ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

PHẠM NGUYÊN BÌNH

ỨNG DỤNG MÔ HÌNH MAXIMUM ENTROPY TRONG PHÂN LỚP QUAN ĐIỂM CHO DỮ LIỆU VĂN BẢN

Ngành: Công nghệ thông tin
Chuyên ngành: Kỹ thuật phần mềm
Mã số: 60480103

TÓM TẮT LUẬN VĂN THẠC SĨ KỸ THUẬT PHẦN MỀM

Hà Nội – 2016

Mục lục

Danh sách hình vẽ

Danh sách bảng biểu

MỞ ĐẦU

1. Tính cấp thiết của đề tài luận văn

Ngày nay, xã hội của chúng ta đang chứng kiến sự bùng nổ của Internet và đặc biệt là sự phát triển đến chóng mặt của các mạng xã hội như Facebook, Twitter cũng như các diễn đàn, các trang thông tin mạng về đa dạng các lĩnh vực. Chúng ta thường gọi chúng với tên chung là các kênh truyền thông xã hội trực tuyến (social media online). Trên các kênh truyền thông này là một lượng dữ liệu về quan điểm, ý kiến khổng lồ (big data) tới trực tiếp từ hàng trăm triệu người dùng trong nước cũng như quốc tế. Vì lẽ đó, việc giám sát thương hiệu thông qua thu thập, phân tích những phản hồi, ý kiến, đóng góp của người sử dụng trên những kênh truyền thông này là vô cùng quan trọng và hữu ích với các công ty, doanh nghiệp và các tổ chức nói chung. Việc thu thập và xử lý kịp thời các thông tin này sẽ hỗ trợ tích cực cho các công ty, doanh nghiệp và tổ chức thực hiện được: (I) nắm bắt được mức độ phổ biến, lan tỏa và tầm ảnh hưởng của thương hiệu; (II) nắm bắt được tâm tư, nguyện vọng và cả những phản hồi, góp ý trực tiếp từ cộng đồng, những người sử dụng dịch vụ để từ đó đưa ra những điều chỉnh phù hợp; (III) nắm bắt và hiểu được những phản hồi và bình luận trên diện rộng đối với các vấn đề, sự kiện quan trọng của tổ chức; (IV) kịp thời bảo vệ thương hiệu của đơn vị trước những thông tin dư luận thiếu chính xác và sai lệch.

Chính vì lẽ đó, việc phát triển một hệ thống có thể tự động thu thập, phân tích và tổng hợp dữ liệu truyền thông là vô cùng cần thiết và hữu ích đối với sự phát triển của bất cứ một công ty, doanh nghiệp hay tổ chức nào, trong đó có cả Đại học Quốc gia (ĐHQG) Hà Nội. Mục tiêu của nhóm đề tài là xây dựng hệ thống tự động phân tích dữ liệu truyền thông xã hội trực tuyến phục vụ quản lý và hỗ trợ ra quyết định, kinh tế, chính trị, giáo dục và xã hội cho Đại học Quốc gia Hà Nội với

tên gọi VNU-SMM (Vietnam National University-Social Media Monitoring).

2. Mục tiêu của luận văn

Luận văn tập trung vào tìm hiểu các mô hình học máy có giám sát phổ biến, được ứng dụng trong bài toán phân lớp quan điểm người dùng cho dữ liệu văn bản thu được từ các kênh truyền thông xã hội. Trong luận văn, chúng tôi cũng đã lựa chọn bộ phân lớp Maximum Entropy để cài đặt và thử nghiệm, đồng thời ứng dụng vào hệ thống tự động phân tích dữ liệu truyền thông xã hội trực tuyến phục vụ quản lý và hỗ trợ ra quyết định trong lĩnh vực đào tạo cho Đại học Quốc gia Hà Nội.

3. Cấu trúc của luận văn

Luận văn được tổ chức thành năm chương. Trong chương 1, chúng tôi sẽ giới thiệu về bài toán phân lớp quan điểm người dùng, các hướng tiếp cận và các giải pháp đã và đang được nghiên cứu, sử dụng trên thế giới. Trong chương tiếp theo, chúng tôi sẽ mô tả tổng quan về hệ thống tự động thu thập và phân tích dữ liệu truyền thông xã hội trực tuyến cho Đại học Quốc gia Hà Nội - VNU-SMM và vai trò của thành phần phân lớp quan điểm người dùng trong hệ thống. Nội dung chi tiết về bộ phân lớp Maximum entropy và ứng dụng của nó trong bài toán phân tích quan điểm người dung sẽ được chúng tôi trình bày trong chương 3. Trong chương 4, chúng tôi sẽ tập trung trình bày về kết quả thực nghiệm, sau đó đánh giá, phân tích kết quả, những lỗi và điểm yếu còn tồn tại. Cuối cùng, chúng tôi sẽ tổng kết lại những nội dung đã thực hiện trong luận văn, từ đó đề xuất hướng nghiên cứu và phát triển trong tương lai.

Bài toán phân lớp quan điểm và các hướng tiếp cận

1.1 Bài toán phân tích quan điểm

Phân tích quan điểm (opinion mining hay sentiment analysis) là một lĩnh vực nghiên cứu về các ý kiến, quan điểm, đánh giá, thái độ và cảm xúc của mọi người về một đối tượng. Hai thuật ngữ Opinion Mining (OM) và Sentiment Analysis (SA) có thể được sử dụng thay thế cho nhau trong các ngữ cảnh sử dụng. Tuy nhiên, một số nhà nghiên cứu cho rằng OM và SA có một điểm khác nhau nhỏ [14].

