ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Hoàng Thị Ngọc Trang

PHƯƠNG PHÁP ĐỒNG HUẤN LUYỆN VÀ ỨNG DỤNG

LUẬN VĂN THẠC SĨ

Ngành**: Công nghệ Thông tin**
Chuyên ngành**: Khoa học Máy tính**
Mã số: 60 48 01

NGƯỜI HƯỚNG DẪN KHOA HỌC

***PGS.TS*. Hoàng Xuân Huấn**

HÀ NỘI – 2009

MỤC LỤC

Trang

BẢNG KÝ HIỆU VÀ CHỮ VIẾT TẮT
MỤC LỤC
MỞ ĐẦU

MỞ ĐẦU

Sự phát trển mạnh mẽ của công nghệ cao nói chung và khoa học máy tính nói riêng ngày càng thu hút nhiều nhà khoa học và công nghệ quan tâm nghiên cứu bài toán nhận dạng mẫu. Thoạt tiên, bài toán nhận dạng mẫu xuất phát từ nhu cầu tạo nên các thành phần máy có khả năng quan sát môi trường. Cùng với sự phát triển của các ứng dụng công nghệ thông tin, đặc biệt trong lĩnh vực học máy, người ta phải đi sâu phát triển các hệ nhận dạng mẫu có khả năng tìm các mẫu mới trong các cơ sở dữ liệu lớn hay còn gọi là khám phá tri thức từ dữ liệu.

Phân lớp mẫu là bài toán thường gặp nhất trong nhận dạng mẫu và phân thành hai loại có giám sát và không có giám sát. Trong bài toán phân lớp có giám sát, dựa trên một tập dữ liệu đã được gán nhãn, người ta xây dựng một bộ phân lớp để gán nhãn cho các dữ liệu chưa biết. Còn trong bài toán không giám sát, người ta phân một tập dữ liệu chưa được gán nhãn thành các các tập con sao cho các đối tượng dữ liệu trong mỗi tập con thì có đặc tính giống nhau hơn so với đối tượng ở các tập con khác.

Trong các bài toán nhận dạng mẫu, bài toán phân lớp có giám sát là bài toán được ứng dụng rộng rãi nhất. Việc xây dựng bộ phân lớp trong bài toán này được thực hiện bởi các thuật toán học máy (học có giám sát). Với học có giám sát truyền thống con người thường phải bỏ ra rất nhiều công sức để gán nhãn cho tập dữ liệu đào tạo nếu muốn có một bộ học tốt. Nhưng trong thực tế lại luôn tồn tại sẵn một nguồn "tài nguyên" phong phú đó là nguồn dữ liệu chưa gán nhãn. Một phương pháp học mới đã ra đời nhằm đạt được mục tiêu "khai thác" được nguồn tài nguyên phong phú này, nó giúp giảm nhiều chi phí và công sức trong việc gán nhãn cho con người. Phương pháp này đã thu hút được rất nhiều sự quan tâm của các nhà khoa học và được đề gọi chung với tên phương pháp học bán giám sát (Semi-supervised learning: SSL). *Phương pháp này đầu tiên được giới thiệu bởi* A. Blum, T. Mitchel *vào năm 1998 [11] và*

Xiaojin Zhu (2006) đã đưa ra một cái nhìn tương đối đầy đủ và tổng quát (chi tiết xem [47]).

Mục đích của học bán giám sát là khai thác sự liên kết giữa dữ liệu đã gán nhãn và dữ liệu chưa gán nhãn để hiểu và thiết kế được thuật toán sao cho có thể tận dụng tốt thông tin từ nguồn dữ liệu chưa gán nhãn. Học bán giám sát được quan tâm nhiều trong khai phá dữ liệu bởi những nguồn dữ liệu chưa gán nhãn thực sự phong phú và sẵn có. Ngoài ra học bán giám sát còn đưa ra một công cụ định lượng để hiểu được cách học phân loại của con người, khi phần lớn dữ liệu học là dữ liệu chưa được gán nhãn [48].

Ban đầu học bán giám sát được áp dụng theo mô hình tự huấn luyện (self-training), trong đó bộ phân lớp được xây dựng dựa trên một tập dữ liệu đào tạo nhỏ đã được gán nhãn sau đó mở rộng dần tập dữ liệu này để đào tạo tăng cường bằng cách bổ sung thêm các dữ liệu được bộ học đoán nhận với độ tin cậy cao. Sau đó ý tưởng này được áp dụng để các phương pháp học có tính tương thích với tên gọi chung là các thuật toán đồng huấn luyện (co-training) hay học đa khung nhìn (multiview learning). Bên cạnh các thuật toán đồng huấn luyện đang ứng dụng rộng rãi cho các bài toán phân lớp, Zhi hua Zhou và Ming Li (2007) cũng đề xuất một thuật toán đồng huấn luyện cho bài toán hồi quy [49].

Luận văn này trình bày khảo cứu của tác giả về các thuật toán đồng huấn luyện (co-training) trong các lược đồ thông dụng nhất và thử nghiệm ứng dụng phương pháp hồi quy đồng huấn luyện để nâng cao chất lượng của mạng nơron RBF trong trường hợp thiếu dữ liệu đào tạo.

Ngoài phần kết luận, bố cục của luận văn được trình bày như sau.

Chương 1 *Giới thiệu chung về nhận dạng mẫu bao gồm các bài toán cơ bản, các hệ nhận dạng mẫu, chu trình thiết kế hệ nhận dạng mẫu.*

Chương 2 *Giới thiệu về học bán giám sát và hai thuật toán học bán giám sát chính là phương pháp tự huấn luyện (self-training) và đồng huấn luyện (co-training).*

Chương *3* *Trình bày các lược đồ ứng dụng chính của giải thuật đồng huấn luyện.*

Chương 4 *Ứng dụng co-training nâng cao chất lượng mạng nội suy RBF (Radial Basis Function).*

CHƯƠNG 1. GIỚI THIỆU VỀ NHẬN DẠNG MẪU

Nhận dạng mẫu là lĩnh vực khoa học với mục đích phân loại và mô tả các đối tượng. Tùy thuộc vào các ứng dụng, các đối tượng này có thể là chữ viết, ảnh, sóng âm thanh, v.v.. Trong chương này phần 1.1 dành để giới thiệu tóm tắt khái niệm nhận dạng mẫu và các bài toán cơ bản. Phần 1.2 giới thiệu một ví dụ về bài toán nhận dạng mẫu. Phần 1.3 giới thiệu các lĩnh vực liên quan. Các hệ thống nhận dạng mẫu được giới thiệu trong phần 1.4. Chu trình thiết kế bộ phân lớp được giới thiệu trong phần 1.5 kết luận được trình bày trong phần 1.6.

1.1. Mẫu và các bài toán nhận dạng thường gặp

1.1.1. Mẫu (pattern)

Có thể phân làm hai hoại: mẫu trừu tượng và mẫu cụ thể. Các ý tưởng, lập luận và khái niệm... là những ví dụ về mẫu trừu tượng, nhận dạng các mẫu như vậy thuộc về lĩnh vực nhận dạng khái niệm.

Các mẫu cụ thể bao gồm các đối tượng có tính không gian, thời gian và hình ảnh.. hoặc các đối tượng vật lý, chữ ký, chữ viết, ký hiệu, ảnh, đoạn sóng âm thanh, điện não đồ hoặc điện tâm đồ, hàm số...là những ví dụ về mẫu cụ thể.

1.1.2. Nhận dạng mẫu là gì?

Không có một định nghĩa thống nhất nào về nhận dạng mẫu (Pattern recognition viết tắt là PR) nhưng điều này cũng không gây ra tranh cãi gì trong giới nghiên cứu. Sau đây là một số định nghĩa theo ngữ cảnh nghiên cứu:

- Duda et al: Nhận dạng mẫu là việc quy những đối tượng vật lí hay sự kiện vào một loại (nhóm) nào đó đã xác định từ trước.

- Jürgen Schürmann: Nhận dạng mẫu là việc gán nhãn w cho một quan sát x.

- Selim Aksoy: Nhận dạng mẫu là việc nghiên cứu cách làm cho một máy có thể thực hiện:
  + Quan sát môi trường.
  + Học cách phân biệt được các mẫu cần quan tâm.
  + Đưa ra các quyết định đúng đắn về loại (nhóm) của các mẫu.

Như vậy thay cho việc tìm định nghĩa chính xác cho khái niệm nhận dạng mẫu ta sẽ liệt kê các bài toán chính trong lĩnh vực này.

1.1.3. Các bài toán nhận dạng mẫu thường gặp

Các bài toán nhận dạng mẫu thường gặp có thể quy về các dạng sau.

- Phân lớp có giám sát hay phân loại (classify): Dựa trên một tập con (tập đào tạo) đã biết nhãn, đưa ra một cách gán nhãn cho các đối tượng mới để phân tập các đối tượng thành các lớp. Ví dụ: nhận dạng chữ viết tay nhờ các chữ đã biết.

- Phân lớp không giám sát hay phân cụm (cluster): Chia tập đối tượng thành nhóm sao cho các đối tượng trong mỗi nhóm tương đối giống nhau còn các đối tượng khác nhóm thì khác nhau.
- Phân tích hồi quy (regression) hay nhận dạng hàm: Xác định một biến (hàm) qua tập các biến khác.
- Nhận thực (Identify): Xác định đối tượng trong tập đã cho có là đối tượng đang quan tâm hay không. Chẳng hạn như nhận thực vân tay, nhận thực mặt người...
- Mô tả: Mô tả các đối tượng dưới hình thức dễ phân tích. Chẳng hạn mô tả điện tâm đồ dưới dạng biểu đồ đặc trưng hoặc xâu mã.

Để hiểu rõ hơn quá trình nhận dạng mẫu, ta xét ví dụ sau.

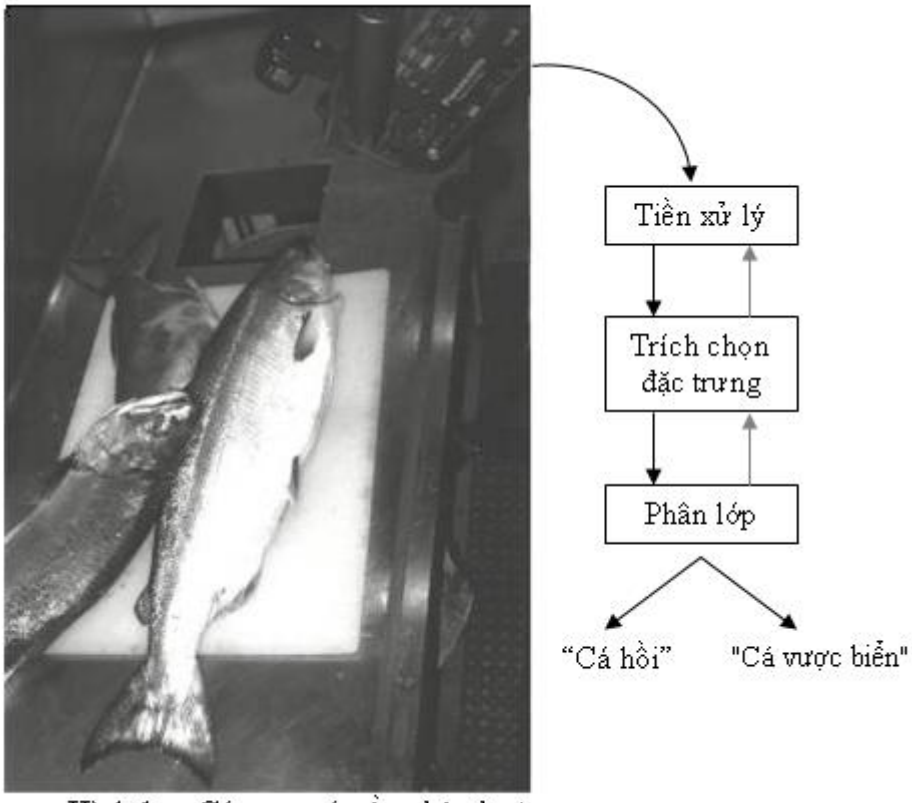
1.2. Ví dụ về bài toán nhận dạng mẫu

Giả sử ta muốn tự động hóa quá trình sắp xếp, hay phân lớp những con cá được nhập vào trên băng truyền dựa theo loài. Với dự án nhỏ, chúng ta cần phân biệt giữa cá hồi (salmon) và cá vược biển (sea bass). Ta thiết lập 1 máy ghi hình (sensor: cảm biến quang học), lấy một số mẫu và bắt đầu ghi chép một số đặc tính vật lý khác nhau giữa 2 loài cá như tính nhanh nhẹn, chiều rộng, số lượng và hình dáng của vây, vị trí của miệng, và tiếp tục sử dụng các đặc trưng này để dùng trong việc phân lớp. Chúng ta cũng phải chú ý đến sự biến đổi giữa các hình ảnh – sự biến đổi về độ sáng, vị trí của con cá trên băng truyền hay ngay cả vị trí của máy ghi hình.

