy, chúng ta đều mong muốn hướng tới P(cj|di) = 1 với cj là nhãn được gán chính xác cho dữ liệu di và P(cf|di) = 0 với cj là những nhãn không chính xác cho dữ liệu di. Với bài toán phân loại ý định mua bán sẽ được trình bày chi tiết ở phía sau đồ án thì chúng ta chỉ quan tâm đến việc bài đăng có ý định mua bán hay không. Nếu dữ liệu di là có ý định mua bán thì chúng ta sẽ mong muốn P(+|di) sẽ đạt giá trị cao nhất có thể và P(-|di) sẽ có giá trị thấp nhất có thể, phân loại cho nhãn không có ý định sẽ ngược lại. [10]

Với dữ liệu di có ý định mua bán, hàm mục tiêu sẽ được xác định như sau: [10]

|  |  |
| --- | --- |
|  | (2‑31) |

Trong đó:

* P(+|di) : Xác suất để dữ liệu di mang nhãn dương
* P(-|di) : Xác suất để dữ liệu di mang nhãn âm

Để tối ưu hóa chúng ta sẽ mong muốn hàm mục tiêu sẽ đạt giá trị lớn nhất nhưng các thư viện trong học máy thường sẽ chỉ hỗ trợ tìm giá trị nhỏ nhất. Khi đó chúng ta cần cực tiểu hóa hàm sau: [10]

|  |  |
| --- | --- |
|  | (2‑32) |

Trong đó:

* : Biểu thị số lần mà từ w xuất hiện trong tài liệu di. [10]
* P(+) : Xác suất dữ liệu mang nhãn dương. [10]
* P(-): Xác suất dữ liệu mang nhãn âm. [10]
* P(w|+) : Xác suất một từ w xuất hiện trong dữ liệu có nhãn dương. [10]
* P(w|-) : Xác suất một từ w xuất hiện trong dữ liệu có nhãn âm. [10]

Dựa vào phương pháp NB đã được trình bày ở phần 2.3 , ta có được 2 công thức sau:

|  |  |
| --- | --- |
|  | (2‑33) |
|  | (2‑34) |

Trong đó:

* : Hệ số làm trơn và được gán bằng 1.
* : Số lần xuất hiện của w trong nhãn (+) ở thời điểm hiện tại sau khi đã được cập nhật với điểm khởi tạo là . [3]
* : Số lần xuất hiện của w trong nhãn (-) ở thời điểm hiện tại sau khi đã được cập nhật với điểm khởi tạo là . [3]

Thay 2 công thức này vào công thức (2‑32) ta thu được công thức sau:

|  |  |
| --- | --- |
|  | (2‑35) |

Khi đó đạo hàm và sẽ được tính như sau:

|  |  |
| --- | --- |
|  | (2‑36) |
|  | (2‑37) | |

Với quy tắc cập nhật SGD được trình bày trong phần 2.6 có tác dụng để cực tiểu hàm mất mát (2‑31) được trình bày như sau:

|  |  |
| --- | --- |
|  | (2‑38) |
|  | (2‑39) | |

Trong đó là tốc độ học như đã trình bày ở phần 2.6.

Thực hiện tương tự với dữ liệu di không có ý định mua bán để hàm mục tiêu đạt giá trị lớn nhất chúng ta cần cực tiểu hóa hàm sau:

|  |  |
| --- | --- |
|  | (2‑40) |

Tiến hành tương tự như trên ta cũng thu được hàm và được tính như sau:

|  |  |
| --- | --- |
|  | (2‑41) |
|  | (2‑42) | |

Với các quy tắc cập nhật dựa vào phương pháp SGD:

|  |  |
| --- | --- |
|  | (2‑43) |
|  | (2‑44) | |

Ở đây tốc độ học sẽ sử dụng giá trị 0,01. Đây là một giá trị được khuyến khích sử dụng bởi giá trị này không quá lớn cũng không quá nhỏ.

# ỨNG DỤNG PHƯƠNG PHÁP CHO BÀI TOÁN

## Mô tả bài toán

Vào thời điểm thực hiện đồ án này, lĩnh vực xử lý ngôn ngữ tự nhiên đang trở thành một đề tài bùng nổ trong lĩnh vực học máy với rất nhiều ứng dụng thực tế đang được sử dụng hàng ngày trong hầu hết mọi lĩnh vực. Chúng ta có thể thấy được những thông tin về trí tuệ nhân tạo liên tục được cập nhật trên những trang báo điện tử, những kênh thông tin truyền hình như VTV,VTC. Đỉnh điểm của sự bùng nổ này là khi chatbot ChatGPT được OpenAI phát hành và được biết đến rộng rãi vào năm 2023. Chatbot này đã mang đến rất nhiều lợi ích cho con người như tư vấn, hỗ trợ và nhiều tác vụ khác với sự tương tác bằng ngôn ngữ tự nhiên. Kéo theo sự thành công của ChatGPT thì hàng loạt những mô hình đã được các công ty công nghệ lớn cho ra đời như Bard của Google, LLaMA của Meta AI hay Bing AI của Microsoft. Các nhà khoa học Việt Nam cũng đang tiến hành phát triển mô hình PhoGPT đang sắp được công bố bới VinAI. Với việc những mô hình học máy như vậy đang liên tục được cho ra đời và phát triển đã khiến lĩnh vực xử lý ngôn ngữ tự nhiên có thể nói là lĩnh vực nóng nhất trong những năm gần đây.

Xử lý ngôn ngữ tự nhiên không chỉ bao gồm chatbot mà còn rất nhiều vấn đề khác nhau và một bài toán cũng đang rất được quan tâm là phát hiện ý định. Bài toán này được biết đến từ khá lâu về trước nhưng nổi lên những năm gần đây với sự bùng nổ của dữ liệu. Với việc mỗi ngày có đến hàng trăm nghìn các bài đăng trên các diễn đàn mạng xã hội thì việc phát hiện ra ý định sẽ giúp ích rất nhiều với các lĩnh vực cần thiết. Có rất nhiều ý định khác nhau có thể phát hiện ra được trong một bài đăng và để phù hợp với đề tài nghiên cứu thì em chọn bài toán phát hiện ý định mua bán. Với một số các trang mua bán trên mạng xã hội trên Facebook thì họ tiến hành việc liên tục gửi những lời đề nghị qua hộp thư messenger cho tất cả mọi người, điều này gây ra sự khó chịu cho những người sử dụng mạng xã hội này mà không có nhu cầu mua bán. Với những trang thảo luận nổi tiếng như Reddit thì việc tìm ra những bài đăng có ý định mua bán hàng hóa sẽ giúp ích rất nhiều cho chiến lược marketing của các doanh nghiệp. Nếu tiến hành phân loại này bằng cách thủ công sẽ tốn rất nhiều công sức và tiền bạc nên việc áp dụng một mô hình học máy cho lĩnh vực này để tự động phân loại sẽ mang lại rất nhiều lợi ích. Đây có thể nói là một bài toán có giá trị lớn và rất quan trọng với những doanh nghiệp mua bán trực tuyến.

