

**ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

Phạm Quốc Thắng

**NÂNG CAO HIỆU NĂNG CÁC PHƯƠNG PHÁP
PHÂN LOẠI ĐỐI TƯỢNG TRONG ẢNH**

Chuyên ngành: Khoa học máy tính

Mã số: 9480101.01

TÓM TẮT LUẬN ÁN TIẾN SĨ CÔNG NGHỆ THÔNG TIN

Hà Nội, 2020

Công trình được hoàn thành tại:

Trường Đại học Công nghệ, Đại học Quốc gia Hà Nội

Người hướng dẫn khoa học: GS.TS Nguyễn Thanh Thủy
PGS.TS Nguyễn Đức Dũng

Phản biện:.....

.....

Phản biện:.....

.....

Phản biện:.....

.....

Luận án sẽ được bảo vệ trước Hội đồng cấp Đại học Quốc gia chấm luận án tiến sĩ họp tại

vào hồi giờ ngày tháng năm

Có thể tìm hiểu luận án tại:

- Thư viện Quốc gia Việt Nam

- Trung tâm Thông tin - Thư viện, Đại học Quốc gia Hà Nội

Mở đầu

Tính cấp thiết của đề tài nghiên cứu:

Phân loại (phân lớp, classification) là một tiến trình xử lý nhằm xếp các dữ liệu hay các đối tượng nào đó vào một trong các lớp đã được định nghĩa trước.

Nhìn chung, trong các bài toán phân loại đối tượng như nhận dạng ngôn ngữ ký hiệu, cử chỉ, hành động, biển báo giao thông... yêu cầu đặt ra thường là đạt được kết quả nhận dạng chính xác cao, thời gian thực hiện nhanh thì hệ thống ứng dụng mới có ý nghĩa thực tiễn.

Mô hình hàm cơ sở (radial basis function models - RBF) là một trong những mô hình được sử dụng rộng rãi trong nhiều ứng dụng khác nhau như phân loại, xấp xỉ hàm, dự đoán dữ liệu theo thời gian. Trong thực tế, các mô hình RBF đã đạt được hiệu năng dự đoán tiên tiến nhất trong nhiều ứng dụng như nhận dạng hình ảnh, nhận dạng chữ số viết tay và nhận dạng cử chỉ của con người.

Một mục tiêu chung của những phương pháp được đề xuất trong thời gian gần đây là tìm kiếm các mô hình RBF có cấu trúc đơn giản với số lượng ít các hàm cơ sở (mô hình RBF thưa). Tốc độ tính toán của mô hình RBF thưa liên quan trực tiếp đến số lượng các hàm cơ sở. Với càng ít hàm cơ sở, mô hình RBF thưa cho kết quả càng nhanh và hệ thống có thể xử lý một lượng thông tin nhiều hơn trong cùng một đơn vị thời gian. Sự đơn giản của các mô hình RBF thưa mang ý nghĩa thực tế quan trọng trong các ứng dụng với yêu cầu xử lý thông tin cực nhanh như xử lý tín hiệu video, hình ảnh, an toàn và bảo mật thông tin.

Bên cạnh đó, sự cần thiết tiến hành nghiên cứu, đánh giá sự phù hợp của các mô hình RBF thưa đối với các loại dữ liệu khác nhau trong các lĩnh vực khác nhau của thực tế. Với sự bùng nổ của thông tin, các ứng dụng và dữ liệu ngày càng trở nên đa dạng và phức tạp hơn. Các sự vật hay hiện tượng có thể được liên kết và mô tả bởi các loại dữ liệu khác nhau như số, phi số. Các véc-tơ biểu diễn có thể là liên tục hay rời rạc. Mối liên hệ giữa các đối tượng là đơn chiều hay đa chiều... Những kết quả của nghiên cứu đó sẽ đóng góp trực tiếp vào các lĩnh vực ứng dụng thời sự hiện nay như xử lý dữ liệu đa phương tiện, phân tích mạng xã hội, an toàn bảo mật thông tin.

Một trong những tồn tại chung trong huấn luyện mô hình RBF thưa là yêu cầu tính toán phức tạp của tất cả các phương pháp nêu trên. Các ứng dụng xử lý dữ liệu đa phương tiện có yêu cầu xử lý lượng thông tin cực lớn với độ chính xác ngày càng cao. Để đạt được các yêu cầu trên, các mô hình học máy thường

được huấn luyện với lượng dữ liệu khổng lồ vượt qua năng lực tính toán và bộ nhớ thông thường. Điều này đòi hỏi có những phương pháp mới với khả năng làm việc tốt trên các tập dữ liệu rất lớn. Đây cũng chính là một yêu cầu cả về lý thuyết và công nghệ cần nghiên cứu, giải quyết.