Chắc chắn là số lượng cá hồi và cá vược sẽ khác nhau, chúng ta xem chúng như mỗi mô hình khác nhau để có thể tạo được mô hình toán học. Vấn đề bao quát trong phân lớp mẫu là đưa ra một lớp các mô hình, xử lý dữ liệu để loại bỏ nhiễu (không phụ thuộc vào mô hình), và với mỗi mẫu chúng ta chọn ra mô hình thích hợp nhất. .

Hệ thống nguyên mẫu để thực hiện công việc rất đặc thù này được mô tả như hình 1. Đầu tiên máy ghi hình thu nhận hình ảnh của con cá. Sau đó tín hiệu từ máy ghi hình được đưa vào công đoạn *tiền xử lý* để đơn giản hóa các thao tác sau này mà không làm mất thông tin liên quan. Đặc biệt chúng ta có thể sử dụng thao tác phân đoạn (*segmentation*) để tách các bức ảnh của các loại cá khác nhau hay kể cả là ảnh nền. Thông tin từ mỗi con cá sau đó được đưa tới bộ trích chọn đặc trưng với mục đích là rút gọn dữ liệu bằng cách đánh giá các "đặc trưng' hay 'thuộc tính' nào đó có cần cho bộ phân lớp hay không. Những đặc trưng này (hay chính xác hơn là giá trị của

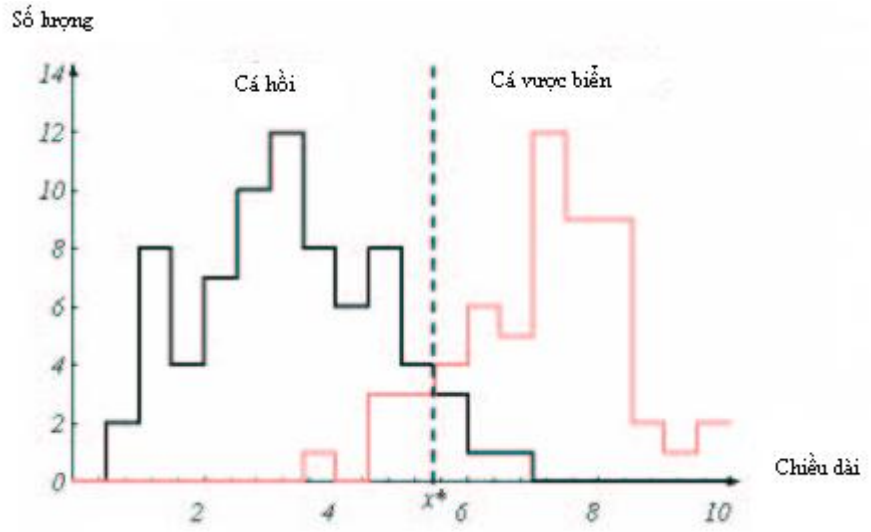
chúng) sau đó được chuyển cho bộ phân lớp để đánh giá các dấu hiệu và đưa ra quyết định cuối cùng về loại cá.



Hình 1 *: Các con cá cần phân loại.*

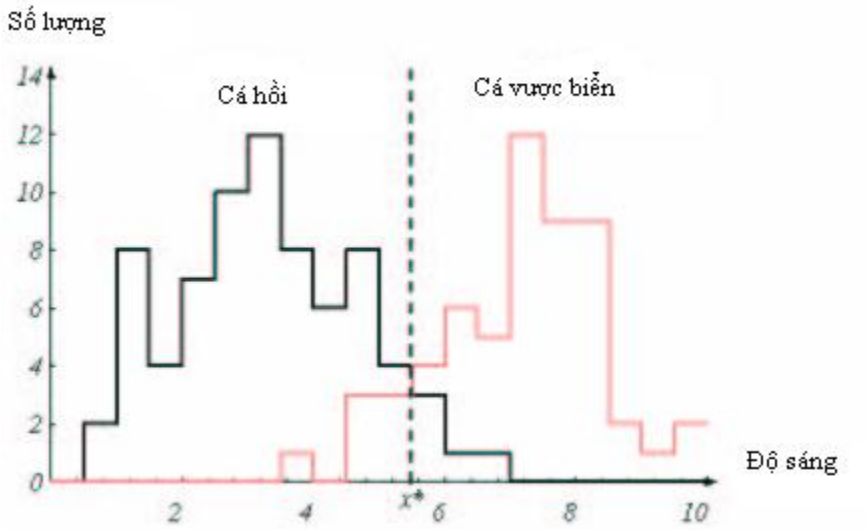
Bộ tiền xử lý sẽ tự động điều chỉnh độ sáng trung bình, hay loại bỏ hình nền của bức ảnh. Tại thời điểm này chúng ta hãy bỏ qua bước phân đoạn mà tập trung vào 2 bước là trích chọn đặc trưng và phân lớp. Giả sử rằng cá vược thường dài hơn cá hồi. Như vậy hiển nhiên chiều dài là một đặc trưng, và chúng ta có thể phân lớp cá bằng cách xem chiều dài của chúng có đạt độ dài L hay không. Để chọn giá trị của L chúng ta xem một vài con cá mẫu, tính giá trị độ dài và phân tích kết quả.

Giả sử rằng chúng ta thực hiện và thu được biểu đồ như hình 2. Biểu đồ này cho chúng ta thấy đúng là chiều dài trung bình của cá vược lớn hơn của cá hồi nhưng lại không có cách gì để chọn ra được một giá trị L khả dĩ để phân biệt chúng bằng chiều dài.



Hình 2. Biểu đồ về đặc trưng chiều dài của hai loại cá

Thật khó khăn, nhưng chúng ta sẽ tiếp tục với các đặc trưng khác như độ sáng trung bình. Bây giờ ta phải rất cẩn thận để loại trừ sự biến thiên của ánh sáng, bởi vì nó có thể làm hỏng bộ phân lớp mới của chúng ta. Kết quả và giá trị tối đa x* được thể hiện trên hình 3 đã thỏa mãn hơn. Các lớp đã được phân biệt tốt hơn.



Hình 3: Biểu đồ về đặc trưng độ sáng của hai loại cá.

Việc chọn yếu tố nào để quyết định sẽ đòi hỏi chi phí liên quan, và ta cần phải làm cho chi phí đó ở mức thấp nhất. Đây là nhiệm vụ trung tâm của lý thuyết quyết định trong đó phân lớp mẫu là lĩnh vực con quan trọng nhất.

Ngay cả khi chúng ta đã biết chi phí của các quyết định và chọn được giá trị x* tốt nhất, chúng ta vẫn có thể chưa thỏa mãn. Chúng ta muốn tìm các đặc trưng khác để phân lớp. Tuy nhiên không có đặc trưng trực quan riêng lẻ nào tốt hơn là độ sáng, vì vậy để tăng hiệu quả chúng, ta phải sử dụng nhiều hơn một đặc trưng để nhận dạng.

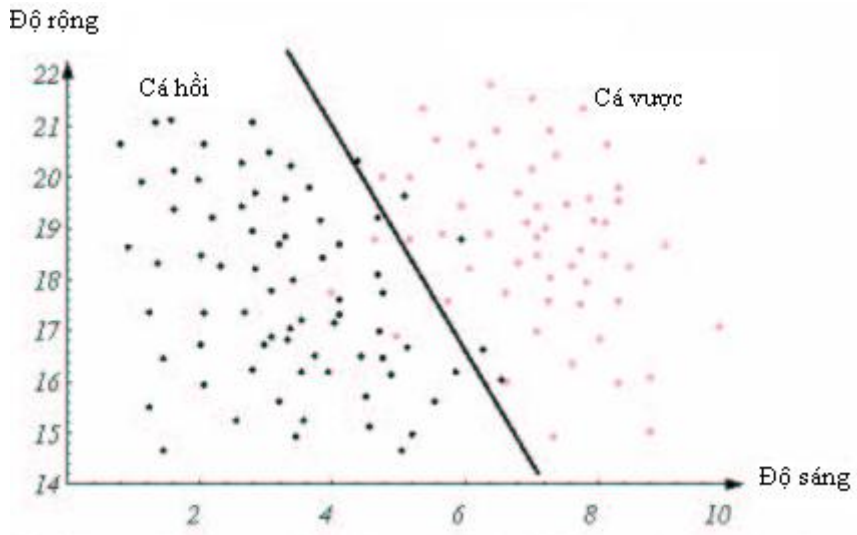
Khi tìm các đặc trưng khác chúng ta có thể thấy là cá vược thường có chiều rộng lớn hơn cá hồi. Bây giờ chúng ta có 2 đặc trưng để đánh giá – độ

sáng x_1 và chiều rộng x_2. Không tính đến thực tế chúng ta nhận ra rằng bộ trích chọn đặc trưng sẽ rút gọn mỗi bức ảnh về thành 1 điểm hay 1 *véc tơ* đặc trưng x trong không gian đặc trưng 2 chiều:

$$X = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

Bài toán của chúng ta là phải phân hoạch không gian đặc trưng thành 2 phần sao cho mọi điểm trong 1 vùng được coi là cá vược, và vùng còn lại là cá hồi. Sau khi xử lý chúng ta có thể có được minh họa như Hình 4. Đường kẻ gợi ý cho ta cách phân biệt các con cá: Quyết định một con cá là cá vược nếu vectơ đặc trưng của nó nằm dưới đường biên, ngược lại thì là cá hồi.

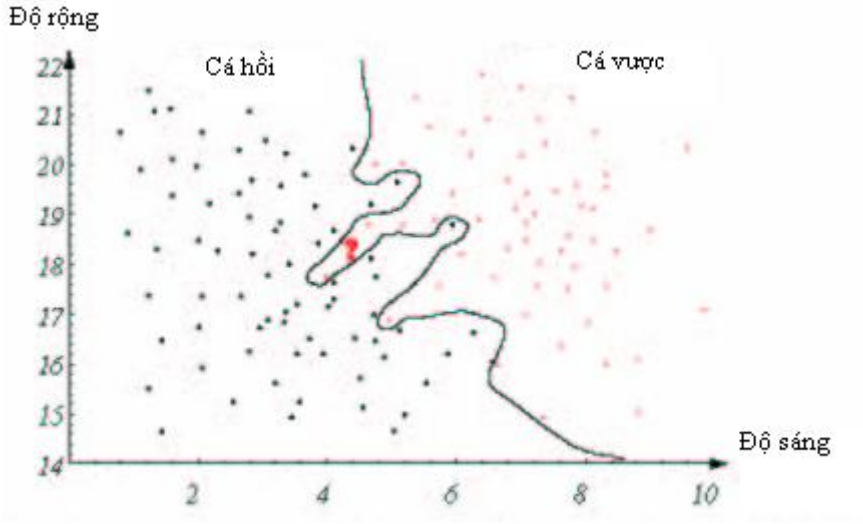
Luật này có vẻ thực hiện tốt và nó gợi ý cho chúng ta rằng có thể dùng thêm nhiều đặc trưng nữa. Bên cạnh độ sáng và chiều rộng, ta có thể cho thêm 1 vài tham số về hình dạng như góc nghiêng của vây ở lưng, hay vị trí của mắt, .v.v.. Nhưng làm sao chúng ta có thể biết trước là đặc trưng nào là thích hợp nhất. Một số đặc trưng có thể giảm bớt. Ví dụ như nếu màu của mắt cá có quan hệ chặt chẽ với chiều rộng thì hiệu quả của chương trình sẽ không tăng nếu ta sử dụng cả 2 đặc trưng, ngay cả khi chúng ta không phải lo lắng về việc tăng chi phí tính toán.. Tại sao chúng ta phải có quá nhiều đặc trưng, như vậy là tự làm khổ mình.