Phân tích quan điểm là một lĩnh vực thu hút được sự quan tâm lớn của cộng đồng nghiên cứu nói chung và cộng đồng xử lý ngôn ngữ nói riêng bởi ba yếu tố chính sau: Thứ nhất, đó là sự đa dạng trong ứng dụng của nó vào nhiều lĩnh vực. Thứ hai, đó là sự bùng nổ của thông tin và mạng xã hội. Thứ ba, đó là sự thách thức của bài toán.

Quan điểm được chia làm hai loại: tích cực (positive) và tiêu cực (negative). Ngoài hai trạng thái này, một câu hoặc văn bản được xếp vào dạng trung lập (neutral).

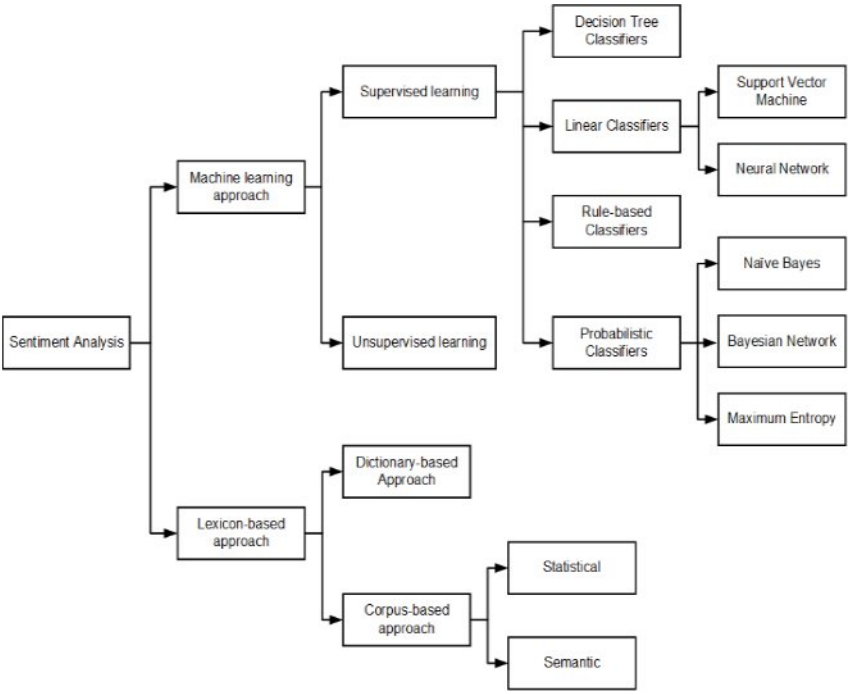
Bài toán phân tích quan điểm người dùng thường được tiếp cận và giải quyết ở ba mức độ: **Mức độ văn bản, tài liệu** (Document level), **Mức độ câu** (Sentence level), **Mức độ khía cạnh** (Aspect level)

1.2 Các hướng tiếp cận và giải quyết bài toán

Trong những năm gần đây, có rất nhiều bài báo và các công trình nghiên cứu cải tiến các thuật toán phân tích quan điểm người dùng. Các kỹ thuật này có thể được phân loại như trong Hình 1.1 [7]. Trong đó ta thấy, có hai hướng tiếp cận chính trong các kỹ thuật ứng dụng trong giải quyết bài toán phân lớp quan điểm người dùng, đó là: sử dụng các thuật toán học máy hoặc tiếp cận theo hướng sử dụng các kiến thức

về từ vựng và ngữ nghĩa. Trong các thuật toán học máy lại có thể được chia ra thành các thuật toán học có giám sát hay học không giám sát. Ngoài ra, trong một, hai năm trở lại đây bắt đầu xuất hiện các ứng dụng thành công của deep learning vào trong bài toán phân tích quan điểm [12,13] đạt kết quả cao.

Các thuật toán học máy có giám sát phổ biến được sử dụng trong giải quyết bài toán phân lớp quan điểm là: Naïve Bayes, Maximum Entropy, Support Vector Machine (SVM) [9]. Các thuật toán này được đánh giá cao về tính chính xác và hiệu quả trong giải quyết bài toán phân lớp quan điểm người dùng. Trong mục này, chúng tôi sẽ giới thiệu tổng quan về các giải thuật học có giám sát này.



Hình 1.1: Các kỹ thuật sử dụng trong giải quyết bài toán phân lớp quan điểm

1.3 Mô hình phân lớp Naïve Bayes

Bộ phân lớp quan điểm Naïve Bayes được xây dựng dựa trên lý thuyết Bayes về xác suất có điều kiện và sử dụng mô hình "bag of words" để phân loại văn bản:

$$P(\mathrm{c} \mid \mathrm{d}) \quad \mathrm{P(c)}.\frac{P(d \mid c)}{P(d)} \tag{1.1}$$

Mục tiêu là tìm được phân lớp c* sao cho P(c*|d) là lớn nhất hay xác suất của tài liệu d thuộc lớp c* là lớn nhất.