Với mục tiêu là tìm ra những bài đăng có hoặc không có ý định mua bán nên em sẽ quy ước các bài đăng có ý định sẽ mang nhãn positive (+) và các bài đăng không có ý định sẽ mang nhãn negative (-). Một ví dụ cụ thể cho hai nhãn này trình bày như sau:

* **Bài đăng positive (+):** “ok guys im in the market for a new tv and i like the look of these two. both have a lan port but i can only find info on the 590 model saying its dlna compatable which is important to me. also the 590 has 2 more hdmi connections one is £550 and the other just under £600 so its the max i want to pay, reading up on here the tv's both sound ideal but since theyve been out a while now, does anyone have any negative thoughts since theyve owned one for a while OR can i get better for the SAME money (not a penny more ) its for watching football and blurays.” (Bài đăng này có ý định mua một chiếc TV để xem bóng đá và đang phân vân về việc chọn loại nào là tốt nhất)
* **Bài đăng negative (-):** “Dear All, Panasonic Plasma So i walk in the the room my TV sits in one morning and WOW i have this BAR of red dots bang on in the middle of the TV from the TOP to the Bottom. I have no idea where its cme from or how its happened. Hoping some of you can shed some light. Model of the TV IS: TH-65PF10. ” (Bài đăng này không có ý định mua bán, người này chỉ muốn tìm hiểu vấn đề chiếc TV đang gặp phải)

Sau khi tiến hành việc quy bài toán về phân loại 2 nhãn là positive (+) và negative (-), chúng ta sẽ tiến hành áp dụng phương pháp Lifelong machine learning cho bài toán này theo như cách đâ đề cập trong mục 2.8.

## Ngôn ngữ và thư viện dùng xây dựng

### Ngôn ngữ Java

Java là ngôn ngữ lập trình đa nền tảng (cross-platform), được phát triển bởi James Gosling tại Sun Microsystems (nay là Oracle Corporation). Ngôn ngữ lập trình này ra đời vào năm 1995 và được thiết kế để có thể chạy trên các nền tảng khác nhau, từ máy tính cá nhân đến thiết bị di động, các máy chủ và thiết bị nhúng.

Java sử dụng cấu trúc lập trình hướng đối tượng (object-oriented programming - OOP) và được xây dựng trên cơ sở của ngôn ngữ lập trình C++. Nó cung cấp một môi trường chạy ảo (virtual machine) gọi là Java Virtual Machine (JVM), giúp các chương trình Java có thể chạy trên nhiều nền tảng khác nhau mà không cần phải biên dịch lại.



Hình 3.1 Ngôn ngữ lập trình Java

Java là một ngôn ngữ lập trình nổi bật nằm trong những ngôn ngữ được sử dụng nhiều nhất trên thế giới với các tính năng:

* **Đa nền tảng**: Java được thiết kế để có thể chạy trên nhiều nền tảng khác nhau, vì vậy nó rất phù hợp cho việc phát triển các ứng dụng đa nền tảng. Java sử dụng một máy ảo (JVM - Java Virtual Machine) để chạy mã nguồn, vì vậy mã nguồn được viết một lần và có thể chạy trên nhiều hệ điều hành khác nhau mà không cần thay đổi.
* **Quản lý bộ nhớ tự động**: Java có tính năng tự động quản lý bộ nhớ, tức là nó tự động thu dọn các vùng nhớ không sử dụng nữa để giảm thiểu các lỗi bộ nhớ. Điều này giúp cho các ứng dụng được viết bằng Java có thể chạy ổn định và tránh các lỗi liên quan đến bộ nhớ.
* **Hỗ trợ đa luồng**: Java có thể xử lý đa luồng, cho phép chương trình thực hiện nhiều tác vụ cùng một lúc. Điều này giúp cho các ứng dụng có thể chạy nhanh và hiệu quả hơn, đặc biệt là khi phải xử lý nhiều tác vụ cùng một lúc.
* **Tính bảo mật cao**: Java có các tính năng bảo mật như kiểm tra kiểu tĩnh và kiểm tra lỗi trên đường dẫn. Java được thiết kế để giảm thiểu các lỗ hổng bảo mật và các vấn đề liên quan đến an ninh.
* **Hỗ trợ các thư viện và công cụ phát triển mạnh mẽ**: Java có các thư viện tiêu chuẩn và các công cụ phát triển như IDE (Integrated Development Environment) để giúp các nhà phát triển tạo ra các ứng dụng dễ dàng hơn. Các thư viện và công cụ này cung cấp các tính năng như: tạo giao diện người dùng, xử lý dữ liệu, kết nối cơ sở dữ liệu và nhiều hơn nữa,...
* **Tính di động**: Java được sử dụng rộng rãi trong lĩnh vực di động bao gồm: các thư viện hỗ trợ việc phát triển ứng dụng di động, đóng gói ứng dụng thành các file .jar hoặc .apk, cung cấp các tính năng như xử lý đa nhiệm, kết nối mạng và tích hợp với các thiết bị phần cứng như máy ảnh.
* **Tính độc lập với nền tảng**: Java có thể chạy trên nhiều nền tảng khác nhau và không phụ thuộc vào bất kỳ nền tảng cụ thể nào. Điều này giúp cho các ứng dụng Java có thể được triển khai trên nhiều hệ thống khác nhau mà không cần sửa đổi mã nguồn.
* **Tính kế thừa và đa hình**: Java là ngôn ngữ lập trình hướng đối tượng (OOP), vì vậy nó có các tính năng như kế thừa, đa hình và đóng gói. Các tính năng này giúp cho mã nguồn được tái sử dụng và giảm thiểu sự trùng lặp trong mã nguồn.
* **Tính mở rộng**: Java có tính năng mở rộng, cho phép các nhà phát triển thêm các tính năng mới vào ngôn ngữ bằng cách tạo các thư viện và API riêng. Điều này giúp cho Java có thể được sử dụng trong nhiều lĩnh vực khác nhau và được phát triển theo các hướng khác nhau.

