**MỤC LỤC**

[LỜI MỞ ĐẦU 2](#_Toc501573995)

[CHƯƠNG 1: GIỚI THIỆU VỀ NGÔN NGỮ JAVA 3](#_Toc501573996)

[1.1 Lịch sử Java 3](#_Toc501573997)

[1.2 Java là gì ? 4](#_Toc501573998)

[1.3 Các đặc trưng cơ bản của Java 4](#_Toc501573999)

[1.4 Các kiểu ứng dụng Java 6](#_Toc501574000)

[1.4.1 Ứng dụng Applets 6](#_Toc501574001)

[1.4.2 Ứng dụng dòng lệnh (console) 6](#_Toc501574002)

[1.4.3 Ứng dụng đồ họa 7](#_Toc501574003)

[1.4.4 JSP/Servlet 7](#_Toc501574004)

[1.4.5 Ứng dụng cơ sở dữ liệu 7](#_Toc501574005)

[1.4.6 Ứng dụng mạng 7](#_Toc501574006)

[1.4.7 Ứng dụng nhiều tầng 7](#_Toc501574007)

[1.4.8 Ứng dụng cho các thiết bị di động 7](#_Toc501574008)

[1.5 giới thiệu về máy ảo Java (JVM – Java Virtual Machine) 8](#_Toc501574009)

[CHƯƠNG 2. BÀI TOÁN PHÂN LOẠI VĂN BẢN VÀ THUẬT TOÁN SỬ DỤNG 10](#_Toc501574010)

[2.1. Giới thiệu 10](#_Toc501574011)

[2.2. Phát biểu bài toán 11](#_Toc501574012)

[2.3. Mô hình tổng quát 11](#_Toc501574013)

[2.4. Môi trường phát triển 13](#_Toc501574014)

[2.5. Thuật toán Naives Bayes 14](#_Toc501574015)

[2.6. Tf-idf 15](#_Toc501574016)

[Tf- term frequency 15](#_Toc501574017)

[IDF- Inverse Document Frequency 15](#_Toc501574018)

[2.7. Cơ sở lý thuyết 15](#_Toc501574019)

[2.7.1. Tiền xử lý văn bản 15](#_Toc501574020)

[2.7.2. Phương pháp biểu diễn văn bản 16](#_Toc501574021)

[2.7.3. Mô hình không gian vector 16](#_Toc501574022)

[2.7.4. Khái niệm trọng số 16](#_Toc501574023)

[CHƯƠNG 3. GIAO DIỆN ỨNG DỤNG DEMO 18](#_Toc501574024)

[3.1. Giao diện form chính 18](#_Toc501574025)

[3.2. Giao diện form kiểm tra 19](#_Toc501574026)

[3.3. Giao diện form huấn luyện 20](#_Toc501574027)

# LỜI MỞ ĐẦU

Phân loại văn bản là một vấn đề quan trọng trong lĩnh vực xử lý ngôn ngữ. Nhiệm vụ của bài toán này là gán các tài liệu văn bản vào nhóm các chủ đề cho trước. Đây là một bài toán rất thường gặp trong thực tế điển hình như : một nhà chuyên phân tích thị thường chứng khoán, anh ta cần phải tổng hợp rất nhiều tài liệu, bài viết về thị trường chứng khoán để đọc và đưa ra phán đoán của mình. Tuy nhiên, anh ta không thể đọc tất cả các bài viết, bài báo hay các tài liệu để rồi phân loại chúng đâu là tài liệu chứng khoán sau đó anh ta mới đọc kỹ chúng cho mục đích của anh ta. Lý do của vấn đề này là bởi ví số lượng bào viết, bài báo hiện nay rất nhiều, đặc biệt là trên internet, nếu để đọc hết được tất cả tài liệu đó thì sẽ mất rất nhiều thời gian. Một ví dụ khác trong thực tế là việc phân loại spam mail. Khi một mail được gửi đến hộp thư, nếu để người dùng phải đọc tất cả các mail thì sẽ tốn rất nhiều thời gian vì spam mail rất nhiều. Vì vậy, cần có một hệ thống phân loại đâu là spam mail và đâu là mail tốt.

Để giải bài toán này đã có rất nhiều phương pháp được đưa ra như : thuật toán Naïve Bayes, K-NN (K-Nearest-Neighbor), Cây quyết định (Decision Tree), Mạng Neuron nhân tạo (Artificial Neural Network) và SVM (Support Vector Machine). Mỗi phương pháp đều cho kết quả khá tốt cho bài toán này, tuy nhiên phương pháp phân loại văn bản bằng thuật toán Bayes được sử dụng phổ biến hơn cả và dễ dàng cài đặt. Chính vì vậy chúng em lựa chọn đề tài: “**Phân loại bài báo sử dụng thuật toán Naïve Bayes**“ làm đề tài kết thúc môn học của mình.

