Mục Lục

[I. Phân loại văn bản. 2](#_Toc278460910)

[1. Khái niệm về phân lớp – phân loại văn bản/ tài liệu (Document /categorization classification) 2](#_Toc278460911)

[2. Xây dựng hệ thống phân loại văn bản. 3](#_Toc278460916)

[2.1 Ba giai đoạn xây dựng hệ thống phân loại 3](#_Toc278460917)

[2.2 Các giai đoạn xử lý dữ liệu trong phân lớp văn bản 4](#_Toc278460918)

[Bước 1: Tiền xử lý số liệu 5](#_Toc278460919)

[Bước 2: Tách từ 6](#_Toc278460920)

[Bước 3: Xác đinh trọng số cho từ. 6](#_Toc278460921)

[Bước 4: Sử dụng thuật toán để phân loại văn bản. 7](#_Toc278460922)

[II. Các thuật toán trong phân lớp văn bản. 8](#_Toc278460923)

[1. Thuật toán cây quyết định (Decision tree) 8](#_Toc278460924)

[2. Thuật toán (K-nearest neighbor -KNN) 11](#_Toc278460925)

[3. Naïve Bayes (NB) 12](#_Toc278460926)

[4. Support Vector Machine (SVM ). 13](#_Toc278460927)

[5. Support Vector Machines Nearest Neighbor (SVM-NN) 14](#_Toc278460928)

# Phân loại văn bản.

## Khái niệm về phân lớp – phân loại văn bản/ tài liệu (Document /categorization classification)

Phân lớp – phân loại văn bản/ tài liệu (document /categorization classification) [[1]](#footnote-2) là việc xác định chuyên mục (categories) cho một văn bản điện tử, dựa trên việc phân tích nội dung của văn bản đó. Một văn bản có thể thuộc một hoặc nhiều chuyên mục. Có hai hướng tiếp cận giải quyết bài toán phân lớp văn bản đó là: phân lớp dựa trên hệ chuyên gia và tiếp cận dựa trên máy học.

### Phân lớp văn bản dựa trên cách tiếp cận hệ chuyên gia:

Theo cách tiếp cận này, việc phân lớp văn bản tự động được điều khiển bằng tay bởi các chuyên gia tri thức và hệ chuyên gia có khả năng đưa ra quyết định phân lớp. Hệ chuyên gia bao gồm một tập các luật logic định nghĩa bằng tay, cho mỗi loại có dạng:

If (DNF formula) then (category).

Công thức DNF (“Disjunctive Normal Form”) là hợp của các mệnh đề liên kết, tài liệu được phân lớp vào chuyên mục (category) nếu nó thỏa mãn công thức, nghĩa là nếu nó thỏa mãn ít nhất một mệnh đề trong công thức.

Điều trở ngại của cách tiếp cận này là hạn chế trong quá trình thu nhận tri thức từ tài liệu của các hệ thống chuyên gia. Nghĩa là, các luật phải được định nghĩa bằng tay bởi tri thức của chuyên gia về lĩnh vực được nêu trong tài liệu. Nếu tập hợp của các loại được cập nhật, thì hai nhà chuyên gia phải can thiệp lại, và nếu phân lớp được chuyển hoàn toàn sang một phạm vi khác, một chuyên gia về lĩnh vực này cần thiết phải can thiệp vào và công việc phải được bắt đầu lại từ tập tài liệu ban đầu.

### Phân lớp văn bản dựa trên cách tiếp cận máy học [1]

Máy học (Machine learning [[2]](#footnote-3)) là một môn khoa học liên quan đến việc thiết kế và phát triển của các thuật toán cho phép các máy tính phát triển các hành vi dựa trên dữ liệu thực nghiệm. Quá trình học của máy sử dụng các dữ liệu được gọi là dữ liệu học để xác định những đặc trưng mà máy cần phải học, từ đó xây dựng bộ quyết định cho dữ liệu mới.