***Hình 4**: Hai đặc trưng về độ sáng và chiều rộng cho cá hồi và cá vược.*

Giả sử rằng các đặc trưng còn lại là rất khó đo, hoặc không giúp cải thiện tốc độ bao nhiêu, đôi khi còn làm giảm, và chúng ta quyết định sẽ dùng hai đặc trưng như trên hình 1.4, đường đậm chỉ ra một biên quyết định của bộ phân loại. Nếu những mô hình của ta phức tạp hơn thì đường biên sẽ là đường cong chứ không phải là đường thẳng như trên biểu đồ. Trong trường hợp đó tất cả các mẫu sẽ được phân loại như ở hình 1.5 dưới đây. Nhưng còn quá sớm để nói đến sự thành công vì mục đích của ta là phân lớp các mẫu mới, có thể rất kỳ lạ. Đó là sự tổng quát hóa, không chắc đường biên ở hình 1.5 đã cho kết quả tốt nhất, nó

có vẻ chỉ như là chia lại các mẫu huấn luyện chứ chưa phải là mô hình thật sự của bài toán.



Hình 5: Một mô hình phức tạp cho cá

Các mô hình quá phức tạp cho cá sẽ dẫn tới các biên quyết định trở nên phức tạp, nó sẽ làm cho các hệ thống tương lai chạy chậm.

Hơn nữa chúng ta có thể đơn giản hóa bộ nhận dạng, vì cũng không cần phải quá phức tạp như hình 5. Chúng ta cần hiệu quả khi chương trình chạy thật sự với các mẫu mới nên khi huấn luyện nếu kết quả có giảm sút một chút thì cũng không hề gì. Khi việc thiết kế các bộ nhận dạng quá phức tạp không đem lại hiệu quả thì tất nhiên ta sẽ ủng hộ cho một bộ phân lớp khác đơn giản hơn.

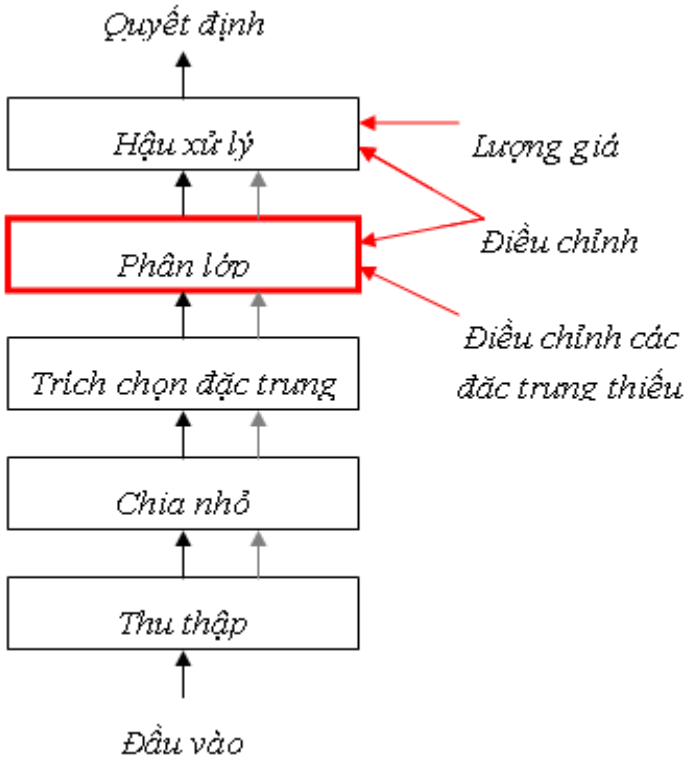
1.3. Các lĩnh vực liên quan

Với nhận dạng mẫu hiện nay, ba lĩnh vực nghiên cứu có quan hệ mật thiết nhất là: hồi quy, nội suy và ước lượng mật độ. Trong hồi quy chúng ta tìm kiếm một số mô tả chức năng của dữ liệu, thường với mục đích dự đoán giá trị cho đầu ra mới. Trong hồi quy tuyến tính hàm đó là hàm tuyến tính theo các biến đầu vào - là phổ biến nhất và là mô hình học tốt của hồi quy.

Trong nội suy, ta biết trước hoặc có thể dễ dàng suy ra hàm cho các mốc đã biết; vấn đề là cần tìm ra hàm cho các khoảng trung gian của đầu vào. Ước lượng mật độ là việc tính mật độ mà một thành viên của một loại bất kỳ sẽ được tìm thấy để có các đặc trưng cụ thể.

1.4. Các hệ thống nhận dạng mẫu

Trong hệ thống phân lớp cá mà đã mô tả ở trên, chúng ta đã phân biệt 3 thao tác khác nhau là xử lý, trích chọn đặc trưng và phân lớp. Hình 6 sẽ cho ta thấy sơ đồ chi tiết hơn của 1 hệ thống nhận dạng mẫu. Để hiểu được vấn đề của việc thiết kế 1 hệ thống thì chúng ta phải hiểu từng thành phần của nó. Hãy cùng nhau xem xét hoạt động của từng thành phần và tìm hiểu những yêu cầu có thể đặt ra.



Hình 6. Sơ đồ hệ thống nhận dạng mẫu thông dụng

*** Bước 1. Thu nhận dữ liệu:** Đầu vào của hệ thống nhận dạng mẫu thường là một loại thiết bị chuyển đổi như 1 máy ghi hình hay ghi âm (sensor) để thu nhận tín hiệu (dữ liệu). Vấn đề khó khăn là vì sự phụ thuộc vào đặc tính và khả năng của thiết bị - như băng thông, độ phân giải, độ méo, tỷ lệ nhiễu tín hiệu, v.v.. Và vì vậy vấn đề thiết kế sensor cho việc nhận dạng mẫu là vượt ra ngoài phạm vi của lận văn này.

*** Bước 2. Chia nhỏ và tạo nhóm (phân đoạn):** Ở trong ví dụ về phân lớp cá chúng ta đã ngầm giả sử rằng các con cá là tách biệt nhau, và có thể dễ dàng phân biệt trên băng truyền. Trong thực tế, các con cá có thể nằm sát nhau hoặc đè lên nhau, và hệ thống của chúng ta phải có khả năng xác định được từng con cá riêng biệt, việc xác định từng cá thể gọi là phân đoạn (segmentation). Nếu chúng ta đã nhận dạng được các con cá thì việc phân lập từng cá thể là tương đối dễ dàng nhưng vấn đề là ta phải thực hiện phân lập khi chưa biết chúng có những loại nào. Do đó chúng ta cần phải có cách để biết được khi nào thì chuyển từ mô hình này sang mô hình khác, hoặc phải biết đó chỉ là ảnh nền hay không có phân loại cho con cá đó.

Phân đoạn là một trong những bài toán khó trong nhận dạng mẫu. Trong hệ thống tự động nhận dạng tiếng nói, chúng ta phải cố gắng nhận ra từng âm riêng biệt (ví dụ như các âm “ss”, ”k”,…) và sau đó kết hợp chúng với nhau để tạo thành từ cần nhận dạng. Nhưng hãy thử xem xét trường hợp hai từ, ‘sklee’ và ‘skloo’. Nói chúng lên và nhận thấy rằng: với từ ‘skloo’ bạn đẩy lưỡi lên phía trước trước khi thốt ra từ ‘ss’.

Liên quan chặt chẽ tới việc phân đoạn là bài toán nhận dạng hoặc nhóm nhiều đối tượng liên quan lại với nhau. Ta dễ dàng đọc từ BEATS, nhưng tại sao

ta không chọn từ khác cho các tập con của đoạn mẫu này, như BE, BEAT, EAT, AT và EATS? làm sao chúng ta có thể thực hiện công việc này một cách tự động?

*** Bước 3. Trích chọn đặc trưng**: Ranh giới về mặt khái niệm giữa việc trích chọn đặc trưng và phân lớp ở mức độ nào đó có phần không rõ ràng. Một bộ trích chọn đặc trưng lý tưởng phải làm cho công việc còn lại của bộ phân lớp trở nên dễ dàng nhưng ngược lại, một bộ phân lớp có thể không cần tới bộ trích chọn đặc trưng phức tạp. Nhưng đó là vấn đề để thực hành chứ không chỉ là lý thuyết.

Mục tiêu chung của bộ trích chọn đặc trưng là mô tả các đối tượng để có thể đo được bằng các giá trị của chúng mà các giá trị đó là xấp xỉ nhau với các đối tượng thuộc cùng loại và khác xa nhau với các đối tượng không cùng loại. Điều này dẫn đến việc phải tìm ra các đặc trưng khác nhau và chúng phải không đổi với các cá thể khác nhau. Như trong ví dụ phân lớp cá thì vị trí tuyệt đối của con cá trên băng truyền là không liên quan đến loại cá, do đó chúng ta không cần quan tâm đến vị trí của các con cá. Trong trường hợp lý tưởng thì ta muốn các đặc trưng phải không thay đổi cho dù ta xoay ngang hay dọc. Bởi vì việc nhận dạng cũng như một số đặc trưng khác phải không bị ảnh hưởng bởi chuyển động quay. Cuối cùng thì kích thước của cá cũng không làm ảnh hưởng đến việc nhận dạng – một con cá hồi dù bé dù nhỏ thì vẫn là một con cá hồi. Ngoài ra chúng ta còn muốn các đặc trưng không bị thay đổi khi điều chỉnh (scale). Nói chung thì các đặc trưng như hình dạng, màu sắc và các đặc tính bề mặt là không đổi khi dịch chuyển, quay hay điều chỉnh.

Một lượng lớn các biến đổi phức tạp được thực hiện trong nhận dạng mẫu. Chúng ta có thể làm cho bộ nhận dạng chữ viết tay không nhạy cảm với độ dày tổng thể của ngòi bút nhờ các biến đổi như vậy.

Khi đi cùng với bộ phân lớp cụ thể, bộ trích chọn đặc trưng cần phù hợp với nhiệm vụ phân lớp. Một bộ trích chọn đặc trưng tốt cho việc sắp xếp cá có thể không được dùng nhiều trong xác định dấu vân tay hay phân loại ảnh chụp dưới kính hiển vi của các tế bào máu. Tuy nhiên một số nguyên lý chung cho phân lớp mẫu có thể được dùng để thiết kế bộ trích chọn đặc trưng cho các bộ phân lớp.

*** Bước 4. Phân lớp:** Nhiệm vụ của bước này trong hệ thống là sử dụng các véc tơ đặc trưng được cung cấp từ bước trích chọn đặc trưng để gắn các đối tượng vào các lớp. Luận văn này quan tâm đến thiết kế bộ phân lớp. Độ khó của bài toán phân lớp phụ thuộc vào sự biến thiên đặc trưng của đối tượng trong cùng một lớp, sự khác biệt giữa nó với đặc trưng của đối tượng trong các lớp khác.