Từ công thức trên ta có thể nhận thấy P(d) không đóng vai trò gì trong việc quyết định phân lớp c → P(c|d) lớn nhất ⇔ P(c).P(d|c) lớn nhất.

Để có thể xấp xỉ giá trị của P(d|c), thuật toán Naïve Bayes giả sử rằng: các vector đặc trưng f_i của một tài liệu khi đã biết phân lớp là độc lập với nhau.

Khi tiến hành huấn luyện, thuật toán sử dụng phương pháp xấp xỉ hợp lý cực đại MLE (Maximum Likelihood Estimation) để xấp xỉ P(c) và $P(f_i|c)$ cùng thuật toán làm mịn add-one (add-one smoothing).

Đánh giá bộ phân lớp sử dụng thuật toán học máy Naive Bayes, ta nhận thấy phương pháp này các ưu điểm như: đơn giản, dễ cài đặt, bộ phân lớp chạy nhanh và cần ít bộ nhớ lưu trữ. Bộ phân lớp cũng không cần nhiều dữ liệu huấn luyện để xấp xỉ được bộ tham số. Tuy nhiên, bộ phân lớp này có nhược điểm là thiếu chính xác do giả thiết độc lập của các vector đặc trưng khi đã biết phân lớp là không có thực trong thực tế.

1.4 Mô hình phân lớp SVM

1.4.1 Giới thiệu về SVM

Máy vector hỗ trợ (Support Vector Machine – SVM) là một phương pháp học máy nổi tiếng được sử dụng để giải quyết bài toán

phân lớp, thuật toán được Vladimir N. Vapnik tìm ra và thuật toán SVM tiêu chuẩn hiện nay sử dụng được tìm ra bởi Vapnik và Corinna Cortes vào năm 1995. Nhiều bài toán trong đời sống thực được SVM giải quyết khá thành công như nhận dạng văn bản, hình ảnh, chữ viết tay, phân loại thư rác điện tử, virus…

Thuật toán SVM ban đầu chỉ được thiết kế để giải quyết bài toán phân lớp nhị phân, tức là số lớp hạn chế là hai lớp, với ý tưởng chính như sau:

Cho trước một tập huấn luyện, được biểu diễn trong không gian vector với mỗi điểm là biểu diễn của một dữ liệu, SVM sẽ tìm ra một siêu phẳng f quyết định tốt nhất có thể chia các điểm trên không gian này thành hai lớp riêng biệt, tương ứng là lớp "+" và lớp "-". Chất lượng của siêu phẳng được đánh giá bởi khoảng cách lề (margin) giữa hai lớp: khoảng cách càng lớn thì siêu phẳng quyết định càng tốt và chất lượng phân lớp càng cao.

1.4.2 Bài toán phân lớp nhị phân với SVM

- **Phát biểu bài toán:**

Cho tập mẫu $\{(x_1, y_1), (x_2, y_2), \dots (x_D, y_D)\}$ trong đó $x_i \in R^D$ và $y_i \in \{-1, +1\}$. Giả sử dữ liệu là phân tách tuyến tính, tức là ta có thể phân tách dữ liệu thành hai lớp bằng cách vẽ một đường phẳng trên đồ thị của x_1, x_2 (với D = 2) hoặc một siêu phẳng trên đồ thị của x_1, x_2,… x_D (với D > 2). Mục đích của thuật toán phân lớp SVM là xây dựng siêu phẳng sao cho khoảng cách lề giữa hai lớp đạt cực đại bằng cách xác định phương trình mô tả siêu phẳng đó trên đồ thị.

1.4.3 Bài toán phân lớp đa lớp với SVM

Đối với bài toán phân lớp với số lớp nhiều hơn hai lớp, ta sử dụng kỹ thuật phân đa lớp dạng Multiple Binary Classification với hai chiến lược chính là One-vs-One và One-vs-Rest.

1.4.4 Đánh giá bộ phân lớp SVM

Bộ phân lớp SVM có các ưu điểm như:

- Độ chính xác phân lớp cao, yêu cầu kích thước bộ dữ liệu huấn luyện nhỏ, dễ áp dụng cho nhiều bài toán.
- Hiệu quả với các bài toán phân lớp dữ liệu có số chiều lớn.
- Hiệu quả với các trường hợp số chiều dữ liệu lớn hơn số lượng mẫu.

Tuy nhiên, bộ phân lớp SVM còn có một số nhược điểm:

- Thời gian huấn luyện lâu, không gian bộ nhớ sử dụng lớn, được thiết kế cho phân lớp nhị phân (trong khi thực tế chủ yếu là phân loại đa lớp).
- Có thể bị overfit trên dữ liệu huấn luyện, nhạy cảm với nhiễu.