### Theo cách tiếp cận này, quá trình học để xây dựng tự động một bộ phân lớp được thực hiện bằng cách, cho máy học các đặc trưng của tập hợp các tài liệu đã được phân loại bằng tay bởi chuyên gia về lĩnh vực cần phân loại. Từ đó, khi cần phân loại một tài liệu, hệ thống sẽ dựa vào bộ phân lớp được học trước đó và các đặc trưng của tài liệu mới để phân loại tài liệu theo đúng nhóm. Trong kỹ thuật máy học, bài toán phân lớp là hoạt động học có giám sát, quá trình học được “giám sát” bởi tri thức của các phân lớp và của các mẫu huấn luyện thuộc chúng.

### Học có giám sát [[3]](#footnote-4) là một kĩ thuật của ngành [học máy](http://vi.wikipedia.org/wiki/H%E1%BB%8Dc_m%C3%A1y) để xây dựng một hàm (function) từ dữ liệu huấn luyện. [Dữ liệu huấn luyện](http://vi.wikipedia.org/wiki/T%E1%BA%ADp_hu%E1%BA%A5n_luy%E1%BB%87n) bao gồm các cặp gồm đối tượng đầu vào (thường dạng vector), và đầu ra mong muốn. Đầu ra của một hàm có thể là một giá trị liên tục (gọi là [hồi qui](http://vi.wikipedia.org/wiki/Ph%C3%A2n_t%C3%ADch_h%E1%BB%93i_qui)), hay có thể là dự đoán một nhãn phân loại cho một đối tượng đầu vào (gọi là [phân loại](http://vi.wikipedia.org/wiki/Ph%C3%A2n_lo%E1%BA%A1i_b%E1%BA%B1ng_th%E1%BB%91ng_k%C3%AA)). Nhiệm vụ của chương trình học có giám sát là dự đoán giá trị của hàm cho một đối tượng bất kì là đầu vào hợp lệ, sau khi đã xem xét một số ví dụ huấn luyện (nghĩa là, các cặp đầu vào và đầu ra tương ứng). Để đạt được điều này, chương trình học phải tổng quát hóa từ các dữ liệu sẵn có để dự đoán được những tình huống chưa gặp phải theo một cách "hợp lí").

Một số thuật toán dựa trên tiếp cận máy học được sử dụng phổ biến hiện nay bao gồm: Cây quyết định, K-Nearest Neighbor (KNN), Support Vector Machine (SVM), thuật toán Naïve Bayes (NB), Neural Network (NNet).

Ngoài ra, còn một số thuật toán được phát triển dựa trên các thuật toán đã nói ở trên ví dụ: với thuật toán SVM có các thuật toán cải tiến như Fuzzy Support Vector Machines.Ngoài ra, cũng có một số phương pháp người ta kết hợp các thuật toán lại với nhau như Support Vector Machines Nearest Neighbor (SVM-NN) việc kết hợp này sẽ tận dụng những ưu điểm và bổ xung những mặt yếu của các thuật toán mà ta sẽ xem sét kỹ hơn ở phần trình bày về các thuật toán.

## Xây dựng hệ thống phân loại văn bản.

Trong phần này, bài báo cáo sẽ trình bày về cách thức chung để xây dựng hệ thống phân loại văn bản và các bước xử lý dữ liệu (nội dung văn bản) trong quá trình phân loại.

### 2.1 Ba giai đoạn xây dựng hệ thống phân loại

Ta có thể nhận ra ba giai đoạn khác nhau trong việc thiết kế hệ thống phân loại văn bản: biểu diễn tài liệu, xây dựng bộ phân loại, đánh giá bộ phân loại.

+ Giai đoạn biểu diễn tài liệu: Tức là chuyển tài liệu từ dạng thô (.doc,.HTML,.PDF...) sang dạng thích hợp với thuật toán rút trích của giai đoạn sau.

+ Giai đoạn xây dựng bộ phân loại: Từ các thuật toán mà ta chọn trong hệ thống phân loại văn bản ta sẽ xây dựng bộ phân loại cho hệ thống. Ở đây ta có thể hiểu là xây dựng mẫu để ta làm mốc đánh giá cho quá trình phân loại.