Chúng em xin chân thành cảm ơn các thầy, cô bộ môn đã tận tình giảng dạy em trong suốt thời gian học tập vừa qua. Nhờ có sự chỉ dạy tận tình của thầy Cù Việt Dũng trực tiếp hướng dẫn giảng dạy, cùng sự đào tạo của các thầy cô bộ môn khác trong thời gian qua giúp chúng em hoàn thành bài báo cáo này. Do đây là lần đầu tiên triển khai một hệ thống có tính thực tiễn cao, nên quá trình triển khai có thể còn nhiều sai sót. Mong các thầy cô đóng góp ý kiến giúp chúng em hiểu rõ hơn về bài toán.

Một lần nữa chúng em xin chân thành cảm ơn các thầy cô!

# CHƯƠNG 1: GIỚI THIỆU VỀ NGÔN NGỮ JAVA

## 1.1 Lịch sử Java

Java là một ngôn ngữ lập trình được Sun Microsystems giới thiệu vào tháng 6 năm 1995. Từ đó, nó đã trở thành 1 công cụ lậptrình của các lập trình viên chuyên nghiệp.

Java được xây dựng trên nền tảng của C và C++. Do vậy nó sử dụng các cú pháp của C và các đặc trưng hướng đối tượng của C++.

Ban đầu Java đượcc thiết kế làm ngôn ngữ viết chương trình cho các sản phẩm điện tử dân dụng như đầu video, tivi, điện thoại, máy nhắn tin... Tuy nhiên với sự mạnh mẽ của Java đã khiến nó nổi tiếng đến mức vượt ra ngoài sự tưởng tượng của các nhà thiết kế ra nó.

Java khởi thủy tên là Oak- là cây sồi mọc ở phía sau văn phòng của nhà thiết kế chính ông Jame Gosling, sau này ông thấy rằng đã có ngôn ngữ lập trình tên Oak rồi, do vậy nhóm thiết kế quyết định đổi tên, “Java” là cái tên được chọn, Java là tên của một quán cafe mà nhóm thiết kế java hay đến đó uống.

## 1.2 Java là gì ?

Java là ngôn ngữ lập trình hướng đối tượng, do vậy không thể dùng Java để biết 1 chương trình hướng chức năng. Java có thể giải quyết hầu hết các công việc mà các ngôn ngữ khác có thể làm được.

Java là ngôn ngữ vừa biên dịch vừa thông dịch. Đầu tiên mã nguồn được biên dịch bằng công cụ JAVAC để chuyển thành dạng ByteCode. Sau đó được thực thi trên từng loại máy cụ thể nhờ chương trình thông dịch JAVA. Mục tiêu của các nhà thiết kế Java là cho phép người lập trình viết chương trình một lần nhưng có thể chạy tốt trên bất cứ phần cứng cụ thể, thế nên khẩu hiệu của các nhà thiết kế Java là ‘Write One, Run Any Where’.

Ngày nay, Java được sử dụng rộng tãi để viết chương trình chạy trên internet. Nó là ngôn ngữ lập trình hướng đối tượng độc lập thiết bị, không phụ thuộc vào hệ điều hành. Java không chỉ dùng để viết các ứng dụng chạy đơn lẻ hay trong mạng mà còn để xây dựng các chương trình điều khiển điện thoại di động, PDA…

## 1.3 Các đặc trưng cơ bản của Java

**Đơn giản**

Những người thiết kế mong muốn phát triển một ngôn dữ dễ học và quen thuộc với đa số người lập trình. Java tựa như C++, nhưng đã lược bỏ đi các đặc trưng phức tạp, không cần thiết của C, C++ như : thao tác con trỏ, thao tác định nghĩa chồng toán tử,… Java không sử dụng lệnh ‘goto’ cũng như file header(.h). Cấu trúc ‘struct’ và ‘union’ cũng đã được loại bỏ khỏi Java. Nên có người bảo Java là ‘C++--‘, ngụ ý bảo java là C++ nhưng đã bỏ đi những thứ phức tạp không cần thiết.

**Hướng đối tượng**

Có thể nói java là ngôn ngữ lập trình hoàn toàn hướng đối tượng, tất cả trong java đều là sự thật, đâu đâu cũng là sự thật.

**Độc lập với nền**

Mục tiêu chính của các nhà thiết kế java là độc lập với hệ nền hay còn gọi là độc lập phần cứng và hệ điều hành. Đây là khả năng một chương trình được viết tại 1 máy nhưng có thể chạy được bất kỳ đâu.