Một bài toán nữa là có thể không xác định được hết các đặc trưng của dữ liệu vào. Trong ví dụ của chúng ta thì có thể có con cá không xác định được chiều rộng vì bị che bởi con khác. Khi bộ nhận dạng cần hai đặc trưng mà chỉ có được một thì làm sao có thể có được quyết định chính xác. Phương pháp tự nhiên là sẽ gán cho giá trị của đặc trưng bị thiếu bằng không hoặc bằng giá trị trung bình của các mẫu đã biết, đây sẽ là điều làm cho chương trình không tối

ưu. Mặt khác làm sao ta có thể huấn luyện bộ phân lớp khi mà không có đủ các đặc trưng.

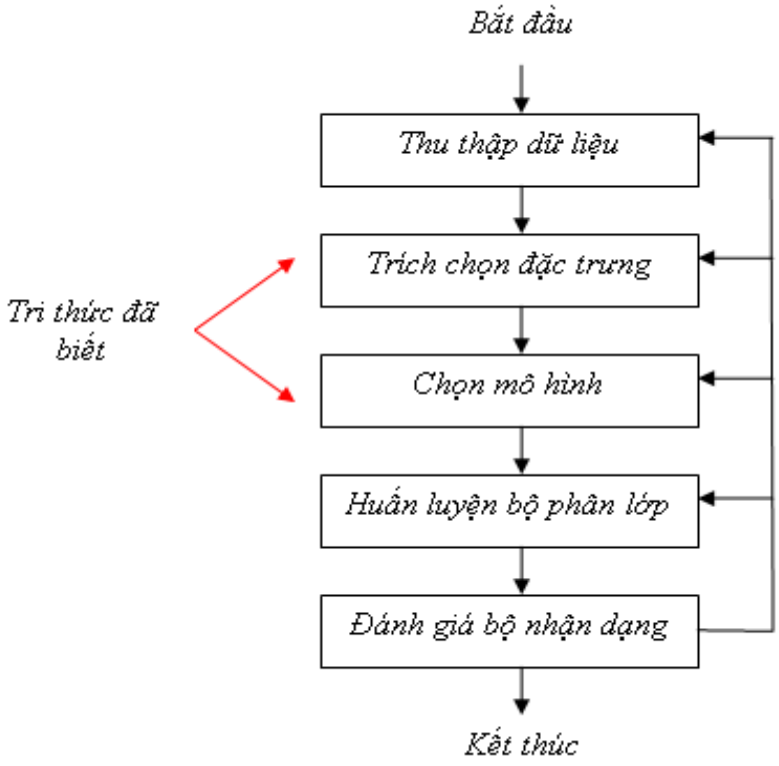
*** Bước 5. Hậu xử lý:** Một bộ phân lớp hiếm khi chỉ để dùng đơn lẻ. Thay vào đó nó thường dùng để đưa ra thao tác tương ứng (đặt con cá này vào giỏ này, đặt con cá khác vào giỏ kia), mỗi thao tác mất một chi phí tương ứng. Hậu xử lý sẽ dùng đầu ra của bộ phân lớp để quyết định thao tác tương ứng.

Theo quan niệm, cách đơn giản nhất để đánh giá hoạt động của một bộ phân lớp là xem tỷ lệ nhận dạng sai với các mẫu mới. Do đó chúng ta cần phải nhận dạng với tỷ lệ lỗi thấp nhất. Tuy nhiên chúng ta cần các thao tác tương ứng phải làm cho tổng chi phí là thấp nhất. Có thể phải kết hợp các tri thức đã biết về chi phí, và nó sẽ có ảnh hưởng đến việc ra các quyết định hành động. Chúng ta cũng cần ước lượng trước chi phí để xem có thỏa mãn hay không. Liệu chúng ta có thể tính được chi phí thấp nhất với mọi bộ phân lớp, để quyết định có thực thi hay không.

Trong ví dụ về phân lớp cá chúng ta đã thấy việc sử dụng nhiều đặc trưng có thể giúp tăng tốc độ thực hiện phân lớp. Do đó có thể kết luận rằng việc thiết lập nhiều bộ nhận dạng có thể làm tăng hiệu quả phân lớp; như việc dùng một bộ nhận dạng âm học và một bộ nhận dạng môi đọc trong nhận dạng tiếng nói. Thực tế không phải lúc nào cũng như vậy.

1.5. Chu trình thiết kế bộ phân lớp

Thiết kế của một hệ thống nhận dạng mẫu thường đòi hỏi sự lặp lại một số thao tác khác nhau: thu thập dữ liệu, lựa chọn đặc trưng, lựa chọn mô hình, huấn luyện, và đánh giá. Trong phần này giới thiệu một cái nhìn toàn cảnh về chu trình thiết kế một bộ phân lớp (Hình 7) và xem xét một số bài toán hay gặp.



Hình 7. Chu trình thiết kế một bộ phân lớp

*** Bước 1. Thu thập dữ liệu**: Bước này có thể chiếm một phần rất lớn trong chi phí phát triển một hệ thống nhận dạng mẫu. Để thực hiện nghiên cứu thì có thể thực hành sơ bộ với tập nhỏ các mẫu dữ liệu điển hình, nhưng để hệ thống có thể hoạt động tốt thì phải cần rất nhiều dữ liệu. Làm sao chúng ta có thể biết dữ liệu thu thập được lớn thế nào là đủ cho việc huấn luyện và việc kiểm tra hệ thống? Chi phí thu thập dữ liệu thường khá lớn và tốn thời gian.

*** Bước 2. Chọn đặc trưng:** Lựa chọn các đặc trưng khác nhau là một khâu then chốt khi thiết kế và phụ thuộc vào tính chất của bài toán. Xem xét dữ liệu mẫu (ảnh cá trên dây truyền) sẽ có giá trị cho việc chọn tập đặc trưng. Tuy nhiên các hiểu biết trước đó cũng có một vai trò quan trọng. Trong ví dụ phân lớp cá của chúng ta, những tri thức đã biết về độ sáng của các loại cá đã giúp chúng ta thiết kế bộ phân lớp bằng cách sử dụng đặc trưng có triển vọng. Sử dụng các tri thức đã biết là việc làm tinh tế và khó khăn.

Khi lựa chọn các đặc trưng, chúng ta rõ ràng muốn tìm các đặc trưng có thể dễ dàng trích chọn, không bị ảnh hưởng khi thay đổi, không nhạy cảm với nhiễu, và hữu dụng trong việc phân lớp.

*** Bước 3. Lựa chọn mô hình:** Chúng ta có thể không bằng lòng với hoạt động của bộ nhận dạng cá ở hình 4 và hình 5, vì vậy chúng ta sẽ chuyển sang mô hình khác, ví dụ như mô hình dựa trên số lượng, hình dáng vây, màu mắt, trọng lượng, hình dạng miệng, v.v.. Nhưng làm sao ta có thể biết được mô hình dự đoán của ta và mô hình thật sự có sự sai khác đáng kể nào hay không, và liệu có cần thay đổi mô hình hay không? Tóm lại làm sao để chúng ta biết được là cần phải thay đổi mô hình? Liệu chúng ta có phải thử hết các mô hình để tìm ra mô hình tối ưu hay không? Trả lời các câu hỏi này là bước lựa chọn mô hình.

*** Bước 4. Huấn luyện**: Quá trình sử dụng dữ liệu để xây dựng bộ phân lớp được gọi là huấn luyện bộ phân lớp. Trên đây ta đã thấy rất nhiều bài toán nảy sinh khi thiết kế một bộ phân lớp. Tuy nhiên, không có phương thức tổng quát để giải quyết các bài toán này.

*** Bước 5. Đánh giá**: Khi chúng ta chuyển từ việc sử dụng một đặc trưng sang sử dụng hai đặc trưng trong việc phân lớp cá, kết quả là tốt hơn. Khi chúng ta chuyển từ bộ phân lớp tuyến tính sang mô hình phức tạp hơn kết quả lại có thể tốt hơn.Việc đánh giá là quan trọng trong việc tăng hiệu quả và tốc độ hoạt động của hệ thống khi xem xét việc cải tiến các thành phần.

*** Bước 6. Đánh giá độ phức tạp tính toán:** Một số bài toán nhận dạng mẫu có thể được giải bằng cách sử dụng các giải thuật không thực tế lắm do đòi hỏi sử dụng thời gian tính toán lớn trong khi phải xử lý một khối lượng dữ liệu lớn. Vì vậy tùy theo bài toán cụ thể mà ta cần quan tâm tới độ phức tạp của thuật toán để điều chỉnh thời gian xử lý dữ liệu.

1.6. Kết luận

Trên đây ta đã hình dung được quá trình nhận dạng mẫu và thiết kế một hệ nhận dạng mẫu. Khi xây dựng hệ nhận dạng có giám sát, việc thu thập các dữ liệu đào tạo thường đòi hỏi nhiều thời gian và tốn nhiều chi phí, mà trong thực tế

dữ liệu lại tồn tại phần lớn là dữ liệu chưa gán nhãn. Luận văn này sẽ đi sâu vào một cách tiếp cận mới để khắc phục khó khăn này.

CHƯƠNG 2. GIỚI THIỆU VỀ HỌC BÁN GIÁM SÁT VÀ PHƯƠNG PHÁP ĐỒNG HUẤN LUYỆN

Chương này dành giới thiệu về học bán giám sát và phương pháp đồng huấn luyện (để đơn giản và thống nhất trong toàn luận văn từ nay ta đề cập tới với tên gọi co-training). Phần 2.1 được dành giới thiệu chung về phương pháp học bán giám sát. Phần 2.2 giới thiệu phương pháp tự huấn luyện (self-training). Phần 2.3 giới thiệu về phương pháp đồng huấn luyện co-training. So sánh sự giống và khác nhau giữa hai phương pháp tự huấn luyện và đồng huấn luyện được đưa ra trong phần 2.4.

2.1. Phương pháp học bán giám sát

Phương pháp học có giám sát (supervised learning) truyền thống là phương pháp học chỉ dựa trên các dữ liệu đã gán nhãn sẵn có, do đó để xây dựng được một bộ phân lớp có độ tin cậy cao đòi hỏi phải có một số lượng lớn các mẫu huấn luyện (các dữ liệu đã được gán nhãn lớp đúng). Tuy nhiên, trong thực tế để có được các mẫu này cần rất nhiều công sức, thời gian và chi phí của con người.

Ví dụ với bài toán học để nhận biết được những bài báo, nhóm tin tức UseNet nào mà người dùng quan tâm. Khi đó hệ thống phải lọc, sắp xếp trước các bài báo và chỉ đưa ra các bài báo mà có nhiều người dùng quan tâm nhất – một bài toán đang thu hút được sự chú ý ngày nay. Lang [27] đã phát hiện ra rằng sau khi một người đọc và gán nhãn khoảng 1000 bài báo, một bộ phân lớp được huấn luyện qua chúng sẽ thu được độ chính xác khoảng 50% trong khi dự đoán chỉ có 10% các bài báo có độ tin cậy cao. Tuy nhiên, hầu hết người sử dụng hệ thống thực sẽ không có đủ kiên nhẫn để gán nhãn hàng nghìn bài báo chỉ để thu được độ chính xác trên. Do đó vấn đề đặt ra là xây dựng một thuật toán đưa ra sự phân lớp chính xác mà chỉ cần một số lượng nhỏ dữ liệu học, tức chỉ với vài chục bài báo được gán nhãn trước thay vì hàng nghìn bài báo.