+ Giai đoạn đánh giá bộ phân loại: Từ kết quả của bộ phân loại và kiểm thử ta đánh giá bộ phân loại đó dựa vào tỉ lệ đúng và chưa đúng của việc phân loại tài liệu.



*Hình 1: Hệ Thống Phân Loại Văn bản*

***Các vấn đề mà hệ thống phân loại cần phải quan tâm và giải quyết***

***Độ chính xác***: Dựa vào phần trăm của văn bản đúng so với văn bản đưa vào phân loại. Tỉ lệ càng cao thì hệ thống sẽ được đánh giá là tốt.

***Tốc độ***: Một hệ thống có tốc độ phân loại nhanh nhưng độ chính xác thấp hoặc hệ thống có tốc độ thấp nhưng độ chính xác cao thì không được cho là hệ thống tốt vì vậy phải đảm bảo tính tốc độ và chính xác cho hệ thống.

***Dễ hiểu***: Một hệ thống phân loại dễ hiểu sẽ tạo cho người sử dụng tin tưởng hơn vào hệ thống, đồng thời cũng giúp cho người sử dụng tránh được việc hiểu lầm kết quả của một luật được đưa ra bởi hệ thống.

***Thời gian để học***: Yêu cầu hệ thống phải học rất nhanh một luật phân lớp hoặc nhanh chóng điều chỉnh một luật đã được học cho phù hợp với thực tế.

### 2.2 Các giai đoạn xử lý dữ liệu trong phân lớp văn bản

Trong quá trình phân lớp tài liệu văn bản, thì dữ liệu (nội dung văn bản) được xử lý qua bốn bước bao gồm:

**Bước 1**: Tiền xử lý số liệu.

**Bước 2**: Tách từ.

**Bước 3**: Xác đinh trọng số cho từ.

**Bước 4**: Sử dụng thuật toán để phân loại văn bản.



*Các bước trong quá trình phân loại văn bản.*

### Bước 1: Tiền xử lý số liệu

Mục đích của bước này là xử lý tương đối sạch dữ liệu đọc vào để các bước sau sẽ xử lý tốt hơn, do đó công việc của bước này sẽ chỉ là chuyển nội dung văn bản có thành chuỗi ký tự thuần túy (text), do đó nó sẽ có yêu cầu như sau:

* Dữ liệu đầu vào quá trình xử lý: Tệp văn bản cần phải phân tích (File PDF, TXT, DOC, HTML, HTM)
* Dữ liệu ra sau khi xử lý: chuỗi ký tự thuần túy (text only) với font chữ định dạng đã định sẵn.

Thực hiện:

* Nếu dữ liệu đầu vào là tệp văn bản dạng text (txt) thì lấy tất cả số liệu
* Nếu dữ liệu đầu vào có các định dạng khác như HTML, MS Word, PDF… thì sẽ sử dụng các công cụ hỗ trợ chuyển đổ dữ liệu đầu vào thành text.

Làm sạch số liệu tiếp theo bao gồm:

* Loại bỏ các khoảng trắng nhiều hơn 1 khoảng trắng
* Các dấu xuống dòng
* Cách dòng trống
* Các ký tự lạ

### Bước 2: Tách từ

Mục đích của bước này là tách một văn bản text thuần túy thành những từ, đảm bảo những từ đó có ý nghĩa trong văn bản chứa nó. Như vậy sau khi tách từ - chặt câu, ta sẽ xét các nhiệm vụ sau đây:

+Tách lọc (Filtration)

Tách lọc được biết đến như một quá trình của sự quyết định những từ nào nên được sử dụng để biểu diễn cho các tài liệu vì thế nó có thể được sử dụng cho:

* Mô tả nội dung của văn bản.
* Có sự phân biệt tài liệu từ những tài liệu khác trong bộ sưu tập.

Ở giai đoạn này ta loại bỏ các từ stopword [[4]](#footnote-5)(danh mục các từ không ảnh hưởng đến nội dung văn bản ).