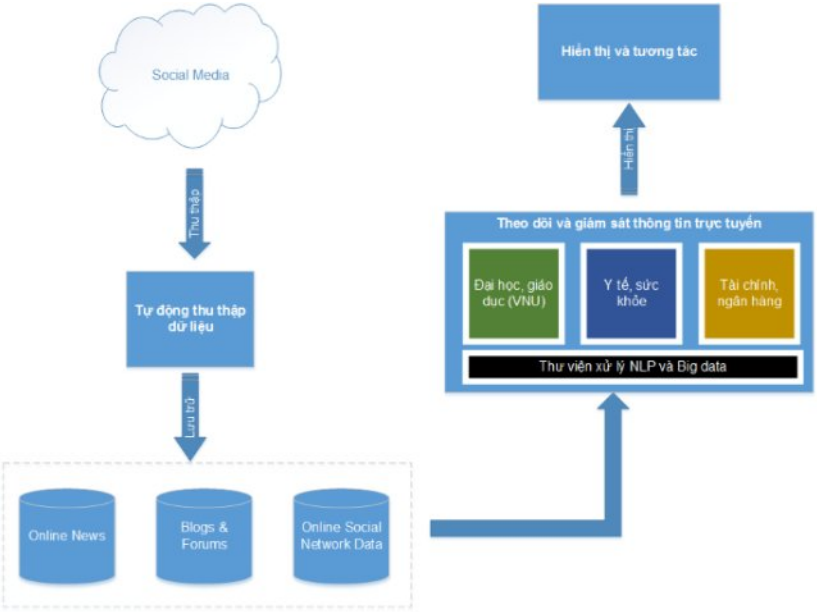
1.5 Mô hình phân lớp Maximum Entropy

Với những nhược điểm của hai bộ phân lớp trên, bộ phân lớp theo nguyên lý entropy cực đại ra đời, giải quyết tương đối tốt các bài toán phân lớp dữ liệu dạng văn bản. Trong chương 3, chúng tôi sẽ trình bày chi tiết về bộ phân lớp này cũng như cách ứng dụng vào trong bài toán phân lớp quan điểm cho dữ liệu văn bản.

Tổng quan hệ thống VNU-SMM

2.1 Kiến trúc tổng thể của hệ thống

Hệ thống VNU-SMM được thiết kế với kiến trúc tổng quan như trong hình 2.1:



Hình 2.1: Thiết kế tổng quan của hệ thống VNU-SMM

Hệ thống cần thu thập, lưu trữ và xử lý, phân tích một lượng thông tin khổng lồ từ các kênh truyền thông xã hội với yêu cầu xử lý nhanh, kịp thời nên thiết kế của hệ thống cần đảm bảo được các yêu cầu này. Về công nghệ, hệ thống được tích hợp và cài đặt nhiều công nghệ hiện đại về điện toán đám mây và xử lý dữ liệu lớn. Thêm vào đó, hệ thống cũng được thiết kế theo kiến trúc mở, phục vụ việc linh động

trong mở rộng ứng dụng của hệ thống ra nhiều lĩnh vực khác ngoài giáo dục như y tế, sức khỏe hay tài chính, ngân hàng.

Từ Hình 2.1, ta có thể thấy hệ thống VNU-SMM được thiết kế với ba khối chức năng chính: khối chức năng tự động thu thập dữ liệu, khối chức năng theo dõi và giám sát thông tin trực tuyến và khối hiển thị, giao diện tương tác với người sử dụng.

2.1.1 Khối chức năng tự động thu thập dữ liệu

Khối chức năng tự động thu thập dữ liệu có các chức năng chính như: tự động thu thập dữ liệu từ các kênh truyền thông xã hội như facebook, twitter, các blog, forums. Sau đó, tiền xử lý dữ liệu (data preprocessing) để chuẩn hóa và làm sạch thông tin. Dữ liệu sau khi được chuẩn hóa và làm sạch sẽ được hệ thống lưu vào cơ sở dữ liệu, đồng thời tự động đánh chỉ mục phục vụ việc truy xuất dữ liệu nhanh chóng khi cần sử dụng. Ngoài ra, khối chức năng này còn thực hiện nhiệm vụ phân tích sơ bộ dữ liệu (data shallow analysis).

2.1.2 Khối chức năng lõi với chức năng theo dõi và giám sát thông tin trực tuyến

Khối chức năng tự động theo dõi và giám sát thông tin trực tuyến là khối chức năng lõi của hệ thống. Khối chức năng thực hiện các nhiệm vụ: phân loại, phân lớp, thống kê và tổng hợp thông tin, phân tích và so sánh thương hiệu, phân tích các khía cạnh, phân tích và so sánh, phân tích bình luận/quan điểm, phân tích ý kiến góp ý và phân tích xu hướng.

2.1.3 Khối hiển thị, giao diện tương tác với người dùng cuối

Khối giao diện hiển thị, tương tác có chức năng cung cấp cho người sử dụng cuối một giao diện trực quan, sinh động cho từng nội

dung là kết quả của các bước phân tích nói trên. Người sử dụng có thể theo dõi thông tin cập nhật theo thời gian thực, khi có dữ liệu mới cập nhật, đồng thời có thể thực hiện các thao tác tìm kiếm, so sánh, thống kê, v.v đối với các dữ liệu đã thu thập được.

2.2 Thu thập và gán nhãn dữ liệu

Dữ liệu của chúng tôi thu được hệ thống gồm 9353 câu, trong đó có 2812 câu là positive, 2662 câu là negative và 3879 câu là gán nhãn other.

2.3 Phân lớp quan điểm

Thành phần phân lớp quan điểm thuộc khối chức năng lõi với khả năng tự động phân lớp quan điểm theo thời gian khi có dữ liệu mới thu thập được. Chi tiết về cách cài đặt bộ phân lớp theo mô hình entropy cực đại sẽ được chúng tôi trình bày chi tiết trong chương 4 của luận văn.