Tính độc lập với phần cứng được hiểu theo nghĩa một chương trình Java nếu chạy đúng trên phần cứng của 1 họ máy nào đó thì cũng chạy đúng trên tất cả các họ máy khác. Một chương trình chỉ chạy đúng trên 1 số họ máy cụ thể được gọi là phụ thuộc vào phần cứng.

Tính độc lập với hệ điều hành được hiểu theo nghĩa một chương trình java có thể chạy được trên tất cả các hệ điều hành. Một chương trình chỉ chạy được trên một số hệ điều hành được gọi là phụ thuộc hệ điều hành.

Các chương trình viết bằng java có thể chạy trên hầu hết các hệ nền mà không cần phải thay đổi gì, điều này đã được những người lập trình đặt cho nó 1 khẩu hiệu : ‘Write One, Run Any Where’, điều này là không thể có với những ngôn ngữ lập trình khác

Đối với các chương trình viết bằng C, C++ hoặc một ngôn ngữ nào khác, trình biên dịch sẽ chuyển tập lệnh thành mã máy, hay lệnh của bộ vi xử lý. Những lệnh này phụ thuộc vào CPU hiện tại trên máy bạn. Nên khi muốn chạy trên loại CPU khác chúng ta phải biên dịch lại chương trình.

**Mạnh mẽ**

Java là ngôn ngữ yêu cầu chặt chẽ về kiểu dữ liệu, việc ép kiểu tư động bừa bãi của C, C++ nay được hạn chế trong Java, điều này làm chương trình rõ ràng, sáng sủa, ít lỗi hơn. Java kiểm tra lúc biên dịch và cả trong thời gian thông dịch vì vậy Java loại bỏ một số loại lỗi lập trình nhất định. Java không sử dụng con trỏ và các phép toán con trỏ. Java kiểm tra tất cả các truy cập đến mảng, chuỗi khi thực thi để đảm bảo rằng các truy cập đó không ra ngoài giới hạn kích thước.

Trong các môi trường lập trình truyền thống, lập trình viên phải tự mình cấp phát bộ nhớ. Trước khi chương trình kết thúc thì phải tự giải phóng bộ nhớ đã cấp. Vấn đề này sinh khi lập trình viên quên giải phóng bộ nhớ đã xin cấp trước đó. Trong chương trình Java, lập trình viên không phải bận tâm đến việc cấp phát bộ nhớ. Qúa trình cấp phát, giải phóng được thực hiện tự động, nhờ dịch vụ thu nhặt những đối tượng không còn sử dục nữa.

Cơ chế bẫy lỗi của java giúp đơn giản hóa quá trình xử lý lỗi và hồi phục sau lỗi.

**Hỗ trợ lập trình đa tuyến**

Đây là tính năng cho phép viết 1 chương trình có nhiều đoạn mã lệnh được chạy song song với nhau. Với java ta cố thể viết các chương trình có khả năng chạy song song một các dễ dàng, hơn thế nữa việc đồng bộ tài nguyên dùng chung trong java cũng rất đơn giản. Điều này là không thể với một số ngôn ngữ lập trình khác như C, ++, pascal…

**Phân tán**

Java hỗ trợ đầy đủ các mô hình tính toán phân tán : mô hình client/ server, gọi thủ tục từ xa…

**Hỗ trợ internet**

Mục tiêu quan trọng của các nhà thiết kế java là tạo điều kiện cho các nhà phát triển ứng dụng có thể viết các chương trình ứng dụng client và web một cách dễ dàng, với java ta có thể viết các chương trình sử dụng các giao thức TCP, UDP một cách dễ dàng, về lập trình web phía máy khách java có công nghệ java applet, về lập trình web phía máy khác java có công nghệ servlet/JSP, về lập trình phân tán java có công nghệ RMI, CORBA, EJB, Web Service.

**Thông dịch**

Các chương trình java cần được thông dịch trước khi chạy, một chương trình java được biên dịch thành mã byte code mã độc lập với hệ nền, chương trình thông dịch java sẽ ánh xạ mã byte code này lên mỗi nền cụ thể, điều này khiến java chậm chạp đi phần nào.

## 1.4 Các kiểu ứng dụng Java

### 1.4.1 Ứng dụng Applets

Applets là chương trình java được tạo ra để sử dụng trên internet thông qua các trình duyệt hỗ trợ java như IE hay Netscape. Applets được nhúng bên trong trang web. Khi trang web hiển thị trong trình duyệt, Applets sẽ được tải về và thực thi trình duyệt.

### 1.4.2 Ứng dụng dòng lệnh (console)

Các chương trình này chạy từ dấu nhắc lệnh và không sử dụng giao diện đồ họa. Các thông tin xuất nhập được thể hiện tại dấu nhắc lệnh.

### 1.4.3 Ứng dụng đồ họa

Đây là các chương trình java chạy độc lập cho phép người dùng tương tác qua giao diện đồ họa.