Hiện nay Java được sử dụng phổ biến trong rất nhiều lĩnh vực của đời sống như:

* **Phát triển ứng dụng di động**: Java là một trong những ngôn ngữ lập trình phổ biến nhất để phát triển các ứng dụng di động cho [hệ điều hành Android](https://fptshop.com.vn/tin-tuc/danh-gia/he-dieu-hanh-android-la-gi-lich-su-va-nhung-cau-chuyen-thu-vi-ve-android-152248). Nó được sử dụng để phát triển các ứng dụng cho điện thoại di động và [máy tính bảng](https://fptshop.com.vn/may-tinh-bang). Java cung cấp một số khung như Android SDK để phát triển ứng dụng di động.
* **Phát triển game**: Java cũng được sử dụng để phát triển các game trên nhiều nền tảng khác nhau, bao gồm các game trên máy tính, [điện thoại](https://fptshop.com.vn/dien-thoai) di động và máy tính bảng. Các tính năng của Java như: tính đa luồng, xử lý đồ họa âm thanh đã được sử dụng để phát triển các game phức tạp.
* **Phát triển web**: Java cung cấp các công cụ để phát triển các ứng dụng web động và tĩnh. Servlets, JSP (Java Server Pages), Spring Framework, Struts Framework,... là một số ví dụ về các công cụ phổ biến được sử dụng để phát triển các ứng dụng web bằng Java. Java cũng được sử dụng để phát triển các ứng dụng trên máy chủ và trình duyệt web.
* **Phát triển các hệ thống quản lý cơ sở dữ liệu**: Java được dùng để phát triển các hệ thống quản lý cơ sở dữ liệu như Oracle và MySQL. Đồng thời, Java cũng cung cấp các API để kết nối và tương tác với các cơ sở dữ liệu, cho phép người phát triển tạo các ứng dụng quản lý cơ sở dữ liệu phức tạp và linh hoạt.
* **Phát triển các ứng dụng IoT**: Java có thể phát triển các ứng dụng IoT (Internet of Things). Java có tính năng đa nền tảng, cho phép các ứng dụng được phát triển trên các thiết bị di động và chạy trên các thiết bị IoT khác nhau. Ngôn ngữ này cung cấp các công cụ và thư viện để phát triển các ứng dụng IoT, dùng để kết nối các thiết bị và xử lý dữ liệu.
* **Phát triển các ứng dụng trí tuệ nhân tạo**: Java cũng được sử dụng để phát triển các ứng dụng trí tuệ nhân tạo (AI). Các tính năng của Java như xử lý đa luồng và tính năng hướng đối tượng cho phép người phát triển tạo các ứng dụng AI phức tạp.
* **Phát triển các ứng dụng blockchain**: Java cũng có thể được sử dụng để phát triển các ứng dụng blockchain. Các tính năng của Java như độ bảo mật cao, độ tin cậy và khả năng xử lý các giao dịch trên mạng đã được sử dụng để phát triển các ứng dụng blockchain phức tạp.

Tuy vậy ngôn ngữ Java vẫn còn một vài hạn chế có thể kể đến như sau:

* Tốc độ chậm hơn so với các ngôn ngữ lập trình gần sát với phần cứng, chẳng hạn như C hoặc C++.
* Java có thể chạy trên nhiều nền tảng khác nhau, nhưng ứng dụng này có thể cần đến một trình biên dịch hoặc máy ảo Java riêng biệt để có thể chạy trên các thiết bị di động.
* Sử dụng bộ nhớ lớn hơn so với một số ngôn ngữ lập trình khác.
* Cú pháp phức tạp hơn so với một số ngôn ngữ lập trình khác

### Thư viện Stanford NLP

Thư viện Stanford NLP (Stanford Natural Language Processing) là một bộ công cụ được phát triển dựa trên ngôn ngữ lập trình Java bởi những giảng viên, sinh viên cùng những nhà nghiên cứu tại Đại học Stanford giúp máy tính có thể hiểu và xử lý ngôn ngữ của con người. Họ phát triển nhiều tài liệu giáo dục về NLP và những công cụ cho cộng đồng sử dụng có thể nói đến như bộ công cụ Stanza xử lý văn bản trong hơn 60 ngôn ngữ khác nhau.

Một trong những ưu điểm chính của Standford NLP là khả năng mở rộng rất cao, điều này làm cho nó trở thành lựa chọn phù hợp cho các tác vụ phức tạp. Một yếu tố khác là nó được xây dựng chú trọng đến tốc độ – nó được tối ưu hóa để thực hiện tác vụ cực kỳ nhanh cũng như là mã nguồn mở khiến cho bộ công cụ này trở thảnh một trong những thư viện NLP được sử dụng nhiều nhất trên thế giới.



Hình 3.2 Thư viện Stanford NLP

Một số tính năng của thư viện Stanford NLP có thể nói đến như:

* **Phân loại văn bản (Text Classification):** Cung cấp các công cụ để phân loại văn bản vào các lớp khác nhau, chẳng hạn như phân loại chủ đề hoặc phân loại tâm trạng.
* **Phân tích cú pháp (Parsing):** Hỗ trợ phân tích cú pháp của câu văn để hiểu cấu trúc ngữ pháp của chúng.
* **Nhận dạng thực thể (Named Entity Recognition - NER):** Nhận diện và phân loại các thực thể như tên người, địa điểm, nhãn hiệu và các loại thực thể khác.
* Tách từ (Tokenization): Tách một từ, câu hay văn bản thành những đơn vị nhỏ hơn. Mỗi đơn vị nhỏ được gọi là một token.

Thư viện Stanford NLP thường được sử dụng trong nghiên cứu và phát triển ứng dụng NLP do cung cấp một bộ công cụ mạnh mẽ và linh hoạt. Thư viện này được viết bằng Java, nhưng có các giao diện và bản dịch cho nhiều ngôn ngữ khác nhau.

## Bộ dữ liệu sử dụng

Đồ án sử dụng bộ dữ liệu NAACL2013-Chen-Intention dành cho bài toán phát hiện ý định được lấy từ 4 domain khác nhau. Mỗi domain trong đó sẽ chứa 1000 bài đăng. Mỗi bài đăng được gắn nhãn "1" (có ý định mua bán) hoặc "0" (Không có ý định mua bán). Dữ liệu từ 4 domain được lấy từ các trang web sau:

* TV: https://www.avforums.com/forums/general-tv-discussions-forum.104/ [13]
* Electronics: <https://www.avsforum.com/forums/> [13]
* Camera: <http://forum.digitalcamerareview.com/> [13]
* Cellphone: <https://www.howardforums.com/forums.php> [13]

Chi tiết về dữ liệu được mô tả theo bảng dưới đây:

Bảng 3.1 Thống kê về dữ liệu sử dụng [13]

|  |  |  |  |
| --- | --- | --- | --- |
| Bộ dữ liệu | Có ý định (+) | Không có ý định (-) | Tổng số dữ liệu |
| TV | 263 | 737 | 1000 |
| Electronics | 280 | 720 | 1000 |
| Camera | 282 | 718 | 1000 |
| Cellphone | 184 | 816 | 1000 |

Chúng ta có thể nhận xét dữ liệu này được phân bố không đều khi số lượng các bài đăng không có ý định mua bán nhiều hơn khá nhiều so với các bài đăng có ý định mua bán. Chính vì vậy chúng ta sẽ sử dụng 3 thang đo Precision, Recall và F1 để đánh giá hiệu quả của mô hình này.