+ Stemming [[5]](#footnote-6)(gốc từ)

Stemming là quá trình liên quan đến việc xử lý giảm đi số từ đối với gốc từ hay cội nguồn khác nhau của chúng. Do vậy, những từ "computer", "computing", "compute" được giảm lại thành từ "compute" và "walks", "walking" và "walker" được giảm lại thành "walk" . Đối với tiếng Anh, bộ xác định gốc từ phổ biến là thuật toán xác định gốc từ của Martin Porter (Martin Porter's Stemming Algorithm).

Để đảm bảo độ chính xác của từ được tách (từ ghép) trong quá trình này còn sử dụng một số thuật toán tách từ như N- gram [[6]](#footnote-7).

Như vậy trong bước này:

* Dữ liệu đầu vào quá trình xử lý: Chuỗi ký tự văn bản thuần túy.
* Dữ liệu ra sau khi xử lý: Vector chứa các từ được tách trong văn bản.

### Bước 3: Xác đinh trọng số cho từ.

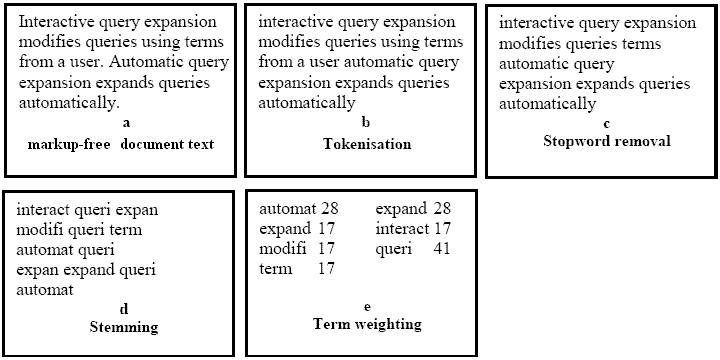
Ứng với mỗi thuật toán áp dụng trong hệ thống phân lớp mà ta đi xác định những trọng số cho từ trong văn bản.

Trọng số ở đây thường dựa trên các đại lượng sau:

* Tần suất từ Term Frequency [[7]](#footnote-8) (tf) - tần số (hay số lần) xuất hiện của từ- thuật ngữ trong văn bản.
* Tần số văn bản Document Frequency (df) – tần số (hay số lần) xuất hiện của từ - thuật ngữ trong khối tài liệu đã có.

Tùy vào thuật toán mà hệ thống sẽ chọn đại lượng đặc trưng – trọng số cho các vector trong văn bản. Như vậy trong bước này:

* Dữ liệu đầu vào quá trình xử lý: Vector các từ
* Dữ liệu ra sau khi xử lý: Vector chứa các từ đã được gán nhãn – đánh trọng số.



Các bước trong quá trình đánh trọng số cho văn bản.

### Bước 4: Sử dụng thuật toán để phân loại văn bản.

Đây là bước chính của hệ thống. Hệ thống sẽ xử dụng thuật toán để xử lý trên nội dung văn bản đã được biểu diễn để phân loại văn bản đó vào một nhóm – chủ đề phù hợp.

Đối với các thuật toán cần phải học thì dữ liệu học cũng được biểu diễn và gán trọng số lần lượt qua các bước 1, 2, 3 cùng với đó chủ để- nhóm của dữ liệu học đã được chuyên gia xác định sẵn. Qua giai đoạn học này bộ phân lớp được xây dựng từ dữ liệu học sẽ làm căn cứ để

* Dữ liệu đầu vào quá trình xử lý: Vector các từ, dữ liệu chuẩn của các nhóm văn bản, bộ phân lớp.
* Dữ liệu ra sau khi xử lý: Xác định nhóm của văn bản.