### 1.4.4 JSP/Servlet

Java thích hợp để phát triển ứng dụng nhiều lớp, Applets là chương trình đồ họa chạy trên trình duyệt tại máy trạm. Ở các ứng dụng web, máy trạm gửi yêu cầu tới máy chủ. Máy chủ xử lý và gửi kết quả trở lại máy trạm. các java API chạy trên máy chủ chịu trách nghiệm xử lý tại máy chủ và trả lời máy trạm. các Java API chạy trên máy chủ này mở rộng khả năng của các ứng dụng Java API chuẩn. Các ứng dụng trên máy chủ này được gọi là các JSP/Servlet, hoặc Applets tại máy chủ. Xử lý Form của HTML là cách sử dụng đơn giản nhất của JSP/Servlet. Chúng còn có thể được dùng để xử lý dữ liệu, thực thi các giao dịch và thường được thực thi thông qua máy chủ Web.

### 1.4.5 Ứng dụng cơ sở dữ liệu

Các ứng dụng này sử dụng JDBC API để kết nối tới cơ sở dữ liệu. Chúng có thể là Applets hay ứng dụng, nhưng Applets bị giới hạn bởi tính bảo mật.

### 1.4.6 Ứng dụng mạng

Java là một ngôn ngữ rất thích hợp cho việc xây dựng các ứng dụng mạng. Với thư viện Socket bạn có thể lập trình với 2 giao thức: UDP và TCP.

### 1.4.7 Ứng dụng nhiều tầng

Với java bạn có thể xây dựng phân tán nhiều tầng với nhiều hỗ trợ khác nhau như: RMI, CORBA, EJB, Web Service.

### 1.4.8 Ứng dụng cho các thiết bị di động

Hiện nay phần lớn các thiết bị di động như điện thoại di động, máy trợ giúp cá nhân… đều hỗ trợ java. Thế nên bạn có thể xây dựng các ứng dụng chạy trên các thiết bị di động này. Đây là một kiểu ứng dụng khá hấp dẫn, bởi bì các thiết bị di động này ngày càng phổ biến và nhu cầu có các ứng dụng chạy trên đó, đặc biệt là các ứng dụng mang tính chất giải trí như game.

## 1.5 giới thiệu về máy ảo Java (JVM – Java Virtual Machine)

Máy ảo là 1 phần mềm mô phỏng 1 máy tính thật (máy tính ảo). Nó có tập hợp các lệnh logic để xác định các hoạt động của máy tính và có 1 hệ điều hành ảo. Người ta có thể xem nó như 1 máy tính thật. Nó thiết lập các lớp trừu tượng cho: phần cứng bên dưới, hệ điều hành, mã đã biên dịch.

Trình biên dịch chuyển mã nguồn thành tập các lệnh của máy ảo mà không phụ thuộc vào phần ứng và hệ điều hành cụ thể. Trình thông dịch mỗi máy sẽ chuyển tập lệnh này thành chương trình thực thi. Máy ảo tạo ra một môi trường bên trong để thực thi các lệnh bằng cách:

* Nạp các file .class
* Quản lý bộ nhớ
* Dọn rác

Việc không nhất quán của phần cứng làm cho máy ảo phải sử dụng ngăn xếp để lưu trữ các thông tin sau:

* Các “Frame” chứa các trạng thái của các phương thức
* Các toán hạng của mã byte code
* Các tham số truyền cho phương thức

Khi JVM thực thi mã, một thanh ghi cục bộ có tên: “Program Counter” được sử dụng. Thanh ghi này trỏ tới lệnh đang thực hiện. Khi cần thiết, có thể thay đổi nội dung thanh ghi để đổi hướng thực thi của chương trình. Trong trường hợp thông thường thì từng lệnh một nối tiếp nhau sẽ được thực thi.

Một khái niệm thông dụng khác trong Java là trình biên dịch “Just In Time – JIT”. Các trình duyệt thông dụng như Netscape hay IE đề có JIT bên trong để tăng tốc độ thực thi chương trình Java. Mục đích chính của JIT là chuyển tập lệnh byte code thành mã máy cụ thể cho từng loại CPU. Các lệnh này sẽ được lưu trữ và sử dụng mỗi khi gọi đến.