Bộ phân lớp Maximum Entropy

3.1. Tổng quan về entropy cực đại

Trong mục này, chúng tôi sẽ giới thiệu về khái niệm entropy cực đại thông qua một ví dụ đơn giản. Giả sử chúng ta cần mô hình hóa lại các quyết định của một chuyên gia khi phân lớp chủ đề cho một bài báo. Mô hình *p* gán cho mỗi phân lớp *f* một giá trị xấp xỉ *p(f)* là xác suất mà chuyên gia sẽ chọn *f* là phân lớp của bài báo. Để có thể xây dựng được mô hình *p*, chúng ta trước tiên cần thu thập một lượng lớn các mẫu lựa chọn phân lớp của chuyên gia. Mục tiêu của chúng ta là (1) trích xuất các dữ liệu thực về quá trình ra quyết định từ tập mẫu thu thập được và (2) xây dựng mô hình *p* cho quá trình ra quyết định này.

3.2. Entropy là gì?

Ta có định nghĩa về Entropy do Shannon đưa ra vào năm 1948:

Với một tập hợp các xác suất $\mathrm{P} = \{p_1, p_2, ..., p_n\}$ ta có entropy của P được định nghĩa như sau:

$$H(P) = -\sum_{i\ 1}^{n} p_i \log p_i \qquad (3.3)$$

3.3.1. Các ràng buộc và đặc trưng

Trong mô hình entropy cực đại, chúng ta sử dụng các tập mẫu huấn luyện (training data) để sinh ra các ràng buộc cho phân phối điều kiện. Mỗi ràng buộc thể hiện một đặc trưng của tập mẫu mà phân phối đã học cần có. Phân phối sau khi học xong phải thỏa mãn tất cả các ràng buộc sinh ra từ tập mẫu, ngoài ra không cho thêm bất kì giả thiết nào khác.

Các hàm đặc trưng $f(x, y)$ (còn gọi tắt là đặc trưng) là một hàm nhị phân với 2 tham số: $y \in$ tập các lớp cần phân loại và $x \in$ tập các ngữ cảnh:

$$f = \varepsilon \to \{0,1\}$$

Việc chúng ta lựa chọn các hàm đặc trưng là tùy thuộc vào từng bài toán khác nhau và cách lựa chọn đặc trưng sẽ ảnh hưởng đến chất lượng của bộ phân lớp.

3.3.2. Nguyên lý Entropy cực đại

Nguyên lý Entropy cực đại cho rằng: *Với một tập các dữ liệu đã biết trước, phân phối xác suất tốt nhất trong tập các phân phối xác suất có thể để biểu diễn trạng thái hiện tại của tri thức, là phân phối xác suất có entropy cực đại và phân phối này là duy nhất.*

Ta có thể tóm tắt ý tưởng, bản chất của nguyên lý entropy cực đại như sau: Nguyên lý entropy cực đại không giả thiết bất cứ điều gì về phân phối xác suất ngoài những gì quan sát được từ tập dữ liệu, đồng thời luôn chọn phân phối xác suất đồng đều nhất phù hợp với các ràng buộc quan sát được này.

3.3.3. Dạng tham số

Bài toán đặt ra theo nguyên lý entropy cực đại có dạng: tìm *p** thuộc C sao cho entropy là lớn nhất. Bài toán có thể dễ dàng được giải quyết khi số ràng buộc là ít và đơn giản, tuy nhiên, trong thực tế số các ràng buộc tăng lên và chồng chéo nhau như trong ví dụ ở mục 2.1 thì ta cần một hướng giải quyết hiệu quả hơn.

Để giải quyết vấn đề này, chúng ta có thể áp dụng phương pháp thừa số Lagrange.

3.3.4. Tính toán các tham số

Có nhiều phương pháp số học được sử dụng, có thể kể đến như IIS (Improved Iterative Scaling), L-BFGS, GIS (Generalized Iterative Scaling). Trong phần này, chúng tôi sẽ giới thiệu tổng quan về hai phương pháp phổ biến và tốt nhất hiện nay cho bộ phân lớp dựa trên mô hình entropy cực đại: IIS và L-BFGS .

1) Phương pháp Improved Iterative Scaling

Phương pháp này được hai nhà khoa học Darroch và Ratcliff giới thiệu vào năm 1972 để tính toán các xấp xỉ cực đại likelihood cho các tham số của các mô hình hàm mũ (exponential model). Thuật toán này được áp dụng với điều kiện các hàm đặc trưng $f_i(x, y)$ không âm:

$$f_i(x, y) \geq 0 \quad \forall \mathrm{x, y, i}$$

Trong bài toán phân lớp chúng ta đang giải quyết, điều kiện này hiển nhiên thỏa mãn do các hàm đặc trưng là các hàm nhị phân. Nội dung của thuật toán được trình bày như sau:

Input: Các hàm đặc trưng $f_i(x, y)$ và phân phối thực nghiệm $\tilde{p}(x, y)$

Output: Các tham số tối ưu λ_i* và mô hình tối ưu p_{λ^*}

Bước 1: Bắt đầu với λ_i = 0 với mọi i ∈{1,2,…,n}

Bước 2: Với mỗi i thực hiện:

a. Gọi $\Delta\lambda_i$ là nghiệm của phương trình:

$$\sum_{x,y} \tilde{p}(x)\, p(y \mid x) f_i(x, y) \exp(\Delta\lambda_i f^{\#}(x, y)) = \tilde{p}(f_i) \qquad (3.13)$$

Trong đó: $f^{\#}(x,y) \quad \sum_{i\ 1}^{n} f_i(x,y)$

b. Cập nhật lại giá trị của λ_i theo công thức: $\lambda_i = \lambda_i + \Delta\lambda_i$

Bước 3: Quay lại bước 2 nếu như tất cả các λ_i đều chưa hội tụ.