Nhu cầu về một lượng lớn các dữ liệu học và những khó khăn để thu được các dữ liệu đó đặt ra một câu hỏi quan trọng: Liệu có thể sử dụng được nguồn thông tin nào khác trong phân lớp mà có thể làm giảm sự cần thiết của dữ liệu gán nhãn? Đây chính là nguồn động lực thúc đẩy sự phát triển của các phương pháp học bán giám sát *(semi-supervised learning).*

Sự tồn tại của dữ liệu trong thực tế thường là ở dạng trung gian: Không phải tất cả đều được gán nhãn cũng như không phải tất cả đều chưa được gán nhãn. Bán giám sát là một phương pháp học sử dụng thông tin từ cả hai nguồn dữ liệu này.

Để hiểu rõ hơn bản chất của học bán giám sát, chúng ta sẽ tìm hiểu thế nào là học có giám sát (*supervised learning*) và học không có giám sát (*unsupervised learning).*

2.1.1. Học có giám sát và học không có giám sát

Học có giám sát (hay còn gọi là học giám sát) là một kĩ thuật của ngành học máy để xây dựng một hàm (*function*) từ dữ liệu huấn luyện bao gồm các cặp gồm đối tượng đầu vào (thường dạng vec-tơ) và đầu ra mong muốn. Đầu ra của một hàm có thể là một giá trị liên tục (gọi là hồi quy), hay có thể là dự đoán một nhãn phân loại cho một đối tượng đầu vào (gọi là phân loại). 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 lý" nhất [50]. Liên quan nhiều nhất tới học giám sát là bài toán phân lớp (Classification).

Học không có giám sát là một phương pháp trong học máy, nhằm tìm ra một mô hình phù hợp nhất với các quan sát. Nó khác biệt với học có giám sát ở chỗ là đầu ra đúng tương ứng cho mỗi đầu vào là không biết trước. Trong học không có giám sát, một tập dữ liệu đầu vào được thu thập và các đối tượng đầu vào được coi như là một tập các biến ngẫu nhiên. Sau đó, một mô hình mật độ kết hợp sẽ được xây dựng cho tập dữ liệu đó. Học không có giám sát được đề cập tới với bài toán phân mảnh hay phân cụm dữ liệu (data clustering) [50]. Tóm lại, học không có giám sát là việc học trên tập dữ liệu chưa được biết trước thông tin với mục đích là tìm ra được mô hình phù hợp nhất với các quan sát đó, hay là quá trình nhóm (phân cụm) các đối tượng giống nhau lại với nhau.

Từ đó, ***học bán giám sát*** có thể được xem là:

- Học giám sát cộng thêm dữ liệu chưa gán nhãn (Supervised learning + additional unlabeled data).
- Học không giám sát cộng thêm dữ liệu gán nhãn (Unsupervised learning + additional labeled data).

Học bán giám sát chính là cách học sử dụng thông tin chứa trong cả dữ liệu chưa gán nhãn và dữ liệu đã được gán nhãn (tập dữ liệu huấn luyện). Các thuật toán học bán giám sát có nhiệm vụ chính là mở rộng dần tập các dữ liệu gán nhãn ban đầu thông qua việc khai thác thông tin từ các dữ liệu chưa gán nhãn. Hiệu quả của thuật toán phụ thuộc vào chất lượng của các dữ liệu được gán nhãn trung gian và thêm vào ở mỗi vòng lặp.

2.1.2. Động lực thúc đẩy và hiệu quả của học bán giám sát

Đã có rất nhiều các nghiên cứu về học bán giám sát. Những kết quả thực nghiệm cũng như lý thuyết đã chỉ ra rằng sử dụng cách tiếp cận đánh giá cực đại khả năng (Maximum Likelihood) có thể cải tiến độ chính xác phân lớp khi có thêm các dữ liệu chưa gán nhãn [28].

Tuy nhiên, cũng có những nghiên cứu chỉ ra rằng, dữ liệu chưa gán nhãn có thể cải tiến độ chính xác phân lớp hay không là phụ thuộc vào cấu trúc bài toán có phù hợp với giả thiết của mô hình hay không? Cozman [20] đã thực nghiệm trên dữ liệu giả hướng vào tìm

hiểu giá trị của dữ liệu chưa gán nhãn. Ông chỉ ra rằng, độ chính xác phân lớp có thể giảm đi khi thêm vào ngày càng nhiều dữ liệu chưa gán nhãn. Ông cũng đã tìm ra nguyên nhân của sự giảm này là do sự không phù hợp giữa giả thiết của mô hình và phân phối dữ liệu thực tế.

Để việc học bán giám sát mang lại hiệu quả cần một điều kiện tiên quyết là: Phân phối các mẫu cần phát hiện phải phù hợp với bài toán phân lớp [36]. Về mặt công thức, các tri thức thu được từ dữ liệu chưa gán nhãn $p(x)$ phải mang lại thông tin hữu ích cho suy luận $p(x|y)$. Olivier Chapelle [36] đã đề xuất một giả thiết làm trơn, đó là hàm nhãn lớp ở vùng có mật độ cao thì trơn hơn ở vùng có mật độ thấp. Giả thiết được phát biểu như sau:

Giả thiết bán giám sát: *Nếu hai điểm* x_1, x_2 *thuộc vùng có mật độ cao là gần nhau thì đầu ra tương ứng của chúng là* y_1, y_2 *cũng gần nhau..*

2.1.3. Phạm vi sử dụng học bán giám sát

Các phương pháp học bán giám sát sẽ rất hữu ích khi dữ liệu chưa gán nhãn nhiều hơn dữ liệu gán nhãn. Việc thu được dữ liệu chưa gán nhãn thì dễ, nhưng để gán nhãn chúng thì tốn rất nhiều thời gian, công sức và tiền bạc. Đó là tình trạng của rất nhiều các lĩnh vực ứng dụng trong học máy như:

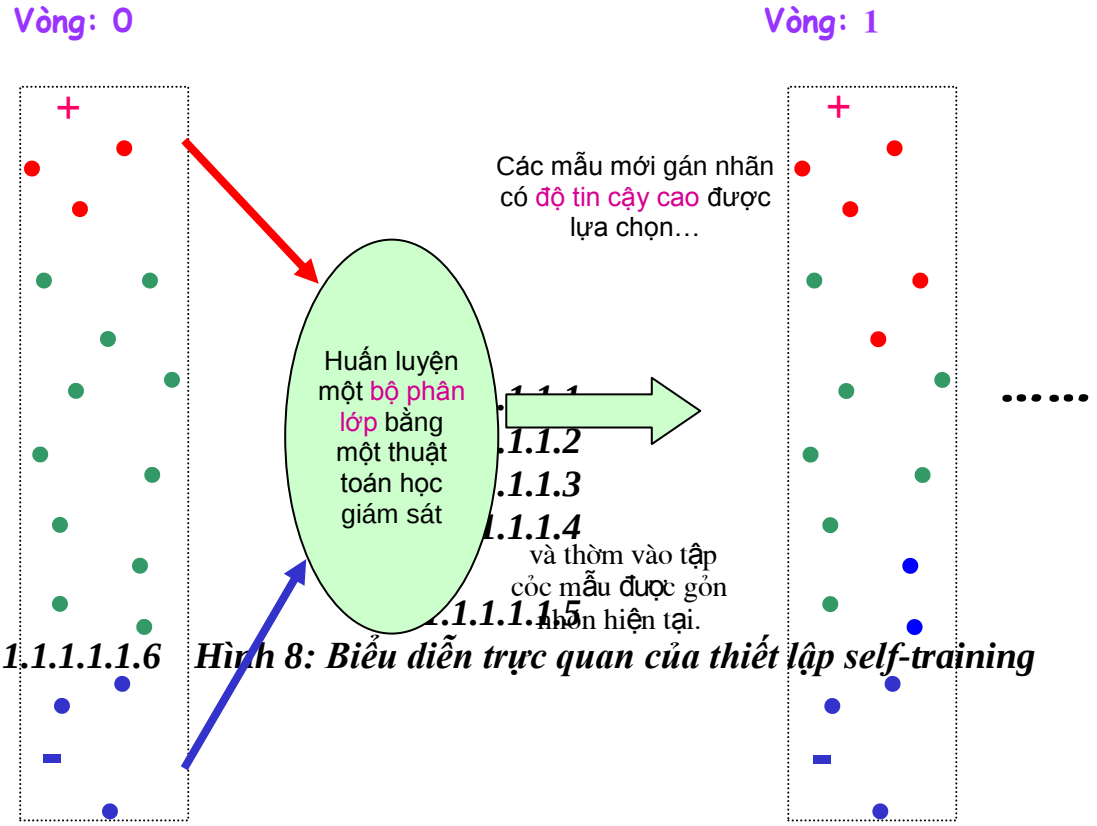
- Trong nhận dạng lời nói, ta sẽ dễ dàng ghi lại một lượng lớn các bài diễn thuyết, nhưng để gán nhãn chúng yêu cầu con người phải lắng nghe rồi đánh máy sao chép lại.
- Sự phong phú của hàng tỉ các trang web sẵn sàng cho xử lý tự động, nhưng để phân lớp chúng một cách tin cậy đòi hỏi con người phải đọc chúng.

Học bán giám sát là việc học trên cả dữ liệu đã và chưa được gán nhãn. Từ một số lượng lớn các dữ liệu chưa được gán nhãn và một luợng nhỏ dữ liệu đã được gán nhãn ban đầu (thường gọi là *seed set*) để xây dựng một bộ học thậm chí là tốt hơn. Trong quá trình học như thế phương pháp sẽ tận dụng được những thông tin phong phú của dữ liệu chưa gán nhãn (*unlabeled data)* mà chỉ yêu cầu một số lượng rất nhỏ các dữ liệu đã được gán nhãn ban đầu (*labeled data).* Song, ý tưởng chung là kết quả thu được phải tốt như đối với việc học trên một tập dữ liệu lớn đã được gán nhãn. Và trong thực tế ta hoàn toàn có thể hy vọng vào sự chính xác của dự đoán khi xét thêm các điểm không gán nhãn với những giả thiết phù hợp *(certain assumptions)* cho từng mô hình [36].

Có rất nhiều phương pháp học bán giám sát nên trước khi quyết định lựa chọn phương pháp học cho một bài toán cụ thể cần phải xem xét các giả thiết của mô hình. Theo Zhu [47], chúng ta nên sử dụng phương pháp học mà giả thiết trên dữ liệu của nó phù hợp với cấu trúc của bài toán. Để hiểu kĩ hơn về học bán giám sát, dưới đây giới thiệu hai phương pháp học bán giám sát điển hình nhất, phương pháp tự huấn luyện (self-training) và phương pháp đồng huấn luyện (co-training).

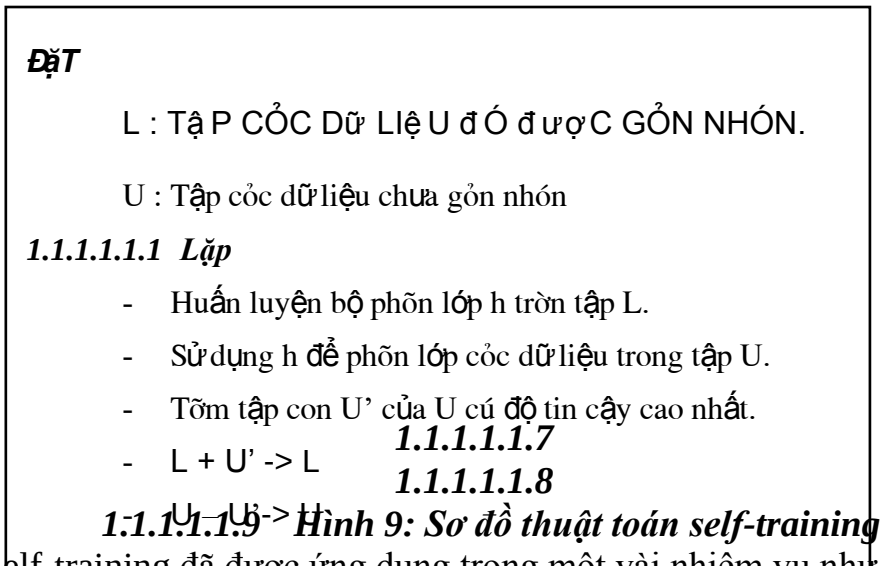
2.2. Phương pháp tự huấn luyện

Ý tưởng đầu tiên về sử dụng dữ liệu chưa gán nhãn trong phân lớp là thiết lập tự huấn luyện (self-training) [36]. Lần đầu tiên xuất hiện là từ những năm 1969 với thuật toán bọc *(wrapper-algorithm)* sử dụng lặp nhiều lần một phương pháp học có giám sát. Hình 8 dưới đây biểu diễn một cái nhìn trực quan của thiết lập self-training.