# CHƯƠNG 2. BÀI TOÁN PHÂN LOẠI VĂN BẢN VÀ THUẬT TOÁN SỬ DỤNG

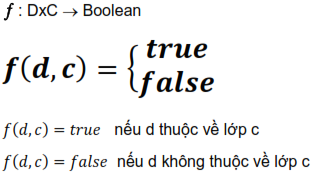
## 2.1. Giới thiệu

Ngày nay, sự bùng nổ thông tin do bị tác động bởi sự xuất hiện của các siêu phương tiện và World Wide Web (WWW) đã làm cho không gian dữ liệu gia tăng thường xuyên, điều này tạo ra một thách thức cho các hệ thống truy vấn thông tin sao cho có hiệu quả. Một trong những khó khăn mà các hệ thống thông tin thường phải gặp đó là tần suất cập nhật của các thông tin quá lớn .Phương thức sử dụng giấy trong giao dịch đang dần được số hóa, do nhiều tính năng vượt trội mà phương thức này mang lại, như là có thể lưu trữ lâu dài, cập nhật, sửa đổi, tìm kiếm một cách nhanh chóng. Do đó số lượng văn bản số hóa ngày nay đang tăng dần theo cấp số nhân, cùng với sự gia tăng của số lượng văn bản, nhu cầu tìm kiếm văn bản cũng tăng theo, khi đó phân loại văn bản tự động là một yêu cầu cấp thiết được đặt ra. Phân loại văn bản giúp sẽ giúp chúng ta tìm kiếm thông tin một cách nhanh chóng hơn thay vì phải tìm lần lượt trong từng văn bản, hơn nữa khi mà số lượng văn bản đang gia tăng một cách nhanh chóng thì thao tác tìm lần lượt trong từng văn bản sẽ mất rất nhiều thời gian, công sức và là một công việc nhàm chán và không khả thi. Chính vì thế nhu cầu phân loại văn bản tự động là thực sự cần thiết.

Có rất nhiều công trình nghiên cứu về phân loại văn bản và đã có được những kết qủa đáng khích lệ, như là: Support Vector Machine, K–Nearest Neighbor, Linear Least Squares Fit, Neural Network, Naïve Bayes, Centroid Base… Điểm chung của các phương pháp này đều dựa vào xác suất thống kê hoặc dựa vào trọng số của các từ, cụm từ trong văn bản. Trong mỗi phương pháp đều có cách tính toán khác nhau, tuy nhiên các phương pháp này đều phải thực hiện một số bước chung như: đầu tiên mỗi phương pháp sẽ dựa vào thông tin về sự xuất hiện của các từ trong văn bản (tần số xuất hiện trong tập văn bản,…) để biểu diễn thành dạng vector, sau đó tùy từng bài toán cụ thể mà chúng ta sẽ quyết định chọn áp dụng phương pháp nào, công thức tính toán nào cho phù hợp để phân loại tập văn bản dựa trên tập các vector đã xây dựng được ở bước trên, nhằm mục đích đạt được kết qủa phân loại tốt nhất.

## 2.2. Phát biểu bài toán

Bài toán phân loại văn bản có thể được phát biểu như sau : Cho trước một tập văn bản D={d1,d2,…,dn} và tập chủ đề được định nghĩa C={c1,c2,…,cn}. Nhiệm vụ của bài toán là gán lớp di thuộc về cj đã được định nghĩa. Hay nói cách khác, mục tiêu của bài toán là đi tìm hàm f :



## 2.3. Mô hình tổng quát

Có rất nhiều hướng tiếp cận bài toán phân loại văn bản đã được nghiên cứu như: tiếp cận bài toán phân loại dựa trên lý thuyết đồ thị, cách tiếp cận sử dụng lý thuyết tập thô, cách tiếp cận thống kê… Tuy nhiên, tất cả các phương pháp trên đều dựa vào các phương pháp chung là máy học đó là : học có giám sát, học không giám sát và học tăng cường.

Vấn đề phân loại văn bản theo phương pháp thống kê dựa trên kiểu học có giám sát được đặc tả bao gồm 2 giai đoạn : giai đoạn huấn luyện và giai đoạn phân lớp.

* Giai đoạn huấn luyện

Chúng ta có một tập huấn luyện, mỗi phần tử trong tập huấn luyện được gán vào một hoặc nhiều lớp mà chúng ta sẽ thể hiện chúng bằng một mô hình mã hoá. Thông thường, mỗi phần tử trong tập huấn luyện được thể hiện theo dạng .Trong đó,  là vector biểu diễn cho văn bản trong tập huấn luyện.

Sau đó, chúng ta định nghĩa một lớp mô hình và một thủ tục huấn luyện. Lớp mô hình là họ các tham số của bộ phân loại, thủ tục huấn luyện là một giải thuật (hay thuật toán) để chọn ra một họ các tham số tối ưu cho bộ phân loại.

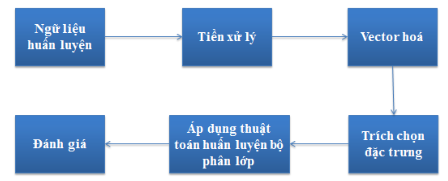


* 1. Mô hình giai đoạn huấn luyện

Đầu vào : ngữ liệu huấn luyện và thuật toán huấn luyện

Đầu ra : mô hình phân lớp (bộ phân lớp – classifier)

Các bước trong giai đoạn huấn luyện:



* 1. Các bước trong giai đoạn huấn luyện

Trong đó :

Ngữ liệu huấn luyện : kho ngữ liệu thu thập từ nhiều nguồn khác nhau.