2) Phương pháp L-BFGS (Limited-memory BFGS)

L-BFGS là một thuật toán tối ưu trong họ các phương pháp quasi-Newton cho phép xấp xỉ thuật toán BFGS gốc sử dụng bộ nhớ giới hạn của máy tính. Để hiểu rõ phương pháp này, chúng tôi sẽ giới thiệu tổng quan về phương pháp Newton và phương pháp Quasi-Newton trước khi giới thiệu về thuật toán L-BFGS

a. Phương pháp Newton

Hầu hết các phương pháp tối ưu số học là các giải thuật lặp trong đó ta thử dần các giá trị của biến cần tìm, hội tụ dần về giá trị tối ưu của hàm số đã cho. Hay nói cách khác, với hàm số x^* $\arg\max f(x)$, giả sử ta có một giá trị xấp xỉ x_n , ta mong muốn giá trị thử tiếp theo là x_{n+1} thỏa mãn: $f(x_n) < f(x_{n+1})$.

Phương pháp Newton tập trung vào xấp xỉ bậc 2 của hàm số cho các điểm xung quanh x_n. Giả sử hàm số f là khả vi hai lần (twice-differentiable), chúng ta có thể sử dụng xấp xỉ bậc 2 của hàm f cho các điểm 'gần' một điểm cố định bằng khai triển Taylor. Xấp xỉ này đúng với giá trị Δx tiến dần tới 0.

Ta có giải thuật lặp NewtonRapshon như sau:

NewtonRaphson(f, x_0) :
  For $n = 0, 1, \ldots$ (until converged) :
    Compute $\mathbf{g}_n$ and $\mathbf{H}_n^{-1}$ for x_n
    $d = \mathbf{H}_n^{-1}\mathbf{g}_n$
    $\alpha = \min_{\alpha \geq 0} f(x_n - \alpha d)$
    $x_{n+1} \leftarrow x_n - \alpha d$

Hình 3.1: Giải thuật lặp NewtonRapshon

Giải thuật trên có thể được chứng minh luôn hội tụ tới điểm tối ưu cho hàm f cực đại nếu f là một hàm số lõm hay hội tụ tới f cực tiểu nếu f là hàm lồi với lựa chọn x_0 bất kỳ.

Trong thực tế với các bài toán học máy như chúng ta đang quan tâm, *f* thường là một hàm số nhiều chiều với số chiều tương ứng với số tham số của mô hình học. Số tham số này thường rất lớn, có thể lên tới hàng trăm triệu hoặc thậm chí hàng tỉ, điều này khiến cho việc thực hiện tính toán theo phương pháp Newton là không thể do không thể tính được ma trận Hessian hay nghịch đảo của nó. Chính vì vậy, trong thực tế, giải thuật NewtonRapshon rất ít khi được sử dụng với các bài toán lớn. Tuy nhiên, thuật toán trên vẫn đúng với ma trận Hessian xấp xỉ đủ tốt mà không cần chính xác tuyệt đối. Phương pháp được sử dụng để xấp xỉ ma trận Hessian này là Quasi-Newton.

b. Quasi-Newton

Phương pháp Quasi-Newton sử dụng một hàm QuasiUpdate để sinh ra ma trận Hessian nghịch đảo tại x_{n+1} dựa trên ma trận Hessian nghịch đảo tại x_n.

Ở đây, chúng ta giả sử rằng phương thức QuasiUpdate chỉ cần ma trận nghịch đảo tại điểm liền trước đó, độ lệch giữa 2 điểm và độ lệch gradient của chúng.

Bốn nhà nghiên cứu Broyden, Fletcher, Goldfarb và Shanno đã tìm ra phương thức tính xấp xỉ ma trận Hessian nghịch đảo H_n^{-1} mà ta gọi là phương thức BFGS Update.

Ta chỉ cần sử dụng phương thức này ứng dụng vào trong phương thức QuasiNewton ở trên để xấp xỉ tham số.

Xấp xỉ BFGS Quasi-Newton có ưu điểm là không cần chúng ta phải tính toán ra ma trận Hessian của hàm số f mà thay vào đó, ta có thể liên tục cập nhật các giá trị xấp xỉ của nó. Tuy nhiên, chúng ta vẫn cần phải lưu lại lịch sử của các vector s_n và y_n trong mỗi vòng lặp. Nếu vấn đề cốt lõi của phương pháp NewtonRapshon là bộ nhớ cần thiết để tính toán ma trận nghịch đảo Hessian là quá lớn thì phương pháp BFGS Quasi-Newton chưa giải quyết được vấn đề này do bộ nhớ liên tục tăng không có giới hạn. Chính vì lẽ đó, phương pháp L-BFGS ra đời với ý tưởng chỉ sử dụng m giá trị s_k và y_k gần nhất để tính toán hàm update BFGS thay vì toàn bộ số lượng vector. Việc này giúp cho bộ nhớ luôn là hữu hạn.