Mục tiêu nghiên cứu:

Luận án tiến hành nghiên cứu và phát triển các phương pháp xây dựng mô hình RBF thưa với số lượng tối thiểu các hàm cơ sở trên các tập dữ liệu lớn theo hai hướng tiếp cận khác nhau là hàm quyết định với lề cực đại (maximum-margin) và mô hình xác suất Bayesian (sparse Bayesian learning). Với hướng tiếp cận thứ nhất, luận án tập trung vào các phương pháp rút gọn số lượng các hàm cơ sở trong khi vẫn đảm bảo độ chính xác của các thuật toán huấn luyện. Với hướng tiếp cận thứ hai, luận án phát triển các phương pháp hiệu quả để đảm bảo các thuật toán đã có có khả năng làm việc tốt hơn trên các tập dữ liệu huấn luyện lớn.

Mục tiêu chính của luận án bao gồm:

- Nghiên cứu tổng quan, đánh giá so sánh các phương pháp huấn luyện mô hình RBF thưa với số lượng hàm cơ sở được rút gọn. Nghiên cứu này sẽ mang lại những nhận định, phân tích, gợi ý về việc sử dụng các phương pháp huấn luyện khác nhau đối với các bài toán thực tế khác nhau.
- Phát triển các phương pháp rút gọn số lượng cần thiết các hàm cơ sở nhằm thỏa mãn hai tiêu chí về độ chính xác và tốc độ thực hiện.
- Phát triển các thuật toán hiệu quả giúp các cách tiếp cận và phương pháp đã nêu có khả năng thực hiện tốt hơn trên các tập dữ liệu lớn hơn.

Đối tượng và phạm vi nghiên cứu:

Để đạt được những mục tiêu trên, luận án thực hiện những nội dung nghiên cứu cụ thể sau:

- Nghiên cứu khảo sát các phương pháp huấn luyện xấp xỉ mô hình RBF thưa với cách tiếp cận hàm quyết định với lề cực đại (RBF kernel support vector machines).
- Nghiên cứu khảo sát các phương pháp huấn luyện mô hình RBF thưa trên cơ chế suy luận xác suất thông kê Bayes (sparse Bayesian learning).
- Nghiên cứu, phân tích, đánh giá hiệu quả của các phương pháp huấn luyện mô hình RBF thưa trong các ứng dụng phân loại đối tượng.
- Phát triển các phương pháp rút gọn số lượng các hàm cơ sở trong các mô hình RBF thưa được huấn luyện bởi các phương pháp và thuật toán khác nhau.
- Phát triển các thuật toán mới nhằm nâng cao hiệu suất của các phương pháp đã khảo sát, tăng cường khả năng làm việc của các thuật toán này đối với các tập dữ liệu lớn.

Phương pháp nghiên cứu:

Về nghiên cứu lý thuyết

- Cơ sở lý thuyết của các cách tiếp cận hàm quyết định với lẽ cực đại như SVM, đặc biệt chú trọng đến ảnh hưởng của việc rút gọn số lượng các véc-tơ hỗ trợ có trong hàm quyết định.

- Cơ sở lý thuyết của các phương pháp xây dựng mô hình RBF thưa với cách tiếp cận dựa trên lý thuyết Bayes. Phân tích sự khác biệt của cách tiếp cận này đối với các loại dữ liệu khác nhau có thể có trong các ứng dụng thực tế.

- Nghiên cứu các phương pháp hiệu quả trong giải quyết các bài toán tối ưu có trong các vấn đề nghiên cứu trên, trong đó các bài toán tối ưu được thực hiện trên các dữ liệu véc-tơ nhiều chiều và ma trận kích thước lớn. Hai hướng nghiên cứu quan trọng cần được quan tâm đặc biệt là việc sử dụng hiệu quả không gian bộ nhớ và các thuật toán tìm kiếm nhanh trong không gian nhiều chiều.

Về nghiên cứu thực nghiệm

- Cài đặt các thuật toán huấn luyện mô hình RBF thưa bằng các công cụ lập trình C/C++, Matlab.

- Thử nghiệm, đánh giá, phân tích các phương pháp được cài đặt trên các tập dữ liệu chuẩn (benchmark data sets) khác nhau.

- Cải tiến việc cài đặt để nâng cao tốc độ tính toán và hiệu quả sử dụng bộ nhớ của các thuật toán khác nhau.