Tiền xử lý : chuyển đổi tài liệu trong kho ngữ liệu thành một hình thức phù hợp để phân loại.

Vector hoá : mã hoá văn bản bởi một mô hình trọng số

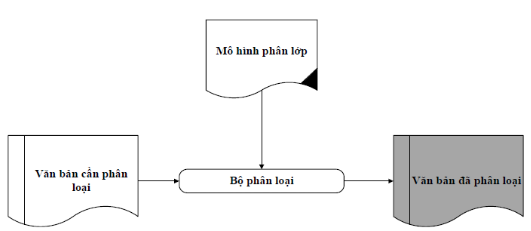
Trích chọn đặc trưng : loại bỏ những từ (đặc trưng) không mang thông tin khỏi tài liệu nhằm nâng cao hiệu suất phân loại và giảm độ phức tạp của thuật toán huấn luyện.

Thuật toán huấn luyện : Thủ tục huấn luyện bộ phân lớp để tìm ra họ các tham số tối ưu.

Đánh giá : bước đánh giá hiệu suất (chất lượng) của bộ phân lớp

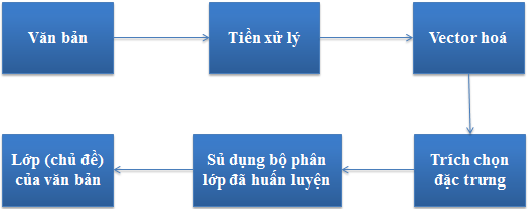
* Giai đoạn phân lớp

Sau khi đã hoàn thành giai đoạn huấn luyện, mô hình phân lớp sẽ được áp dụng cho các văn bản mới cần phân loại.



* 1. Mô hình giai đoạn phân lớp

Các bước trong giai đoạn phân lớp:



* 1. Các bước trong giai đoạn phân lớp

## 2.4. Môi trường phát triển

Hệ điều hành: Windows, Linux

Ngôn ngữ lập trình: Java

IDE: IntelIJ IDEA + Netbeans IDE

## 2.5. Thuật toán Naives Bayes

Cho X, C là các biến bất kỳ (rời rạc, liên tục, cấu trúc). Mục tiêu của ta là dự đoán C từ X. Từ mô hình Bayes ta có thể lượng giá các tham số của P(X|C), P(C) trực tiếp từ tập huấn luyện. Sau đó, ta sử dụng định lý Bayes để tính P(C|X=x).

**Độc lập điều kiện (conditional independence):** X độc lập điều kiện với Y khi cho Z nếu phân bố xác suất trên X độc lập với các giá trị của Y khi cho các giá trị của Z. Ta thường viết P(X|Y,Z) = P(X|Z). Ví dụ, P(Sấm sét|Mưa,Chớp) = P(Sấm sét|Chớp).

**Naive Bayes**

Giả sử D là tập huấn luyện gồm các mẫu biểu diễn dưới dạng X = . C_{i,D} là tập các mẫu của D thuộc lớp C_i ( i = \{1, ...,m\}). Các thuộc tính x_1, ...,x_n độc lập điều kiện đôi một với nhau khi cho lớp C.

**Thuật toán**

**Bước 1:** Huấn luyện Naive Bayes trên tập dữ liệu huấn luyện. Lượng giá P(C_i) và P(X_k|C_i).

**Bước 2:** X\_new được gán vào lớp cho giá trị công thức lớn nhất:

argmax_{C_k} P(C_i) \prod\limits_{k = 1}^n P(x_k | C_i)

**Trường hợp X là giá trị rời rạc**

Giả sử X = . Trong đó, x_i nhận các giá trị rời rạc. Khi đó, lượng giá P(C_i) và lượng giá P(X_k|C_i)theo công thức

P(C_i) \approx \frac{|C_{i, D}|}{D} \hskip 2em P(X_i|C_i) \approx \frac{\#C_{i, D}\{x_k\}}{|C_{i, D}|}

Để tránh trường hợp giá trị P(X\_k |C\_i ) = 0 do không có mẫu nào trong dữ liệu huấn kuyện thỏa mãn tử số, ta làm trơn bằng cách thêm một số mẫu ảo. Ta áp dụng phương pháp làm trơn Laplace.

với m là số lớp, r là số giá trị rời rạc của thuộc tính.