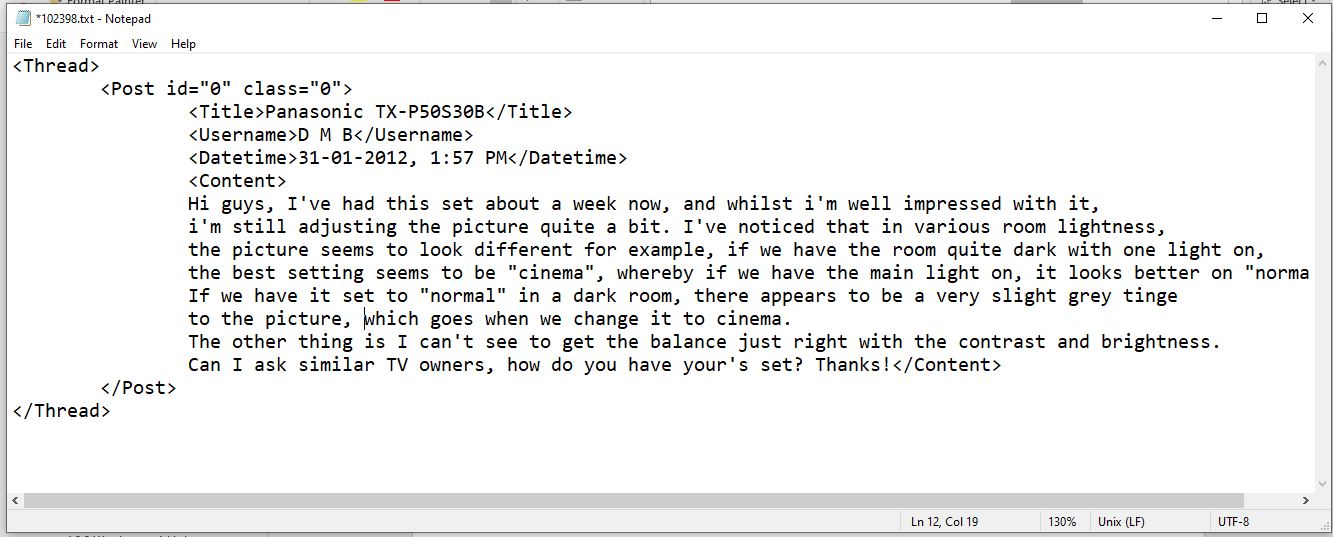
# THỰC NGHIỆM VÀ KẾT QUẢ

## Các bước thực hiện

Các bước thực hiện đã được nêu trong hình Hình 2.6. Chúng ta tiến hành theo các bước sau.

### Tiền xử lý dữ liệu

Sau khi có được bộ dữ liệu NAACL2013-Chen-Intention, mỗi file trong bộ dữ liệu đó sẽ có định dạng như hình sau:



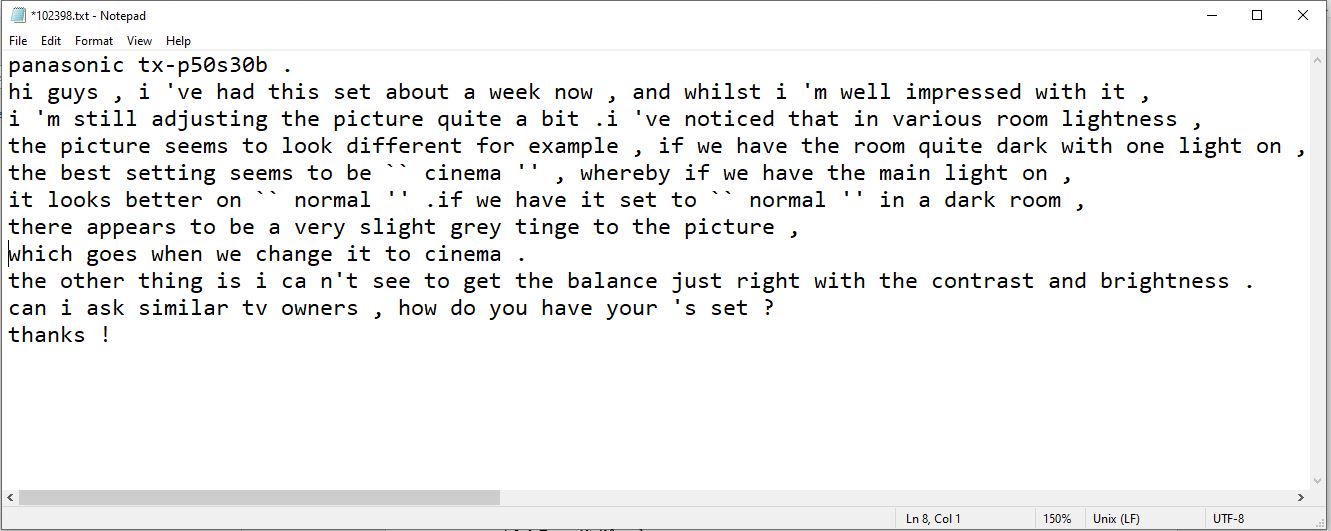
Hình 4.1 Cấu trúc một file dữ liệu không mang ý định mua bán

Ta có thể thấy cấu trúc file này là định dạng file XML với ý nghĩa của từng thẻ được phát biểu như sau:

Bảng 4.1 Ý nghĩa của các thẻ trong một dữ liệu

|  |  |
| --- | --- |
| Tên thẻ | Ý nghĩa của thẻ |
| Thread | Là nơi chứa bài đăng mới của một diễn đàn |
| Post | Chứa 2 trường id và class biểu thị cho mã số của bài đăng và lớp mà bài đăng thuộc về (1: có ý định, 0:không có ý định) |
| Title | Chủ đề của bài đăng |
| Username | Tên tài khoản của người đăng bài |
| Datetime | Thời gian đăng bài |
| Content | Nội dung của bài đăng |

Với mục đích của đồ án là xác định xem bài đăng là có hay không có ý định mua bán nên chúng ta chỉ quan tâm tới tiêu đề và nội dung của bài đăng qua thẻ Content. Việc này có thể tiến hành bằng cách sử dụng thư viện Standford Tokenizer để tách câu, tách từ. Kết quả sau khi tách sẽ có định dạng như sau:



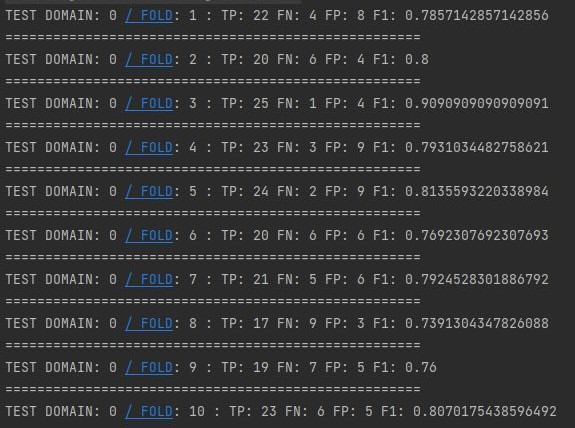
Hình 4.2 Định dạng dữ liệu sau khi lấy nội dung trong thẻ Content

## Thực nghiệm

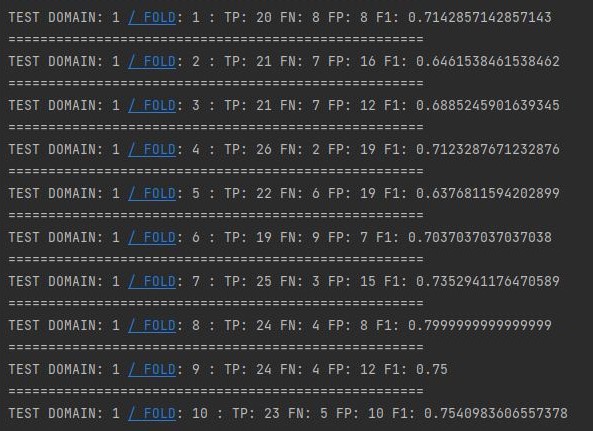
Các thực nghiệm sử dụng trong đồ án này sẽ sử dụng áp dụng phương pháp trích chọn đặc trưng IG kết hợp với đặc trưng trigram. Điều này có được dựa trên kết quả của tài liệu tham khảo Z. Chen et al [13]. Phương pháp kiểm tra độ chính xác k-folds cross-validation sẽ được áp dụng với k=10. Toàn bộ dữ liệu sẽ được chia làm 10 tập con và quá trình học máy sẽ tiến hành 10 lần. Trong mỗi lần một tập con sẽ được sử dụng để tiến hành kiểm thử còn 9 tập con còn lại sẽ được phục vụ cho việc huấn luyện. 3 độ đo là Precision, Recall và F1 được tính trung bình trên 10 fold cho mỗi target domain.

### Thực nghiệm 1

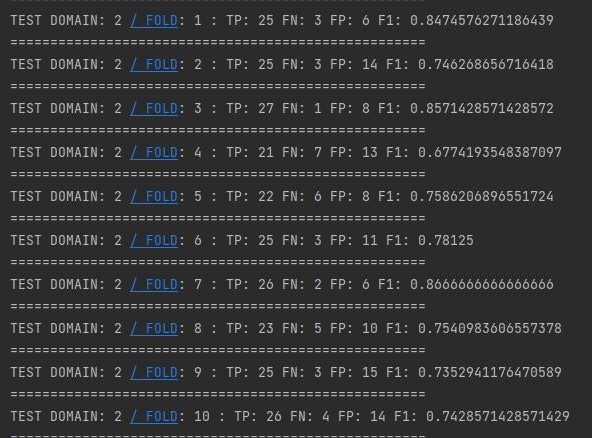
Như đã đề cập về việc chạy giải thuật trên các thực nghiệm trong phần 2.2.2.3, thực nghiệm đầu tiên áp dụng cho mô hình sẽ sử dụng thuật toán NB theo phương pháp truyền thống trên dữ liệu mới mà không áp dụng những tri thức tích lũy được trong quá khứ. Thực nghiệm này sẽ chỉ sử dụng dữ liệu của target domain với 90% lượng dữ liệu sẽ được sử dụng làm dữ liệu huần luyện và 10% còn lại sẽ được phục vụ cho quá trình kiểm thử. Thực nghiệm này sẽ sử dụng 1500 đặc trưng tốt nhất cho quá trình tiến hành.



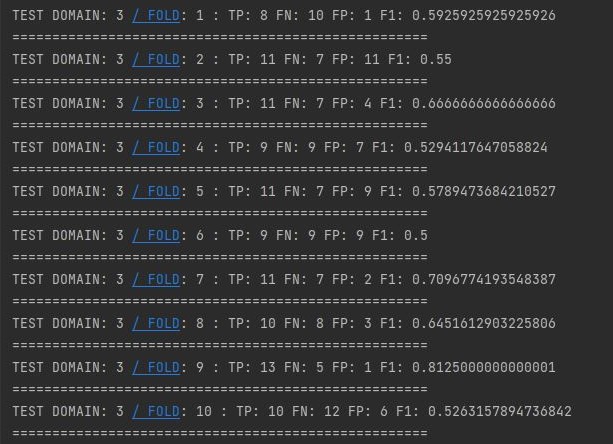
Hình 4.3 Kết quả thực nghiệm 1 cho domain TV



Hình 4.4 Kết quả thực nghiệm 1 cho domain Electronics



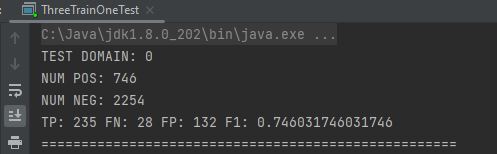
Hình 4.5 Kết quả thực nghiệm 1 cho domain Camera



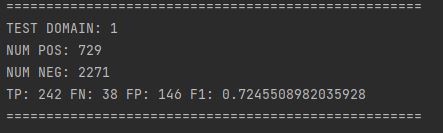
Hình 4.6 Kết quả thực nghiệm 1 cho domain Cellphone

### Thực nghiệm 2

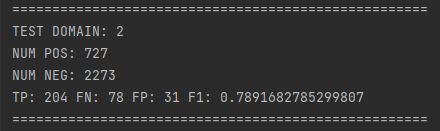
Thực nghiệm 2 chúng ta sẽ chỉ sử dụng toàn bộ dữ liệu của source domain phục vụ cho quá trình huấn luyện mô hình. Quá trình kiểm thử sẽ áp dụng trên dữ liệu từ target domain. Cứ 3 domain sẽ được chọn làm source domain và domain còn lại sẽ được chọn làm target domain. Thực nghiệm này sẽ sử dụng 3000 đặc trưng tốt nhất cho quá trình tiến hành.