# Các thuật toán trong phân lớp văn bản.

### Thuật toán cây quyết định (Decision tree) [[8]](#footnote-9)

Đây là phương pháp học xấp xỉ các hàm mục tiêu có giá trị rời rạc. Mặt khác cây quyết định còn có thể chuyển sang dạng biểu diễn tương đương dưới dạng cơ sở tri thức là các luật *Nếu – Thì.*

**\*Ý tưởng thuật toán**

Bộ phân lớp cây quyết định là một dạng cây, mà mỗi nút được gán nhãn là một đặc trưng, mỗi nhánh là giá trị trọng số xuất hiện của đặc trưng trong văn bản cần phân lớp, và mỗi lá là nhãn của phân lớp tài liệu. Việc phân lớp của một tài liệu dj sẽ được duyệt đệ quy theo trọng số của những đặc trưng có xuất hiện trong văn bản dj. Thuật toán lặp đệ quy đến khi đạt đến nút lá và nhãn của dj chính là nhãn của nút lá tìm được. Thông thường việc phân lớp văn bản nhị phân sẽ tương thích với việc dùng cây nhị phân.

Cây quyết định này được tổ chức như sau: Các nút trong được gán nhãn bởi các thuật ngữ, nhãn của các cung tương ứng với trọng số của thuật ngữ trong tài liệu mẫu, nhãn của các lá tương ứng với nhãn của các lớp. Cho một tài liệu dj, ta sẽ thực hiện so sánh các nhãn của cung xuất phát từ một nút trong (tương ứng với một thuật ngữ nào đó) với trọng số của thuật ngữ này trong dj, để quyết định nút trong nào sẽ được duyệt tiếp. Quá trình này được lặp từ nút gốc của cây, cho tới khi nút được duyệt là một lá của cây. Kết thúc quá trình này, nhãn của nút lá sẽ là nhãn của lớp được gán cho văn bản

**Ví dụ**

Ta có bảng dữ liệu gồm 10 tài liệu được mô tả bằng vector nhị phân thông qua 7 thuật ngữ “*thời tiết*”, “*độ ẩm*”, “*lượng mưa*”, “*gió*”, “*khí hậu*”, “*thuyền*”, “*nhiệt độ*”. Trong đó cột cuối cùng trong bảng là nhãn được gán cho từng tài liệu với chủ đề ***thời tiết***, giá trị của tài liệu di trong cột này bằng 1 tương ứng di thuộc chủ đề thời tiết, nếu giá trị này bằng 0 thì di không thuộc chủ đề thời tiết.

Bảng : Biểu diễn văn bản bằng vector nhị phân

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tài liệu | *thời tiết* | *độ ẩm* | *lượng mưa* | *gió* | *khí hậu* | *thuyền* | *nhiệt độ* | ***thời tiết*** |
| d1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| d2 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| d3 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 |
| d4 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| d5 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| d6 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| d7 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| d8 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| d9 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| d10 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

Cây quyết định được xây dựng tương ứng với bảng trên là:



Hình Xây dựng cây quyết định cho tập mẫu dùng để huấn luyện

Từ cây quyết định trên ta xây dựng được cơ sở tri thức dưới dạng luật *Nếu -Thì*như sau:

***Nếu*** (*thời tiết=*1) và (*lượng mưa=*1) và (*độ ẩm*=1) ***Thì*** class *thời tiết*=1

***Nếu*** (*thời tiết=*1) và (*lượng mưa=*0) và (*độ ẩm*=1) ***Thì*** class *thời tiết*=0

***Nếu*** (*thời tiết=*1) và (*gió=*0) và (*độ ẩm*=0) ***Thì*** class *thời tiết*=0

***Nếu*** (*thời tiết=*1) và (*gió=*1) và (*độ ẩm*=0) ***Thì*** class *thời tiết*=1

***Nếu*** (*thời tiết=*0) và (*khí hậu=*0) ***Thì*** class *thời tiết*=0

***Nếu*** (*thời tiết=*0) và (*khí hậu=*1) và (*nhiệt độ*=0) ***Thì*** class *thời tiết*=0

***Nếu*** (*thời tiết=*0) và (*khí hậu=*1) và (*nhiệt độ*=1) ***Thì*** class *thời tiết*=1

Xét tài liệu d, được biểu diễn bởi vector nhị phân như sau:

d = (*thời tiết, lượng mưa, độ ẩm, gió, khí hậu, thuyền, nhiệt độ*) = (1, 1, 1, 0, 0, 1, 0)

Quá trình tìm kiếm lời giải trên cây quyết định sẽ như sau:



Hình Quá trình tìm kiếm lời giải trên cây quyết định

Class *thời tiết*=1, hay nói cách khác văn bản d thuộc lớp văn bản nói về chủ đề thời tiết (lớp thời tiết).