Kết quả thử nghiệm và đánh giá

4.1. Tiến hành thử nghiệm

- **Bước 1:** Tự động thu thập dữ liệu từ các trang mạng trực tuyến: baomoi.com, vnexpress.net và dantri.com.vn.
- **Bước 2:** Tiền xử lý dữ liệu thu thập được: làm sạch và chuẩn hóa dữ liệu, gán nhãn loại từ cho từng câu bình luận.
- **Bước 3:** Nhận dạng thủ công từng câu trong bộ dữ liệu mẫu và phân vào các lớp *positive* (tích cực), *negative* (tiêu cực) và *other* (khác)
- **Bước 4:** Tách 1832 câu trong bộ dữ liệu đã gán nhãn thành bộ test và 7521 câu còn lại là bộ huấn luyện.
- **Bước 5:** Chạy bộ phân lớp và so sánh kết quả phân lớp tự động so với kết quả phân lớp thủ công.

4.2. Tiền xử lý dữ liệu

Dữ liệu sau khi được crawl tự động về sẽ được đưa qua bộ tiền xử lý dữ liệu trước khi đưa vào nhận dạng thủ công. Bộ tiền xử lý là JvnTextPro do các tác giả của trường Đại học Công nghệ phát triển.

4.3. Xây dựng mô hình

4.3.1. Lựa chọn đặc trưng

Như ta đã biết từ nội dung chương 2, các hàm đặc trưng f gồm hai tham số: ngữ cảnh và nhãn phân lớp.

Các hàm đặc trưng được xác định theo quy tắc sau:

- **Bước 1**: Tìm tất cả unigram, bigram của từng câu hay từng quan sát (observation).

➢ **Bước 2**: Sắp xếp danh sách các unigram và bigram thu được theo thứ tự giảm dần của loại từ (ưu tiên các tính từ, rồi đến danh từ, rồi đến động từ, rồi đến các loại từ khác).

➢ **Bước 3**: Lấy top 50 của danh sách sau khi sắp xếp làm đặc trưng cho câu hay quan sát đó.

4.3.2. Cài đặt thuật toán học

Chúng tôi cài đặt bộ phân lớp sử dụng hệ điều hành windows 10 và ngôn ngữ lập trình Java với công cụ lập trình Eclipse.

Hệ thống cài đặt thuật toán học ME sử dụng phương pháp L-BFGS để xấp xỉ tham số cho mô hình.

4.4. Kết quả thử nghiệm

4.4.1. Các chỉ số đo kiểm chất lượng bộ phân lớp

Hệ thống được đánh giá dựa trên bộ ba tiêu chí đánh giá sau: Độ chính xác (precision), độ bao phủ (recall) và F_1.

4.4.2. Kết quả thực nghiệm bài toán phân lớp mức độ câu

Kết quả phân loại với tập kiểm tra được thể hiện trong Bảng 4.1:

Bảng 4.1: Kết quả thực nghiệm bài toán phân lớp mức độ câu sử dụng ME

	Số thực thể	**Nhận dạng được**	**Nhận dạng đúng**	**Độ chính xác (%)**	**Độ bao phủ (%)**	**F_1 (%)**
	(1)	**(2)**	**(3)**	**(4)=(3)/(2)**	**(5)=(3)/(1)**	**(6)=2.(4)x(5)/((4)+(5))**
Positive	555	543	325	59.85	58.56	59.20
Negative	514	530	309	58.30	60.12	59.20
Other	763	759	460	60.61	60.29	60.45
All	1832	1832	1094	59.72	59.72	59.72

Từ bảng kết quả trên chúng ta có thể thấy, kết quả của bộ phân lớp tính theo tiêu chí độ chính xác của các nhãn positive, negative, other lần lượt là 59.85%, 58.30% và 60.61%. Các giá trị này xấp xỉ với kết quả tính theo độ bao phủ, lần lượt là 58.56%, 60.12% và 60.29%. Điều này cho thấy, bộ phân lớp tương đối ổn định khi đánh giá theo hai tiêu chí trên, kết quả là giá trị F1 theo từng nhãn cũng xấp xỉ nhau. Kết quả tính theo tiêu chí F1 đạt 59.72% nếu tính theo tổng toàn bộ nhãn của chương trình.

4.5. So sánh với bộ phân lớp Naïve Bayes

Để so sánh, chúng tôi cũng đã cài đặt bộ phân lớp Naïve Bayes và đánh giá trên cùng tập dữ liệu huấn luyện và kiểm tra như trên. Chúng tôi đã sử dụng thư viện mã nguồn mở để cài đặt và kiểm tra bộ phân lớp Naïve Bayes[1].

Kết quả cụ thể như trong Bảng 4.2.