**Trường hợp X là giá trị liên tục**

Nếu thuộc tính nhận giá trị liên tục thì xác suất P(X\_k|C\_i) thường được tính dựa theo phân bố Gauss với giá trị trung bình mu và độ lệch sigma:

g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x - \mu)^2}{2\sigma^2}}

Và P(X_k|C_i) là:

P(X|C_i) = g(x_k, \mu_{c_i}, \sigma_{c_i})

## 2.6. Tf-idf

### Tf- term frequency

dùng để ước lượng tần xuất xuất hiện của từ trong văn bản. Tuy nhiên với mỗi văn bản thì có độ dài khác nhau, vì thế số lần xuất hiện của từ có thể nhiều hơn . Vì vậy số lần xuất hiện của từ sẽ được chia độ dài của văn bản (tổng số từ trong văn bản đó)

TF(t, d) = ( số lần từ t xuất hiện trong văn bản d) / (tổng số từ trong văn bản d)

### IDF- Inverse Document Frequency

dùng để ước lượng mức độ quan trọng của từ đó như thế nào . Khi tính tần số xuất hiện tf thì các từ đều được coi là quan trọng như nhau. Tuy nhiên có một số từ thường được được sử dụng nhiều nhưng không quan trọng để thể hiện ý nghĩa của đoạn văn , ví dụ :

Từ nối: và, nhưng, tuy nhiên, vì thế, vì vậy, …

Giới từ: ở, trong, trên, …

Từ chỉ định: ấy, đó, nhỉ, …

Vì vậy ta cần giảm đi mức độ quan trọng của những từ đó bằng cách sử dụng IDF :

IDF(t, D) = log\_e( Tổng số văn bản trong tập mẫu D/ Số văn bản có chứa từ t )

## 2.7. Cơ sở lý thuyết

### 2.7.1. Tiền xử lý văn bản

Văn bản trước khi được vector hoá, tức là trước khi sử dụng, cần phải được tiền xử lý. Quá trình tiền xử lý sẽ giúp nâng cao hiệu suất phân loại và giảm độ phức tạp của thuật toán huấn luyện.

Tuỳ vào mục đích bộ phân loại mà chúng ta sẽ có những phương pháp tiền xử lý văn bản khác nhau, như :

Chuyển vẳn bản về chữ thường.

Loại bỏ dấu câu (nếu không thực hiện tách câu).

Loại bỏ các kí tự đặc biệt biệt([ ],[.], [,], [:], [“], [”], [;], [/], [[]], [~], [`], [!], [@], [#], [$],[%],[^],[&],[\*],[(],[)]), các chữ số, phép tính toán số học.

Loại bỏ các stopword (những từ xuất hiện hầu hết trong các văn bản) không có ý nghĩa khi tham gia vào phân loại văn bản.

…

### 2.7.2. Phương pháp biểu diễn văn bản

Một trong những nhiệm vụ đầu tiền trong việc xử lý phân loại văn bản là chọn được một mô hình biểu diễn văn bản thích hợp. Một văn bản ở dạng thô (dạng chuỗi) cần được chuyển sang một mô hình khác để tạo thuận lợi cho việc biểu diễn và tính toán. Tuỳ thuộc vào từng thuật toán phân loại khác nhau mà chúng ta có mô hình biểu diễn riêng. Một trong những mô hình đơn giản và thường được sử dụng trong nhiệm vụ này là mô hình không gian vector. Một văn bản trong nhiệm vụ này được biểu diễn theo dạng , với  là một vector n chiều để đo lường giá trị của phần tử văn bản.

### 2.7.3. Mô hình không gian vector

Mô hình không gian vector là một trong những mô hình được sử dụng rộng rãi nhất cho việc tìm kiếm (truy hồi) thông tin. Nguyên nhân chính là bởi vì sự đơn giản của nó.

Trong mô hình này, các văn bản được thể hiện trong một không gian có số chiều lớn, trong đó mỗi chiều của không gian tương ứng với một từ trong văn bản. Phương pháp này có thể biểu diễn một cách hình tượng như sau : mỗi văn bản D được biểu diễn dưới dạng  (vector đặc trưng cho văn bản D). Trong đó,  =( x1, x2, … , xn ), và n là số lượng đặc trưng hay số chiều của vector văn bản, xi là trọng số của đặc trưng thứ i (với 1≤ i ≤n).

Như vậy, nếu trong kho ngữ liệu của quá trình huấn luyện nhiều văn bản, ta kí hiệu Dj, là văn bản thứ j trong tập ngữ liệu, và vector j=( x1j , x2j , … , xnj ) là vector đặc trưng cho văn bản Dj, và xij là trọng số thứ i của vector văn bản j.