### Thuật toán (K-nearest neighbor -KNN) [[9]](#footnote-10)

Đây là phương pháp truyền thống về hướng tiếp cận dựa trên thống kê.

**\* Ý tưởng của thuật toán.**

Ý tưởng chính của thuật toán K-láng giềng gần nhất (K-NN) là *so sánh độ phù hợp của văn bản d với từng nhóm chủ đề, dựa trên k văn bản mẫu trong tập huấn luyện mà có độ tương tự với văn bản d là lớn nhất.*

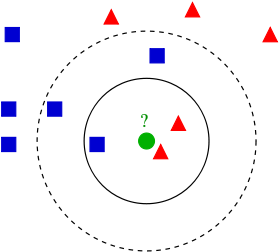
Khi cần phân loại một văn bản mới, thuật toán sẽ tính khoảng cách (khoảng cách Euclide, Cosine ...) của tất cả các văn bản trong tập huấn luyện đến văn bản này để tìm ra k văn bản “gần nhất” (gọi là k “láng giềng”), sau đó dùng các khoảng cách này đánh trọng số cho tất cả chủ đề. Trọng số của một chủ đề chính là tổng tất cả khoảng cách ở trên của các văn bản trong k láng giềng có cùng chủ đề, chủ đề nào không xuất hiện trong k láng giềng sẽ có trọng số bằng 0. Sau đó các chủ đề sẽ được sắp xếp theo mức độ trọng số giảm dần và các chủ đề có trọng số cao sẽ được chọn là chủ đề của văn bản cần phân loại.

Có 2 vấn đề cần quan tâm khi phân lớp văn bản bằng thuật toán K- láng giềng gần nhất là xác định khái niệm gần, công thức để tính mức độ gần; và làm thế nào để tìm được nhóm văn bản phù hợp nhất với văn bản đó (nói cách khác là tìm được chủ đề thích hợp để gán cho văn bản).

Khái niệm gần ở đây được hiểu là độ tương tự giữa các văn bản. Có nhiều cách để xác định độ tương tự giữa hai văn bản, trong đó công thức Cosine trọng số được coi là hiệu quả để đánh giá độ tương tự giữa hai văn bản. Cho *T={t1, t2, …, tn*} là tập hợp các thuật ngữ; *W={wt1, wt2, …, wtn*} là vector trọng số, *wti* là trọng số của thuật ngữ *ti*. Xét hai văn bản X={x1, x2, …, xn} và Y={y1, y2, …, yn}, xi, yi lần lượt là tần số xuất hiện của thuật ngữ ti trong văn bản X, Y. Khi đó độ tương tự giữa hai văn bản X và Y được tính theo công thức sau:



Trong vector X, Y các thành phần xi, yi được chuẩn hoá theo tần số xuất hiện của thuật ngữ ti trong các văn bản X và Y. Vector W được xác định bằng tay hoặc tính vector W theo nghịch đảo tần suất văn bản IDF khi đó văn bản được biểu diễn dưới dạng vector tần xuất TFxIDF



Mô hình thuật toán KNN

### Naïve Bayes (NB) [[10]](#footnote-11)

NB là phương pháp phân loại dựa vào xác suất được sử dụng rộng rãi trong lĩnh vực máy học.

**\* Ý tưởng của thuật toán**

Ý tưởng cơ bản của cách tiếp cận Naïve 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. Điểm quan trọng của phương pháp này chính là ở chỗ giả định rằng sự xuất hiện của tất cả các từ trong văn bản đều độc lập với nhau. Với giả định này NB không sử dụng sự phụ thuộc của nhiều từ vào một chủ đề, không sử dụng việc kết hợp các từ để đưa ra phán đoán chủ đề và do đó việc tính toán NB chạy nhanh hơn các phương pháp khác.