1.1.1.1.1.6 Hình 8: Biểu diễn trực quan của thiết lập self-training

Self-training là kỹ thuật học bán giám sát được sử dụng rất phổ biến. Ý tưởng là: một bộ phân lớp (*classifier*) ban đầu được huấn luyện bằng một số lượng nhỏ các dữ liệu đã gán nhãn (tập dữ liệu huấn luyện mẫu). Sau đó, sử dụng bộ phân lớp này để gán nhãn các dữ liệu chưa gán nhãn. Các dữ liệu được gán nhãn có độ tin cậy cao (vượt trên một ngưỡng nào đó) và nhãn tương ứng của chúng được đưa vào tập huấn luyện. Tiếp đó, bộ phân lớp được học lại trên tập huấn luyện mới ấy và thủ tục lặp tiếp tục. Thuật toán self-training sẽ dừng nếu nó thoả mãn điều kiện: Hoặc số vòng lặp đạt tới một số đã được xác định trước hoặc là khi tập dữ liệu chưa gán nhãn là rỗng. Tên gọi self-training xuất phát từ việc nó sử dụng dự đoán của chính nó để dạy chính nó. Sơ đồ thuật toán self-training được mô tả như hình 9.



Hình 9: Sơ đồ thuật toán self-training

Self-training đã được ứng dụng trong một vài nhiệm vụ như xử lý ngôn ngữ tự nhiên: Riloff, Wiebe và Wilson (2003) [18] sử dụng self-training để xác định các danh từ có thuộc quan điểm cá nhân hay không... ngoài ra self-training cũng được ứng dụng trong phân tích cú pháp và dịch máy.

2.3. Phương pháp đồng huấn luyện

2.3.1. Thiết lập đồng huấn luyện

Thiết lập ban đầu của đồng huấn luyện (co-traning) là cùng lúc huấn luyện hai bộ học trên cùng một bộ dữ liệu nhưng với hai thuật toán khác nhau. Với hai thuật toán riêng, mỗi cách huấn luyện tạo nên một "bộ học" độc lập (*independent*) và "đầy đủ" (sufficient) với bộ còn lại để sử dụng, tức là chỉ với một trong hai bộ học này ta có thể giải quyết được bài toán (ví dụ với bài toán phân lớp, một bộ học là đủ để phân lớp đúng các dữ liệu). Ý tưởng của co-training là sau mỗi bước lặp, sử dụng kết quả tốt nhất, với độ tin cậy cao nhất của bộ học này đưa sang để "dạy" cho bộ học kia và ngược lại.

Đến năm 1998 khi nghiên cứu, khai thác việc sử dụng thông tin kết hợp cả dữ liệu có nhãn và dữ liệu chưa có nhãn trong bài toán học bán giám sát hai ông A. Blum và T. Mitchell [11] đã đưa ra lược đồ cụ thể của phương pháp co-training là dựa trên hai tập đặc trưng (set of features) độc lập và đầy đủ của dữ liệu để xây dựng nên hai bộ học. Hai tập đặc trưng này còn được gọi là hai

khung nhìn (views) của dữ liệu. Tính *độc lập* và *đầy đủ* để đảm bảo rằng với mỗi khung nhìn ta có thể xây dựng được một bộ học đúng cho bài toán. Khi đã xây dựng được song song hai bộ học thì kết quả tốt ở môic bước của bộ học này được sử dụng để huấn luyện lại bộ học kia và ngược lại.

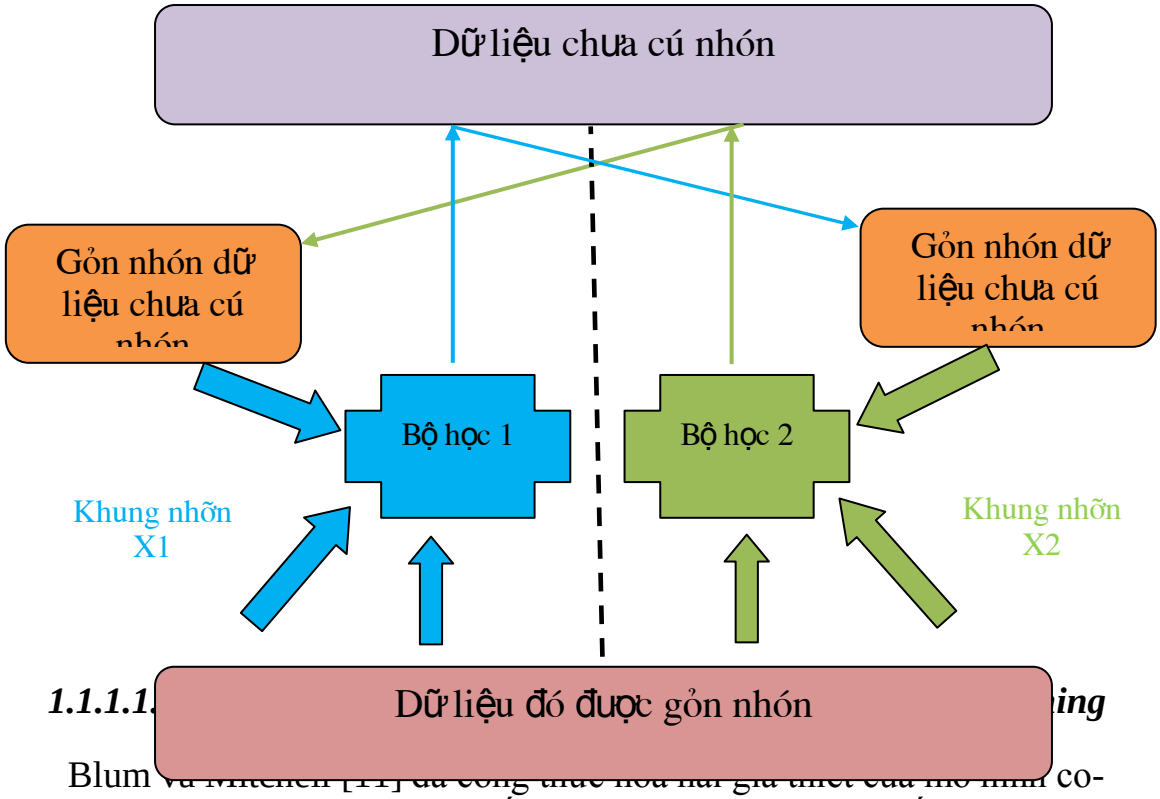
Năm 2000 khi nghiên cứu, phân tích hiệu quả và tính ứng dụng của co-training thì Nigam và Ghani [25,26] đã tập trung trả lời các câu hỏi như: tại sao những thuật toán co-training lại thành công, co-training có thực sự khai thác được sự phân tách độc lập của các đặc trưng hay không, khi không có sự phân tách tự nhiên (natural split) trên các đặc trưng của dữ liệu thì co-training có nên được áp dụng hay không….Hai ông đã kết luận rằng nếu tập dữ liệu thỏa mãn điều kiện có sự phân tách độc lập và đầy đủ trên hai khung nhìn của dữ liệu thì co-training thực thi tốt hơn những thuật toán học bán giám sát khác như phương pháp cực đại kì vọng (Expectation Maximization- EM) hay phương pháp tự huấn luyện self-training [25]. Ngoài ra khi thử với tập dữ liệu mà không biết trước sự phân tách tự nhiên trên các đặc trưng của nó, Nigam và Ghani [26] cũng chỉ ra rằng nếu con người có thể "tạo ra" một sự phân tách độc lập trên các đặc trưng của dữ liệu thì co-training vẫn có thể phát huy được hiệu quả của nó.

Với bài toán cụ thể khi giả thiết các đặc trưng có thể được phân chia thành 2 khung nhìn độc lập và đầy đủ thì thủ tục học co-training được tiến hành như sau:

- Huấn luyện 2 bộ phân lớp riêng rẽ bằng dữ liệu đã được gán nhãn trên hai tập thuộc tính con tương ứng. (Lúc đầu, đây là quá trình học có giám sát trên tập dữ liệu huấn luyện).
- Mỗi bộ phân lớp sau đó được dùng để phân lớp các dữ liệu mới chưa được gán nhãn (*unlabel data)*. Các *dữ liệu chưa gán nhãn + nhãn dự đoán* của chúng sẽ được lựa chọn nếu chúng có độ tin cậy cao dựa vào một ngưỡng nào đó, các dữ liệu này được dùng để dạy cho bộ phân lớp kia.
- Sau khi thêm một số dữ liệu mới vào tập mẫu huấn luyện, từng bộ phân lớp được huấn luyện lại *(retrain)* và tiến trình lặp bắt đầu.

Những ý tưởng về sử dụng sự dư thừa đặc trưng đã được thi hành trong một vài nghiên cứu. Yarowsky đã sử dụng co-training để tìm nghĩa cho từ vựng, ví dụ quyết định xem từ "plant" trong một ngữ cảnh cho trước có nghĩa là một sinh vật sống hay là một xí nghiệp. Yarrowsky [14] tiến hành tìm nghĩa của từ bằng cách xây dựng một bộ phân lớp nghĩa *(sense classifier)* sử dụng ngữ cảnh địa phương của từ và một bộ phân lớp nghĩa dựa trên nghĩa của những lần xuất hiện khác trong cùng một văn bản; Riloff và Jones [16] phân lớp cụm danh từ chỉ vị trí địa lý bằng cách xem xét chính cụm danh từ đó và ngữ cảnh ngôn ngữ mà cụm danh từ đó xuất hiện; Collin và Singer [29] thực hiện phân lớp tên thực thể định danh sử dụng chính từ đó và ngữ cảnh mà từ đó xuất hiện; S.

Kiritchenko và S.Matwin [42] áp dụng co-training trong bài toán phân lớp thư điện tử. Sơ đồ co-training đã được sử dụng trong rất nhiều lĩnh vực như phân lớp, phân tích thống kê và xác định cụm danh từ v.v.. Hình 10 dưới đây cho chúng ta một cái nhìn trực quan của thiết lập co-training.



1.1.1.1. ing

Blum và Mitchell [11] đã công thức hóa hai giả thiết của mô hình co-training và chứng minh tính đúng đắn của mô hình dựa trên thiết lập học giám sát theo mô hình xấp xỉ đúng xác suất – Probably Approximately Correct (PAC) chuẩn [43].

Cho trước một không gian: $X = X_1 \times X_2$, ở đây X_1 và X_2 tương ứng với hai khung nhìn khác nhau của cùng một mẫu *(examples)*. Mỗi mẫu x vì vậy có thể được biểu diễn bởi một cặp (x_1, x_2). Chúng ta giả thiết rằng mỗi khung nhìn là đủ và phù hợp để phân lớp chính xác. Cụ thể, nếu D là một phân phối trên X, và C_1, C_2 là các lớp khái niệm *(concept classes)* được định nghĩa tương ứng trên X_1 và X_2; giả thiết rằng tất cả các nhãn trên các mẫu với xác suất lớn hơn không dưới phân phối D là trùng khớp với một hàm đích *(target function)* $f_1 \in C_1$, và cũng trùng khớp với hàm đích $f_2 \in C_2$. Nói cách khác, nếu f biểu diễn khái niệm đích kết hợp trên toàn bộ mẫu, thì với bất kỳ mẫu $x = x_1 \times x_2$ có nhãn l, ta có $f(x) = f_1(x_1) = f_2(x_2) = l$. Nghĩa là D gán xác suất bằng không cho mẫu (x_1, x_2) bất kỳ mà $f_1(x_1) \neq f_2(x_2)$.