### 2.7.4. Khái niệm trọng số

Một vấn đề quan trọng nữa trong việc biểu diễn một văn bản đó là tính trọng số cho vector đặc trưng của văn bản. Có nhiều cách khác nhau để tính trọng số này như :

Word frequency weighting

Boolean weighting

tf\*idf weighting

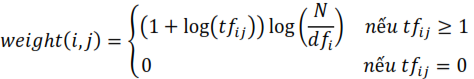
Entropy weighting

Tuy nhiên, để đơn giản cho vấn đề này, chúng ta sẽ chỉ xem xét cách tính Word frequency weighting (trọng số tần suất từ) và tf\*idf, một cách đơn giản đó là đếm số từ đó trong văn bản. Tuy nhiên vẫn có nhiều cách khác nhau để tính trọng số dạng này.

Có ba thông tin được sử dụng trong cách tính trọng số bằng tần suất từ là : term frequency ( tfij số lần suất hiện của từ wi trong văn bản dj ), document frequency (dfi số văn bản có chứa từ wi), collection frequency ( cfi số lần suất hiện của từ wi trong cả tập ngữ liệu). Trong đó, dfi ≤ cfi và ∑j tfij =cfi.

Thông tin được nắm bắt bởi term frequency là sự nổi bật của thông tin (hay từ) trong một văn bản. Term frequency càng cao (số lần xuất hiện càng nhiều trong văn bản) thì đó là từ miêu tả tốt cho nội dung văn bản. Giá trị thứ hai, document frequency, có thể giải thích như là một bộ chỉ định nội dung thông tin. Một từ được tập trung ngữ nghĩa thường xảy ra nhiều lần trong một văn bản nếu nó cũng xuất hiện trong tất cả các văn bản khác. Nhưng từ không được tập trung ngữ nghĩa trải đều đồng nhất trong tất cả các văn bản.

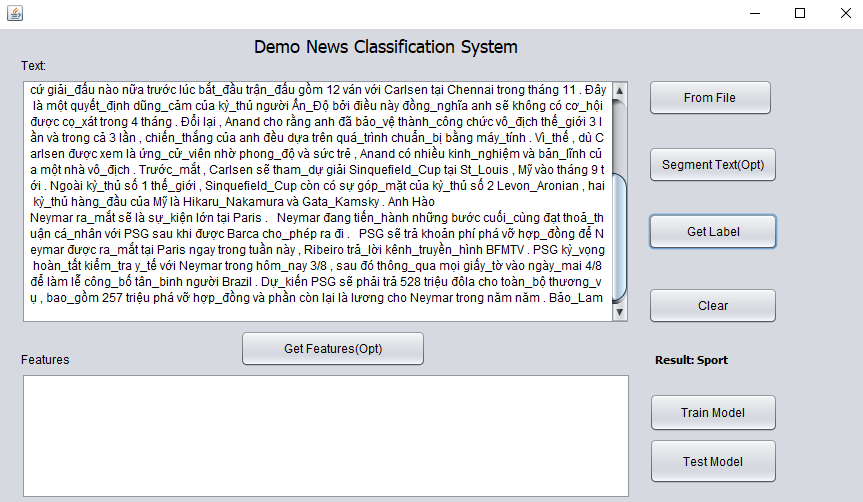
Để thể hiện trọng số phản ánh hết thông tin của từ, thường ta sẽ kết hợp cả hai loại trọng số là tf và df trong một đơn vị chung. Dạng biểu diễn trọng số này được gọi là tf \* idf. Công thức kết hợp hai giá trị trọng số :



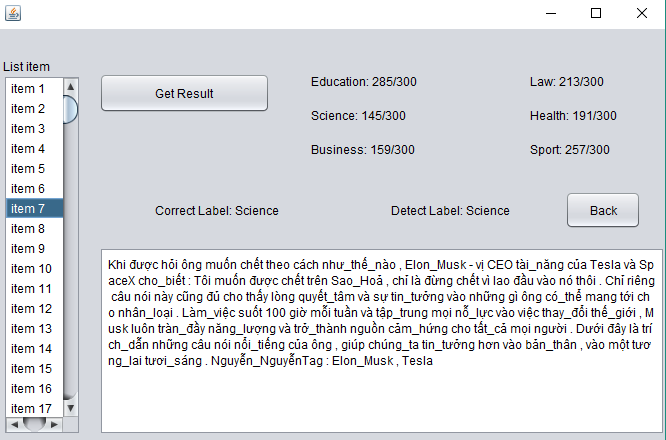
Trong đó, N là tổng số văn bản. Biểu thức thứ nhất áp dụng cho các từ có xuất hiện trong văn bản, còn biểu thức thứ hai cho các từ không xuất hiện trong văn bản.

# CHƯƠNG 3. GIAO DIỆN ỨNG DỤNG DEMO

## 3.1. Giao diện form chính



## 3.2. Giao diện form kiểm tra



## 3.3. Giao diện form huấn luyện