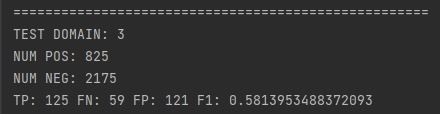
Hình 4.7 Kết quả thực nghiệm 2 cho domain TV



Hình 4.8 Kết quả thực nghiệm 2 cho domain Electronics



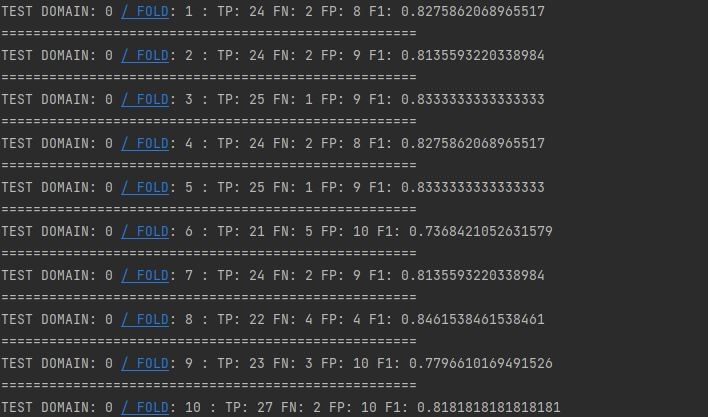
Hình 4.9 Kết quả thực nghiệm 2 cho domain Camera



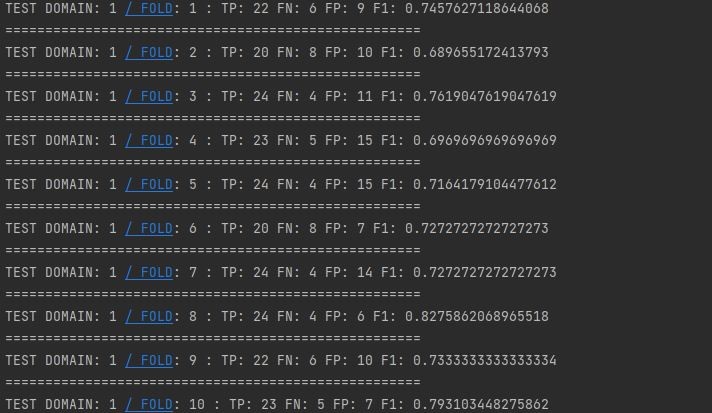
Hình 4.10 Kết quả thực nghiệm 2 cho domain Cellphone

### Thực nghiệm 3

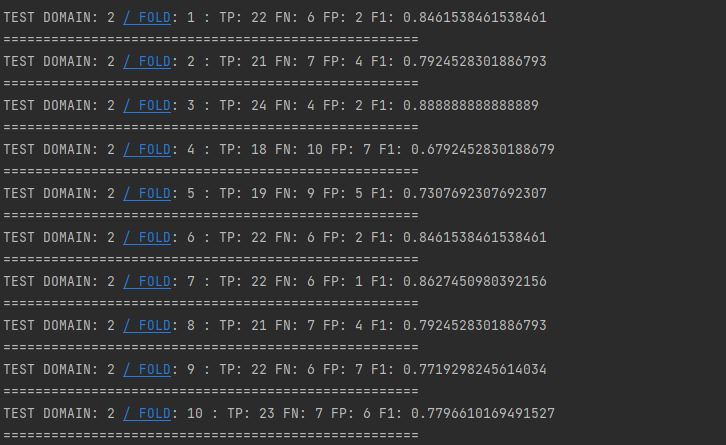
Trong thực nghiệm này, chúng ta sẽ sử dụng dữ liệu trong cả source domain và target domain. Toàn bộ dữ liệu của source domain cùng với 90% dữ liệu trong target domain sẽ được sử dụng để phục vụ quá trình huấn luyện. 10% dữ liệu còn lại trong target domain sẽ được sử dụng cho quá trình kiểm thử. Thực nghiệm này sẽ sử dụng 1500 đặc trưng từ mỗi source domain và 2500 đặc trưng từ target domain cho quá trính tiến hành.



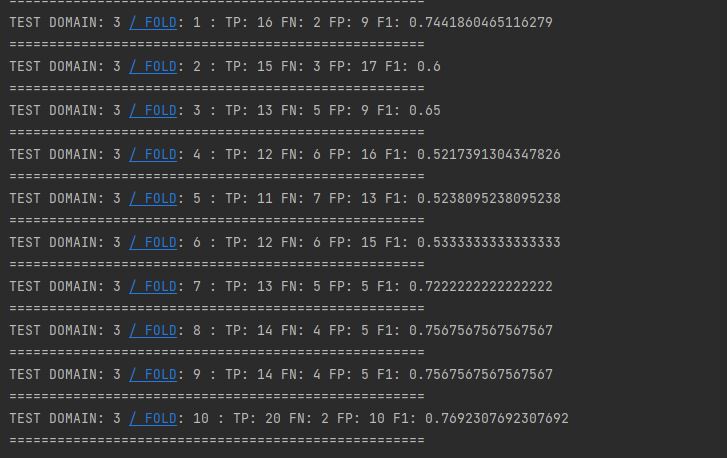
Hình 4.11 Kết quả thực nghiệm 3 cho domain TV



Hình 4.12 Kết quả thực nghiệm 3 cho domain Electronics



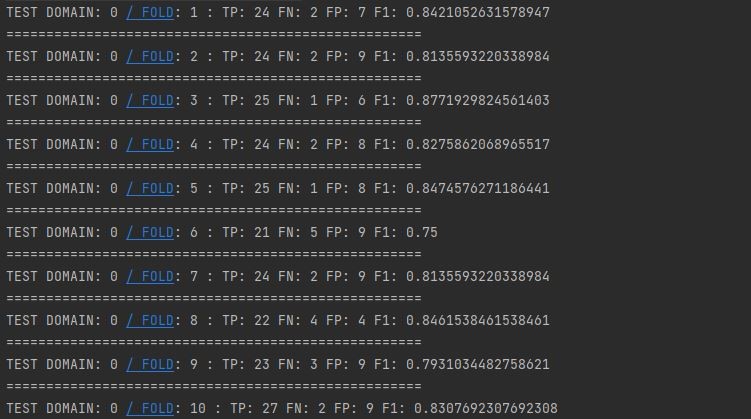
Hình 4.13 Kết quả thực nghiệm 3 cho domain Camera