- ***Giả thiết thứ nhất***: Tính tương thích *(compatibility)*

Với một phân phối D cho trước trên X, ta nói rằng hàm đích $f = (f_1, f_2) \in C_1 \times C_2$ là tương thích *(compatible)* với D nếu thoả mãn điều kiện: D

gán xác suất bằng không cho tập các mẫu (x_1, x_2) mà $f_1(x_1) \neq f_2(x_2)$. Nói cách khác, mức độ tương thích của một hàm đích $f = (f_1, f_2)$ với một phân phối D có thể được định nghĩa bằng một số $0 \leq p \leq 1$: $p = 1 - \Pr_D\left[(x_1, x_2) : f_1(x_1) \neq f_2(x_2)\right]$.

- ***Giả thiết thứ hai***: Độc lập điều kiện *(conditional independence assumption)*

Ta nói rằng hàm đích f_1, f_2 và phân phối D thoả mãn giả thiết độc lập điều kiện nếu với bất kỳ một mẫu $(x_1, x_2) \in X$ với xác suất khác không thì ta có:

$$\Pr_{(x_1,x_2)\in D}\left[x_1 = \hat{x}_1 \middle| x_2 = \hat{x}_2\right] = \Pr_{(x_1,x_2)\in D}\left[x_1 = \hat{x}_1 \middle| f_2(x_2) = f_2\left(\hat{x}_2\right)\right]$$

và tương tự,

$$\Pr_{(x_1,x_2)\in D}\left[x_2 = \hat{x}_2 \middle| x_1 = \hat{x}_1\right] = \Pr_{(x_1,x_2)\in D}\left[x_2 = \hat{x}_2 \middle| f_1(x_1) = f_1\left(\hat{x}_1\right)\right]$$

Hai ông đã chỉ ra rằng, cho trước một giả thiết độc lập điều kiện trên phân phối D, nếu lớp đích có thể học được từ nhiễu phân lớp ngẫu nhiên theo mô hình PAC chuẩn, thì bất kỳ một bộ dự đoán yếu ban đầu nào cũng có thể được nâng lên một độ chính xác cao tuỳ ý mà chỉ sử dụng các mẫu chưa gán nhãn bằng thuật toán co-training.
Hai ông cũng đã chứng minh tính đúng đắn của sơ đồ co-training bằng định lý sau:

Định lý (A.Blum & T. Mitchell).

Nếu C_2 có thể học được theo mô hình PAC với nhiễu phân lớp, và nếu giả thiết độc lập điều kiện thoả mãn, thì (C_1, C_2) có thể học được theo mô hình co-training chỉ từ dữ liệu chưa gán nhãn, khi cho trước một bộ dự đoán yếu nhưng hữu ích ban đầu $h(x_1)$.

(chi tiết hơn nữa về các bổ đề và chứng minh trong thiết lập co-training xin xem thêm trong [11]).

2.3.2. Sơ đồ thiết lập đồng huấn luyện

Sơ đồ thiết lập co-training cho bài toán hai lớp lúc đầu được A. Blum và T. Mitchell [11] xây dựng như hình dưới:

Cho trước: + L là tập cỏc mẫu huấn luyện đó gỏn nhón.

+ U là tập cỏc mẫu chưa gỏn nhón.

Lặp k vũng hoặc tới khi U rỗng:

- Sử dụng L huấn luyện bộ phõn lớp h_1 dựa trờn khung nhỡn x_1 của x.
- Sử dụng L huấn luyện bộ phõn lớp h_2 dựa trờn khung nhỡn x_1 của x.
- Sử dụng h_1 gỏn nhón tập U.
- Sử dụng h_2 gỏn nhón tập U.

1.1.1.1.1.12 Hình 11: Sơ đồ thiết lập co-training gốc cho bài toán hai lớp

Lúc đầu hai bộ học cùng sử dụng tập dữ liệu đã được gán nhãn L (hạn chế với số lượng nhỏ) để xây dựng hai bộ học h_1 và h_2 dựa trên hai khung nhìn dữ liệu X_1 và X_2. Sau đó hai bộ học này được dùng để gán nhãn các dữ liệu chưa gán nhãn, kết quả "tốt" nhất được bổ sung vào L để dạy lại cho hai bộ học đó. Quá trình lặp tiếp tục và dừng khi đạt tới một số hữu hạn bước nào đó hoặc khi không còn dữ liệu nào chưa gán nhãn, U rỗng.

Sau này khi phát hiện ra rằng, với U lớn thuật toán sẽ rất tốn thời gian chạy, nên trong thực nghiệm phân lớp trang web, Blum và Mitchell đã cải tiến co-training bằng cách sử dụng U' thay cho U, với U' là tập con lấy ngẫu nhiên đại diện trong U. Hình 12 dưới đây biểu diễn thiết lập co-training cải tiến cho bài toán hai lớp.

Cho trước: + L là tập cỏc mẫu huấn luyện đó gỏn nhón.

+ U là tập cỏc mẫu chưa gỏn nhón.

Tạo một tập U' gồm u mẫu được chọn ngẫu nhiờn từ U

Lặp k vũng

- Sử dụng L huấn luyện bộ phõn lớp h_1 dựa trờn khung nhỡn x_1 của x.
- Sử dụng L huấn luyện bộ phõn lớp h_2 dựa trờn khung nhỡn x_2 của x.
- Dựng h_1 gỏn nhón tập U'.
- Dựng h_2 gỏn nhón tập U'.
- Thờm cỏc mẫu mới được gỏn nhón cú độ tin cậy cao vào tập L.
- Chọn ngẫu nhiờn u mẫu từ tập U bổ sung vào tập U'.

Đầu ra: Cỏc mẫu mới được gỏn nhón và hai bộ phõn lớp h_1, h_2

bắt buộc hai bộ phân lớp lựa chọn các mẫu có tính đại diện hơn trong tập U,

ngoài ra vì *U'* chỉ là tập con của *U* nên thuật toán sẽ giảm bớt được thời gian chạy.

2.3.3. Sự hiệu quả và tính ứng dụng của co-training

Co-training đã được nghiên cứu và ứng dụng trong nhiều lĩnh vực, các kết quả thử nghiệm đạt được cho thấy đây là phương pháp có thể khai thác trực tiếp sự phân tách độc lập tự nhiên của các đặc trưng và cho kết quả tốt hơn so với các thuật toán không khai thác đặc điểm này. Vậy, còn khi dữ liệu không có sự phân tách tự nhiên thì thế nào?

Để trả lời câu hỏi này K. Nigam và R. Ghani [25] đã thử nghiệm sử dụng co-training trong bài toán phân lớp các trang web với mục đích là phân loại ra các trang chủ của các khóa học đào tạo (trên thực tế thì số lượng các trang này chiếm khoảng 22%). Để đánh giá hiệu quả của phương pháp học sử dụng co-training, kết quả được các tác giả so sánh với kết quả của phương pháp học giám sát Naïve Bayes và phương pháp bán giám sát sử dụng cực đại hóa kì vọng (Expectaion Maximization-EM). Lúc đầu là kiểm thử trên dữ liệu thỏa mãn có sự phân tách độc lập trên hai khung nhìn dữ liệu, sau đó hai ông kiểm thử trên dữ liệu bán nhân tạo và cuối cùng là thử trên dữ liệu mà tri thức về sự phân tách tự nhiên là không có.

Với trường hợp đầu khi dữ liệu là lý tưởng thì hai tác giả kết luận chắc chắn co-training cho kết quả tốt hơn phương pháp bán giám sát EM, điều này chứng tỏ hiệu suất quan sát của co-training tốt hơn của EM, hay EM bị giảm hiệu suất từ việc bị bẫy trong các cực trị địa phương.

Với trường hợp thứ ba, khi chưa có thông tin gì về sự phân tách tự nhiên trên tập các đặc trưng của dữ liệu thì hai tác giả đề suất một ý tưởng phân tách đặc trưng sao cho thông tin chung có điều kiện giữa các tập đặc trưng là bằng không. Giả sử, với dữ liệu là văn bản ta có thể dựa trên thông tin chung liên quan thu được thông qua việc phân tích thông tin chung có điều kiện giữa các cặp từ và từ đó tính tổng của từng cặp thông tin chung của những tập khác nhau. Quy trình này có thể tóm tắt gồm các bước sau:

+ Tính thông tin chung có điều kiện giữa mỗi cặp từ trong bộ từ vựng.

+ Xây dựng một đồ thị trọng số vô hướng với các đỉnh là các từ, trọng số của các cạnh là thông tin chung giữa các đỉnh tính được trong bước 1.

+ Tách cân bằng hai tập đỉnh trên đồ thị sao cho tổng các trọng số của các cạnh là nhỏ nhất.

Hai tập đỉnh thu được tạo thành hai tập đặc trưng độc lập mà co-training có thể sử dụng được. Tuy bước ba của quy trình trên là bài toán NP-khó nhưng vẫn có thể thực hiện được nhờ các thuật toán xấp xỉ.

Để hiểu rõ hơn về co-training, sau đây ta thực hiện so sánh tổng quát trên hai thuật toán này co-training và self-training.

2.4. So sánh hai phương pháp đồng huấn luyện và tự huấn luyện

Tuy self-training là một phương pháp đơn giản, dễ dùng và khai thác tốt thông tin từ nguồn dữ liệu chưa gán nhãn, song nếu ở bước đánh giá độ tin cậy của các dữ liệu tại mỗi vòng lặp mà không chính xác thì theo Cozman [20] việc thêm các dữ liệu mới này vào tập dữ liệu huấn luyện ban đầu dần làm cho việc học mất tính tương thích và bộ học trở nên tồi đi. Ngoài ra, vì chỉ có một bộ học nên nó lại phải "dựa" và "tin tưởng" vào chính nó nên ý tưởng về đồng huấn luyện hai bộ học và dùng kết quả của bộ học này để "dạy" bộ học kia sẽ giúp tăng chất lượng huấn luyện, phương pháp đồng huấn luyện này được biết tới với tên co-training.

Co-training và self-training là hai phương pháp học bán giám sát có nhiệm vụ chính là mở rộng dần tập các dữ liệu gán nhãn dựa vào tập huấn luyện (đã gán nhãn) ban đầu và khai thác, sử dụng thông tin bổ trợ từ các mẫu được gán nhãn trung gian có độ tin cậy cao . Để có cái nhìn tổng quan về hai phương pháp học này bảng 1 dưới đưa ra sự so sánh giữa hai thiết lập self-training và co-training. Sự khác nhau cơ bản giữa thuật toán self-training và co-training là ở chỗ: self-training chỉ sử dụng một khung nhìn dữ liệu, trong khi đó co-training sử dụng hai khung nhìn dữ liệu. Self-training không yêu cầu sự phân chia của các đặc trưng thành hai khung nhìn độc lập như co-training. Nó chỉ cần một bộ phân lớp với một khung nhìn duy nhất của dữ liệu.

Với điều kiện lý tưởng về sự độc lập trên các khung nhìn thì rõ ràng co-training là khó hơn self-training trong việc ứng dụng vào thực tế. Tuy vậy, với những bài toán cụ thể phù hợp thì co-training vẫn sẽ cho kết quả tốt.

Bảng 1 dưới đây cho ta một cái nhìn tổng quát về sự khác nhau và giống nhau giữa hai phương pháp này.