Bảng 4.2: Kết quả thực nghiệm bài toán với bộ phân lớp Naïve Bayes

	Số thực thể	**Nhận dạng được**	**Nhận dạng đúng**	**Độ chính xác (%)**	**Độ bao phủ (%)**	**F_1 (%)**
	(1)	**(2)**	**(3)**	**(4)=(3)/(2)**	**(5)=(3)/(1)**	**(6)=2.(4)x(5)/((4)+(5))**
Positive	555	348	214	61.49	38.56	61.49
Negative	514	463	262	56.59	50.97	56.59
Other	763	1021	543	53.18	71.17	53.18
All	1832	1832	1019	55.62	55.62	55.62

Từ bảng kết quả trên, chúng ta có độ chính xác của ba nhãn positive, negative và other lần lượt là 61.49%, 56.59% và 53.18%. So sánh với độ bao phủ, ta thấy có sự chênh lệch lớn (38.56%, 50,97% và

[1] https://github.com/datumbox/NaiveBayesClassifier

55.62%) và đồng thời kết quả đo theo tiêu chí F1 đạt 55.62%, thấp hơn so với bộ phân lớp Maximum entropy. Sự không ổn định trong phân loại của bộ phân lớp Naïve Bayes có thể dẫn đến hiệu quả phân lớp rất khác nhau đối với các bộ dữ liệu khác nhau.

4.6. Đánh giá kết quả

Mặc dù bộ phân lớp Maximum entropy cho kết quả cao hơn so với bộ phân lớp sử Naïve Bayes, kết quả đạt được chưa cao (~60%). Kết quả này có thể do một số nguyên nhân sau:

+ Tập dữ liệu sử dụng để huấn luyện và kiểm tra gán nhãn còn chưa chính xác: bộ dữ liệu này sau khi được crawl về và chạy qua bộ tiền xử lý (lọc bỏ stopword, dấu câu, chữ số; đưa về dạng chữ viết thường (lowercase); phân tách từ và thực hiện pos tagging) đã được phân loại và gán nhãn bằng tay theo phương pháp crowdsourcing do khối lượng câu cần phân loại lớn. Điều này dẫn đến những bất thường và khó kiểm soát trong chất lượng nguồn dữ liệu.

+ Các đặc trưng lựa chọn chưa thực sự hiệu quả: đối với các thuật toán học máy có giám sát, việc chọn lựa được các đặc trưng hiệu quả là điểm mấu chốt quyết định đến chất lượng của cả bộ phân lớp. Trong hệ thống, chúng tôi đã sử dụng các đặc trưng phổ biến cho các bộ phân lớp chủ đề truyền thống (unigram và bigram), Part-of-speech (POS) của từng từ, đồng thời kết hợp với sử dụng các đặc trưng riêng của bài toán phân lớp quan điểm như sử dụng từ điển các từ và cụm từ mang quan điểm (sentiment words and phrases) để tăng độ chính xác cho bộ phân lớp. Tuy nhiên, các đặc trưng được lựa chọn vẫn còn mang tính kinh nghiệm và đánh giá qua thực tế nên kết quả chưa được cao.

Tổng kết và hướng phát triển tiếp theo

Luận văn đã nghiên cứu và tìm hiểu về bài toán phân lớp quan điểm với dữ liệu là các comment, phản hồi, các góp ý từ các kênh truyền thông xã hội phổ biến, đánh giá thuật toán học maximum entropy với dữ liệu thực tế trong chủ đề giáo dục. Các kết quả chính mà luận văn đạt được như sau:

- Tìm hiểu, giới thiệu và đánh giá sơ bộ một số thuật toán học có giám sát ứng dụng trong xây dựng bộ phân lớp văn bản nói chung và phân lớp quan điểm người dùng nói riêng: thuật toán Naïve Bayes, SVM và Maximum Entropy.
- Giới thiệu và đi sâu vào thuật toán Maximum Entropy và cách ứng dụng trong hệ thống phân lớp quan điểm người dùng.
- Thử nghiệm với dữ liệu thật thu được từ các kênh truyền thông xã hội.

Tuy đã cố gắng nâng cao chất lượng bộ phân lớp, nhưng kết quả thử nghiệm với mức câu còn chưa cao (~60%) do một số nguyên nhân cả về khách quan và chủ quan, trong đó nguyên nhân chủ yếu do chất lượng của bộ dữ liệu huấn luyện và kiểm tra còn thấp, chưa đồng bộ, các đặc trưng được lựa chọn chưa hiệu quả. Trong tương lai, để cải tiến hiệu năng của bộ phân lớp, chúng tôi có thể giảm số lượng các câu trong tập huấn luyện để có thể tập trung nâng cao chất lượng gán nhãn của tập này. Bên cạnh đó, để nâng cao chất lượng của các đặc trưng, chúng tôi đề xuất sử dụng thêm các kiến thức chuyên gia về ngôn ngữ và hiểu biết về các lĩnh vực cụ thể để có thể tránh được các trường hợp phân lớp sai cơ bản nếu chỉ dựa vào việc đếm các từ trong câu. Ví dụ như chúng tôi có thể phân biệt các câu điều kiện để xử lý riêng, các câu ghép có sự so

sánh, thay đổi về quan điểm để xử lý riêng, v.v. Ngoài ra, như đã trình bày trong chương 1, chúng tôi cũng cân nhắc một hướng nghiên cứu khả thi và rất có tiềm năng để tăng độ chính xác của bộ phân lớp là nghiên cứu và cài đặt phương pháp học máy deep learning cho bộ phân lớp.