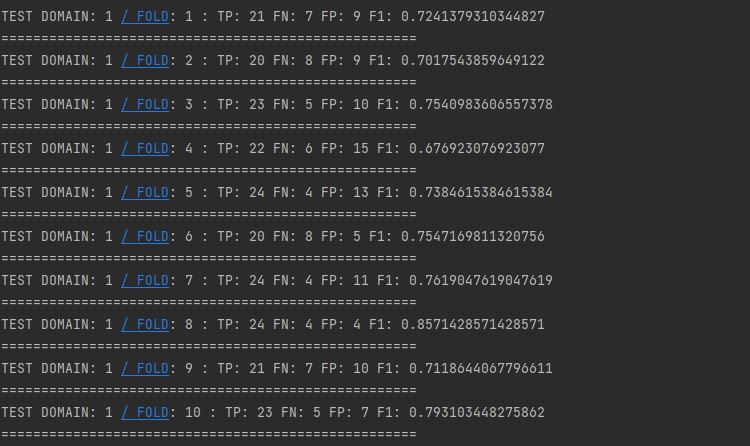
Hình 4.14 Kết quả thực nghiệm 3 cho domain Cellphone

### Thực nghiệm 4

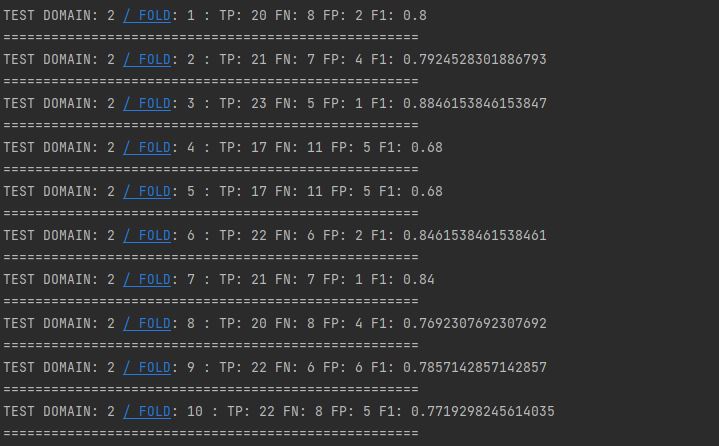
Đây sẽ là thực nghiệm mà chúng ta áp dụng mô hình Lifelong machine learning đã xây dựng được. Quá trình chia dữ liệu cũng giống như thực nghiệm 3. Thực nghiệm này sẽ sử dụng 1500 đặc trưng từ mỗi source domain và 2500 đặc trưng từ target domain cho quá trính tiến hành.



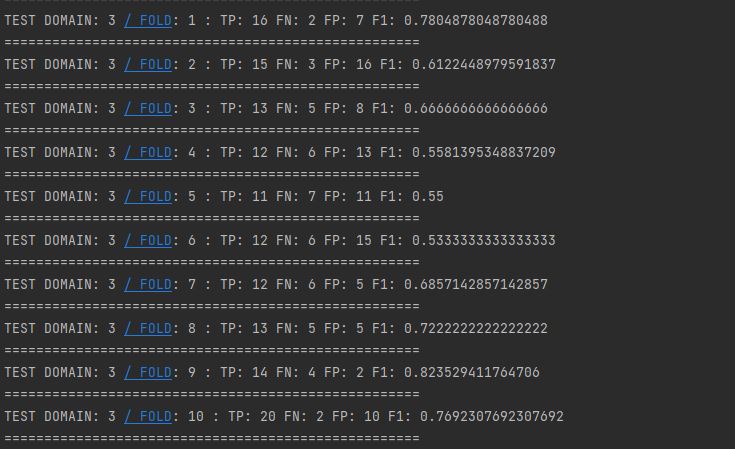
Hình 4.15 Kết quả thực nghiệm 4 cho domain TV



Hình 4.16 Kết quả thực nghiệm 4 cho domain Electronics



Hình 4.17 Kết quả thực nghiệm 4 cho domain Camera



Hình 4.18 Kết quả thực nghiệm 4 cho domain Cellphone

## Tổng hợp kết quả và đánh giá

### Tổng hợp kết quả

Sau khi có được kết quả từ việc chạy mô hình trên các thực nghiệm, chúng ta tổng hợp được bảng kết quả như sau:

Bảng 4.2 Kết quả thực nghiệm với domain TV

|  |  |  |  |
| --- | --- | --- | --- |
| Thực nghiệm | Precision(%) | Recall(%) | F1(%) |
| Thực nghiệm 1 | **78,85** | 81,39 | 80,10 |
| Thực nghiệm 2 | 64,03 | 89,35 | 74,60 |
| Thực nghiệm 3 | 73,76 | **90,85** | 81,42 |
| Thực nghiệm 4 | 75,58 | **90,85** | **82,51** |

Bảng 4.3 Kết quả thực nghiệm với domain Electronics

|  |  |  |  |
| --- | --- | --- | --- |
| Thực nghiệm | Precision(%) | Recall(%) | F1(%) |
| Thực nghiệm 1 | 65,02 | 80,36 | 71,88 |
| Thực nghiệm 2 | 62,37 | **86,43** | 72,45 |
| Thực nghiệm 3 | 69,09 | 80,71 | 74,45 |
| Thực nghiệm 4 | **71,17** | 79,28 | **75,01** |

Bảng 4.4 Kết quả thực nghiệm với domain Camera

|  |  |  |  |
| --- | --- | --- | --- |
| Thực nghiệm | Precision(%) | Recall(%) | F1(%) |
| Thực nghiệm 1 | 70,49 | **86,88** | 77,83 |
| Thực nghiệm 2 | **86,81** | 72,34 | 78,92 |
| Thực nghiệm 3 | 84,56 | 75,88 | **79,99** |
| Thực nghiệm 4 | 85,58 | 72,69 | 78,61 |

Bảng 4.5 Kết quả thực nghiệm với domain Cellphone

|  |  |  |  |
| --- | --- | --- | --- |
| Thực nghiệm | Precision(%) | Recall(%) | F1(%) |
| Thực nghiệm 1 | **69,04** | 56,21 | 61,79 |
| Thực nghiệm 2 | 50,81 | 67,93 | 58,14 |
| Thực nghiệm 3 | 58,94 | **75,76** | 66,30 |
| Thực nghiệm 4 | 61,93 | 74,65 | **67,70** |

### Đánh giá kết quả

Ở đây chúng ta sẽ tập trung đánh giá dựa vào độ đo F1 bởi nó thể hiện sự hài hòa giữa 2 độ đo Precision, Recall và là độ đo được ưu tiên sử dụng trong các mô hình học máy cho bài toán phân lớp. Một vài nhận xét có thể rút ra như sau:

* Dựa vào tiêu chí đầu tiên đánh giá cho mô hình Lifelong machine learning, khi nhìn vào kết quả tổng hợp cho độ đo F1 từ thực nghiệm 1 và 2 chúng ta thấy rằng: F1 ở domain Camera và Electronics tại thực nghiệm 2 cho kết quả tốt hơn còn F1 ở domain Cellphone và TV tại thực nghiệm 1 lại cho kết quả tốt hơn. Từ đó có thể nhận xét rằng việc sử dụng thêm dữ liệu từ ngoài target domain có thể mang lại kết quả tốt hơn hoặc tệ hơn tùy vào trường hợp cụ thể.
* So sánh thực nghiệm 1,2 với thực nghiệm 3,4 chúng ta có thể thấy được rằng việc có thêm nhiều dữ liệu từ những nhiệm vụ cũ được thêm vào sẽ phần nào mang lại hiệu quả tốt hơn cho mô hình.
* Độ đo F1 cho kết quả gần như tốt nhất tại thực nghiệm 4 khi có giá trị cao nhất tại 3/4 domain. Điều này có thể nói việc áp dụng phương pháp Lifelong machine learning phần nào sẽ mang tới hiệu quả tốt hơn khi áp dụng được khi thức tích lũy được trong quá khứ.

Dữ liệu sử dụng cho đồ án chỉ là 4000 mẫu dữ liệu từ 4 domain khác nhau, việc này phần nào gây ra độ chính xác chưa được cao cho bài toán. Để khắc phục vấn đề này có thể sử dụng bộ dữ liệu lớn hơn hay thử nghiệm những kĩ thuật tối ưu khác.

# KẾT LUẬN

Sau khi thực hiện xong đồ án, một vài mục tiêu cơ bản đã đạt được như sau:

* Nắm được cơ bản về phương pháp Lifelong machine learning.
* Áp dụng phương pháp Lifelong machine learning cho bài toán cụ thể.
* Rút ra được những nhận xét về việc sử dụng phương pháp Lifelong machine learning.

Một vài hạn chế của đồ án có thể nói đến như:

* Chỉ phân loại được bài đăng có hay không ý định mua bán với dữ liệu Tiếng Anh, chưa sử dụng được cho Tiếng Việt.
* Các phương pháp sử dụng trong đồ án là những phương pháp cũ, chưa sử dụng tới deep learning.

Qua đồ án này chúng ta cũng có thể rút ra được nếu ta tận dụng được các tri thức đã được tích lũy trong quá trình học và sử dụng chúng cho nhiệm vụ mới cần học sẽ thu được kết quả tốt hơn so với việc chỉ học trên một domain nhất định.

Lifelong machine learning là một phương pháp hiện vẫn còn nhiều khó khăn và đang liên tục được các nhà khoa học nghiên cứu. Ý tưởng chủ đạo của phương pháp này vẫn sẽ là 3 thành phần chính: (1) học một cách liên tục, (2) tích lũy tri thức qua các nhiệm vụ đã được học, (3) khả năng sử dụng những tri thức đã được tích lũy đó để giúp đỡ các nhiệm vụ cần học trong tương lai. Phương pháp này trong tương lai gần hứa hẹn sẽ mang lại nhiều giá trị cho lĩnh vực trí tuệ nhân tạo.

**Hướng phát triển của bài toán**

Trong tương lai, bài toán này có thể hướng tới việc phân loại cho dữ liệu Tiếng Việt, điều này yêu cầu phải thu thập dữ liệu dữ liệu Tiếng Việt và gán nhãn để sử dụng. Các đặc trưng về dữ liệu Tiếng Việt cũng sẽ được trích chọn khác với dữ liệu Tiếng Anh.

Không chỉ ý định mua bán, bài toán này còn có thể phát triển ra những ý định khác đang tiềm ẩn trong dữ liệu, có thể mang lại nhiều lợi ích cho cuộc sống hơn.

Hiện này Lifelong machine learning đã phát triển thêm một lĩnh vực là Continual learning (cách gọi cho việc áp dụng phương pháp này trong deep learning) xuất hiện trong năm 2022. Các nghiên cứu về phương pháp này mang lại hiệu quả cao và đang liên tục cập nhật trong Continual AI. Việc áp dụng những phương pháp đó hứa hẹn sẽ mang lại hiệu quả tốt hơn so với phương pháp mà đồ án đang sử dụng.

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | Z. Chen and B. Liu, "Lifelong Machine Learning," in *Synthesis Lectures on Artificial Intelligence and Machine Learning*, Morgan and Claypool, 2016. |
| [2] | G. Fei et al, "Learning Cumulatively to Become More Knowledgeable," *KDD,* 2016. |
| [3] | Z. Chen et al, "Lifelong Learning for Sentiment Classification," *ACL,* pp. 750-756, 2015. |
| [4] | "What is Transfer learning ?," Geeksforgeeks, 29 11 2023. [Online]. Available: https://www.geeksforgeeks.org/ml-introduction-to-transfer-learning/. |
| [5] | Z. Chen and B. Liu, "Lifelong and Continual Learning," [Online]. Available: https://www.cs.uic.edu/~liub/lifelong-learning.html. |
| [6] | "Difference between single task learning and multitask learning," ResearchGate, [Online]. Available: https://www.researchgate.net/figure/Difference-between-single-task-learning-and-multitask-learning-a-single-task-learning\_fig2\_307622018. |
| [7] | N. T. Trung and T. M. Tuấn, "Phân tích dữ liệu lớn: Phần 2," Đại học Thủy Lợi, [Online]. Available: http://tailieuso.tlu.edu.vn/handle/DHTL/8914. |
| [8] | V. H. Tiệp, "Bài 32: Naive Bayes Classifier," 8 8 2017. [Online]. Available: https://machinelearningcoban.com/2017/08/08/nbc/. |
| [9] | "N-Grams," deepai.org, [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/n-gram. |
| [10] | N. X. Bách et al, "Cross-Domain Intention Detection in Discussion Forums," *SoICT,* pp. 173-180, 2017. |
| [11] | V. H. Tiệp, "Bài 8: Gradient Descent (phần 2/2)," 16 1 2017. [Online]. Available: https://machinelearningcoban.com/2017/01/16/gradientdescent2/. |
| [12] | "Các phương pháp đánh giá mô hình học máy, học sâu (Machine learning & Deep learning)," Rabiloo, [Online]. Available: https://rabiloo.com/vi/blog/cac-phuong-phap-danh-gia-mo-hinh-machine-learning-va-deep-learning. |
| [13] | Z. Chen et al, "Identifying Intention Posts in Discussion Forums," *ACL,* p. 1041–1050. |
| [14] | "Stanford Tokenizer," [Online]. Available: https://nlp.stanford.edu/software/tokenizer.html. |
| [15] | "Java Tutorial," w3schools, [Online]. Available: https://www.w3schools.com/java/. |