Tiêu chí	Self-training	*1.1.1.1.1.15.1.1 Co-training*
1.1.1.1.1.15.2 Khung nhìn	1 khung nhìn	2 khung nhìn độc lập
1.1.1.1.1.15.3 Tình huống sử dụng	Khi bộ phân lớp cũ là khó chỉnh sửa	Thoả mãn thiết lập co-training
1.1.1.1.1.15.4 Ưu điểm	Tận dụng nguồn dữ liệu chưa gán nhãn rất phong phú	
	Học tốt trong trường hợp các *features* không thể phân chia thành các khung nhìn độc lập	Cho kết quả tốt nếu các giả thiết được thoả mãn Vì học trên 2 khung nhìn dữ liệu nên chúng sẽ cung cấp nhiều thông tin hữu ích cho nhau hơn.
Nhược điểm	- Khó khăn trong lựa chọn ngưỡng tin cậy của dự đoán (để làm giảm nhiễu trong dự đoán). - Có thể có trường hợp có mẫu không được gán nhãn ➔ cần xác định số lần lặp để tránh lặp vô hạn.	
Khó khăn		Giả thiết độc lập điều kiện thường không đúng trong thực tế. Nên phải xét kĩ bài toán trước khi dùng.

Bảng 1. Bảng so sánh hai thiết lập self-training và co-training

Rõ ràng, hiệu quả của cả hai phương pháp bán giám sát này là phụ thuộc vào chất lượng của các mẫu gán nhãn được thêm vào ở mỗi vòng lặp, và được đo bởi hai tiêu chí:

- Độ chính xác của việc gán nhãn cho các mẫu được thêm vào đó.
- Thông tin hữu ích mà các dữ liệu mang lại cho bộ phân lớp.

Xem xét tiêu chí thứ nhất ta thấy, bộ phân lớp chứa càng nhiều thông tin thì độ tin cậy cho các dự đoán càng cao. Phương pháp co-training sử dụng hai khung nhìn khác nhau của một mẫu dữ liệu với giả thiết là mỗi khung nhìn là đủ *(sufficient)* để dự đoán nhãn cho các mẫu dữ liệu mới. Nếu điều kiện lý tưởng này được thỏa mãn thì co-training là phương pháp cho hiệu quả phân lớp cao. Tuy nhiên, trong thực tế thường thì khó để có điều kiện lý tưởng này, bởi tất cả các đặc trưng đôi khi còn chưa đủ để phân lớp đúng chứ chưa xét tới việc tách thành 2 tập độc lập riêng rẽ. Chính vì lý do đó mà co-training sẽ thực sự hiệu quả với các bài toán thỏa mãn điều kiện này.

Với tiêu chí thứ hai, ta biết rằng thông tin mà mỗi mẫu dữ liệu gán nhãn mới đem lại thường là các *features* mới. Vì thuật toán co-training huấn luyện

trên hai khung nhìn khác nhau nên nó sẽ hữu ích hơn trong việc cung cấp các thông tin mới cho nhau.

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1]. Nguyễn Việt Cường (2005), *Bài toán lọc và phân lớp nội dung Web tiếng Việt theo hướng tiếp cận entropy cực đại*. Khóa luận tốt nghiệp đại học, Đại học Công nghệ - Đại học Quốc gia Hà Nội.

[2]. Hoàng Tiến Dũng (2006), *Mạng nơron RBF và ứng dụng*, Luận văn thạc sĩ, Đại học Công nghệ - ĐH Quốc Gia Hà nội.

[3]. Đặng Thanh Hải (2004),*Thuật toán phân lớp văn bản Web và thực nghiệm trong máy tìm kiếm VietSeek*. Khóa luận văn tốt nghiệp đại học, Đại học Công nghệ - Đại học Quốc gia Hà Nội.

[4]. Hoàng Xuân Huấn (2009), *Bài giảng nhận dạng mẫu*.

[5]. Hoàng Xuân Huấn và *Đặng* Thị Thu Hiền (2005), *Phương pháp lặp huấn luyện mạng nội* suy *RBF*, kỷ yếu hội thảo quốc gia các vấn đề chọn lọc của CNTT lần thứ VIII, Hải phòng, pp. 314-323.

[6]. Hoàng Xuân Huấn (2004), *Giáo trình các phương pháp số*, NXB Đại học quốc gia Hà Nội.

[7]. Đặng Thị Thu Hiền và Hoàng Xuân Huấn (2008), *Thuật toán một pha huấn luyện* nhanh *mạng nội suy RBF với mốc cách đều*, kỷ yếu Hội thảo quốc gia các vấn đề chọn lọc của CNTT lần thứ X, Đại Lải 9/2007, pp. 532-542.

[8]. Đặng Thị Thu Hiền (2009), *Bài toán nội suy và mạng nơron RBF,* Luận án tiến sĩ Công nghệ thông tin (ban thảo).

[9]. Lê Tiến Mười (2009), *Mạng neural RBF và ứng dụng nhận dạng chữ viết tay*, Khoá luận tốt nghiệp Đại học, ĐH Công nghệ - ĐH Quốc Gia Hà nội.

Tiếng Anh

[10]. A. McCallum, K. Nigam (1998), *A Comparison of Event Model for Naive Bayes Text Classification*, Working Notes of the 1998 AAAI/ICML Workshop on Learning for Text Categorization.

[11]. A. Blum and T. Mitchell (1998), *Combining labeled and unlabeled data with co-training*. In Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT-98).

[12]. A. P. Dempster, N. M. Laird, and D. B. Rubin (1977), *Maximum likelihood from incomplete data via the EM algorithm*. Journal of the Royal Statistical Society, Series B, 39(1):138.

[13]. C.G. Looney (1997). *Pattern recognition using neural networks: Theory and algorithm for engineers and scientist*, Oxford University press, New York.

[14]. D. Yarrowsky (1995), *Unsupervised Word Sense Disambiguation Rivaling Supervised Methods*, In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics, 189-196.

[15]. D.S. Broomhead, D. Lowe (1988). *Multivariable functional interpolation and adaptive networks*. Complex Systems, vol. 2, 321-355.

[16]. E. Riloff and R. Jones (1999), *Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping*. In Proceedings of the 16th National Conference on Artificial Intelligence.

[17]. E. Blanzieri (2003), *Theoretical Interpretations and Applications of Radial Basis Function Networks*, Technical Report DIT-03-023, Informatica Telecomunicazioni, University of Trento.

[18]. E. Rillof, J. Wiebe, T. Wilson (2003), *Learning Subjective Nouns using Extraction Pattern Bootstrapping*. 2003 Conference on Natural Language Learning (CoNLL-03), ACL SIGNLL.

[19]. E.J. Hartman, J.D. Keeler and J.M. Kowalski (1990). *Layered neural networks with Gaussian hidden units as universal approximations*, Neural Comput., vol. 2,no. 2, 210-215.

[20]. F. G. Cozman, and I. Cohen (2002), *Unlabeled data can degrade classification performance of generative classifiers*, Int'l Florida Artificial Intell. Society Conf., 327-331.

[21]. F. Cozman, I. Cohen, & M. Cirelo.(2003), *Semi-supervised learning of mixture models*. ICML-03, 20th International Conference on Machine Learning.

[22]. F. Roli (2005), *Semi-Supervised Multiple Classifier Systems: Background and Research Directions*, Multiple Classifier Systems, Springer Verlag , vol 3541.

[23]. F. Schwenker. H.A. Kesler, Günther Palm (2001), *Three learning phases for radial-basis-function networks*, Neural networks, Vol.14, 439-458.

[24]. Hoang Xuan Huan, Dang Thi Thu Hien and Huu Tue Huynh (2007), *A Novel Efficient Algorithm for Training Interpolation Radial Basis Function Networks*, Signal Processing 87, 2708 - 2717.

[25]. K. Nigam, R. Ghani (2000), *Analyzing the effectiveness and applicability of cotraining*. In Proceedings of Ninth International Conference on Information and Knowledge Management (CIKM-2000), 86–93.

[26]. K. Nigam, R. Ghani (2000), *Understanding the behavior of co-training*. In Proceedings of KDD-2000 Workshop on Text Mining.

[27]. K. Nigam, A. McCallum, S. Thrun, T. Mitchell (2000). *Text Classification from Labeled and Unlabeled Documents using EM*. Machine Learning, 39(2/3):103-134.

[28]. Le C. A., Huynh V. N., and A. Shimazu(2005), *Combining Classifiers with Multi-Representation of Context in Word Sense Disambiguation*. In Proc. PAKDD, 262–268.

[29]. M. Collins and Y. Singer, *Unsupervised Model for Name Entity Recognition*, In EMNLP.

[30]. M. Thelen and E. Riloff (2002), *A bootstrapping method for Learning Semantic Lexicons using Extraction Pattern Contexts*. 2002 Conf. on Empirical Methods in Natural Language Processing, Philadelphia, PA, July 2002, 214-221.

[31]. M. Solyman, N.El Gayar (2006), *A Co-Training Approach for Semi-Supervised Multiple Classifiers.* Infos.

[32]. M. Seeger (2002), *Learning with labeled and unlabeled data*. Technical Report, University of Edinburgh, Institute for Adaptive and Neural Computation, Dec. 2002, pp. 1-62.

[33]. M.J.D.Powell (1998). *Radial basis function approximations to polynomials*. Numerical analysis 1987 Proceeding, 223-241, Dundee, UK.

[34]. N. El Gayar (2004), *A Multi-classifier Approach to Selfsupervised Learning*, Proc. 1st International Computer Engineering Conference, Cairo, pp. 197-201

[35]. N. El Gayar (2004). *An Experimental Study of a Self-Supervised Classifier Ensemble*, International Journal of Information Technology, Vol. 1, No. 1, ISSN:1305-239X.

[36]. O. Chapelle, A. Zien, & B. Sch¨olkopf (Eds.) (2006), *Semi supervised learning*. MIT Press.

[37]. R. Jones, A. McCallum, K. Nigam, E. Rillof (1999), *Bootstrapping for text learning Tasks*, IJCAI-99 Workshop on Text Mining: Foundations, Techniques and Applications.

[38]. R.O. Duda; P.E. Hart and D.G. Stork (2001) , *Pattern clasification* , JoHn Wiley & Sons (2 nd edition).

[39]. S.Eyheramendy, D. David Lewis, David Madigan (2003), *On the Naive Bayes Model for Text Classification*, to appear in Artificial Intelligence & Statistics.

[40]. S. Vanderlooy (2005), *Co-Training of Version Space Support Vector Machines*. A thesis submitted in partial fulfillment of the requyrements for the degree of Master of Science.

[41]. S. Haykin (1999): *A Comprehensive Foundation*. Prentice-Hall Inc, second edition.

[42]. T. M. Mitchell (1997), *Machine learning*, McGraw-Hill.

[43]. T. Joachims (2003), *Transductive learning via spectral graph partitioning*. In Proceeding of. The Twentieth International Conference on Machine Learning (ICML2003), 290-297.

[44]. T. Joachims (1999), T*ransductive Inference for Text Classification using Support Vector Machines*. International Conference on Machine Learning (ICML).

[45]. W. Wu, D. Chen and J. Yang (2005), *Integrating Co-training and Recognition for text detection*, IEEE International Conference on Multimedia & Expo, Amsterdam, The Netherlands, July 6-8.

[46]. X. Zhu (2006), *Semi-Supervised Learning Literature Survey*. Computer Sciences TR 1530, University of Wisconsin – Madison, February 22.

[47]. X. Zhu , A. B. Goldberg (2009) , *Introduction to Semi-Supervised Learning*, Synthesis Lectures on Artificial Intelligence and Machine Learning, Series ISSN.

[48]. Z. H. Zhou, M. Li (2007), *Semi- Supervised Regression with Co-training Style Algorithms*, IEEE transactions on knowledge and data engineering, Vol .X, No .XX, Month.

[49]. http://en.wikipedia.org/wiki/

http://www.scholarpedia.org/article