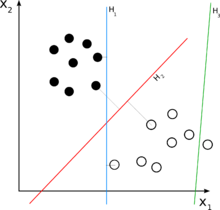
### Support Vector Machine (SVM [[11]](#footnote-12)).

SVM là một phương pháp phân lớp xuất phát từ lý thuyết học thống kê.

**\* Ý tưởng của thuật toán**

Ý tưởng của nó là ánh xạ (tuyến tính hoặc phi tuyến) dữ liệu vào không gian các vector đặc trưng (space of feature vectors) mà ở đó một siêu phẳng tối ưu được tìm ra để tách dữ liệu thuộc hai lớp khác nhau.

Cho trước một tập huấn luyện được biểu diễn trong không gian vector trong đó mỗi tài liệu là một điểm, phương pháp này tìm ra một siêu mặt phẳng h quyết định tốt nhất có thể chia các điểm trên không gian này thành hai lớp riêng biệt tương ứng lớp + và lớp –. Chất lượng của siêu mặt phẳng này được quyết định bởi khoảng cách (gọi là biên) của điểm dữ liệu gần nhất của mỗi lớp đến mặt phẳng này. Khoảng cách biên càng lớn thì mặt phẳng quyết định càng tốt đồng thời việc phân loại càng chính xác. Mục đích thuật toán SVM tìm được khoảng cách biên lớn nhất.



*Hình mô hình support vector machine\*

## Support Vector Machines Nearest Neighbor (SVM-NN)

Support Vector Machines Nearest Neighbor (SVM-NN) (Blanzieri & Melgani 2006) là một thuật toán phân lớp cải tiến gần đây nhất của phương pháp phân lớp SVM. SVM-NN là một kỹ thuật phân loại văn bản máy học sử dụng kết hợp cách tiếp cận K-láng giềng gần nhất (K-NN) với những luật ra quyết định dựa trên SVM (SVM-based decision rule).

### *Ý tưởng của thuật toán SVM-NN*

Thuật toán phân lớp SVM-NN kết hợp các ý tưởng của thuật toán phân lớp SVM và thuật toán phân lớp K-NN.

Nó hoạt động theo cách sau:

- Cho một mẫu để phân loại, thuật toán xác định k mẫu gần nhất trong các mẫu dữ liệu của tập dữ liệu huấn luyện.

- Một phân loại SVM được huấn luyện trên những mẫu này.

- Sau đó, các bộ phân loại SVM được huấn luyện sẽ được sử dụng để phân loại các mẫu chưa biết.

[1] Sebastiani, F.: Machine learning in automated text categorization. In: ACM Computing Sur-veys, 34(1), Kluwer Academic Publishers (2002) 1–47

1. <http://en.wikipedia.org/wiki/Document_classification> [↑](#footnote-ref-2)
2. <http://en.wikipedia.org/wiki/Machine_learning> [↑](#footnote-ref-3)
3. <http://vi.wikipedia.org/wiki/H%E1%BB%8Dc_c%C3%B3_gi%C3%A1m_s%C3%A1t> [↑](#footnote-ref-4)
4. <http://en.wikipedia.org/wiki/Stop_words> [↑](#footnote-ref-5)
5. <http://en.wikipedia.org/wiki/Stemming> [↑](#footnote-ref-6)
6. <http://en.wikipedia.org/wiki/N-gram> [↑](#footnote-ref-7)
7. <http://en.wikipedia.org/wiki/Term_frequency> [↑](#footnote-ref-8)
8. <http://en.wikipedia.org/wiki/Decision_tree> [↑](#footnote-ref-9)
9. <http://en.wikipedia.org/wiki/K-nearest_neighbor> [↑](#footnote-ref-10)
10. <http://en.wikipedia.org/wiki/Na%C3%AFve_Bayes> [↑](#footnote-ref-11)
11. <http://en.wikipedia.org/wiki/Support_Vector_Machine> [↑](#footnote-ref-12)