Các đóng góp của luận án:

- Kết quả phân tích, so sánh và đánh giá các phương pháp huấn luyện mô hình RBF thưa đối với phân loại đối tượng và yêu cầu tính toán khác nhau. Những kết luận về sự phù hợp của các phương pháp này trong những điều kiện hoàn cảnh cụ thể.
- Đề xuất và phát triển các thuật toán stochastic SimpSVM hiệu quả đẩy nhanh việc huấn luyện, xây dựng mô hình SimpSVM giảm lược nghiệm SVM với ít hơn đáng kể các SVs, từ đó tăng tốc pha test của SVM trong khi giữ được độ chính xác phân lớp không thay đổi nhiều.
- Đề xuất và phát triển các thuật toán FastRVM hiệu quả đẩy nhanh việc huấn luyện RVM lên nhiều lần trong khi đảm bảo được độ chính xác phân loại cuối. Đồng thời, tăng cường khả năng làm việc của phương pháp RVM hiện tại trên các tập dữ liệu lớn. Những phương pháp được đề xuất sẽ đóng góp và sự phát triển chung của cộng đồng nghiên cứu về mô hình RBF nói riêng và trong lĩnh vực học máy hay trí tuệ nhân tạo nói chung.
- Góp phần nghiên cứu ứng dụng các mô hình RBF thưa trong nhận dạng đối tượng, cụ thể là phân loại đối tượng trong các bài toán thực tế như nhận dạng ngôn ngữ ký hiệu, nhận dạng cử chỉ, hành động...

Bố cục của luận án:

Ngoài phần mở đầu và phần kết luận, Luận án được chia thành 3 chương.

Chương 1

Mô hình RBF thưa trong bài toán phân lớp

1.1 Giới thiệu mô hình RBF

Mô hình RBF được sử dụng rộng rãi trong nhiều ứng dụng khác nhau như phân loại, xấp xỉ hàm, dự đoán dữ liệu theo thời gian. Với cấu trúc đơn giản dưới dạng tổ hợp tuyến tính của các hàm cơ sở

$$f(x) = \sum_{i=1}^M w_i \phi_i(x) + b \quad (1.1)$$

các hàm quyết định (1.1) có nhiều lợi thế: khả năng tổng quát hóa mạnh và cấu trúc của mô hình nhỏ gọn. $\phi_i(x) = \exp(-\gamma \|x - c_i\|^2)$ là hàm cơ sở có tâm tại c_i và độ rộng hàm γ .

1.2 SVM: mô hình RBF phân lớp dựa trên lề cực đại

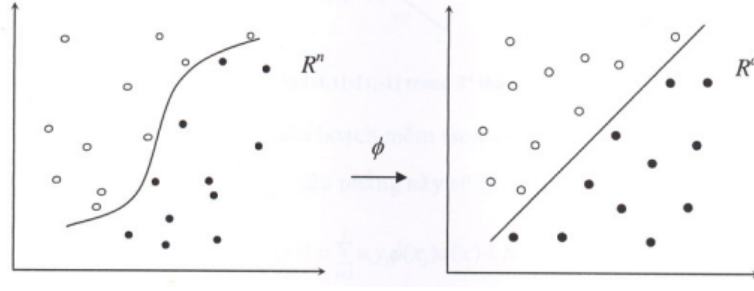
Ý tưởng chính của SVM là chuyển tập mẫu từ không gian biểu diễn R^n của chúng sang một không gian R^d có số chiều lớn hơn và tìm một siêu phẳng tối ưu để phân hoạch tập mẫu này dựa trên phân lớp của chúng.

1.2.1 SVM tuyến tính

Tập mẫu là một tập có thể phân chia tuyến tính được bằng một siêu phẳng và SVM đi tìm siêu phẳng này.

1.2.2 SVM phi tuyến: sử dụng hàm nhân RBF

Tìm một hàm nhân $k(x, y)$, sau đó giải bài toán siêu phẳng phân hoạch mềm với việc thay $(x_1, x_2) = k(x_1, x_2)$.



Hình 1.1: Một mặt phân chia phi tuyến có thể trở thành một siêu phẳng trong không gian lớn hơn

1.3 RVM: mô hình RBF phân lớp dựa trên phép suy luận Bayes

Một tập các ứng viên phổ biến cho $y(x)$ có dạng:

$$y(x; w) = \sum_{i=1}^M w_i \phi_i(x) = w^T \phi(x) \quad (1.2)$$

1.3.1 Mô hình Bayesian thưa

Công thức xác suất chuẩn và giả định rằng các mục tiêu là các mẫu từ mô hình với nhiễu bổ sung:

$$t_n = y(x_n; w) + \epsilon_n \quad (1.3)$$

Do giả thiết về tính độc lập của t_n , likelihood của tập dữ liệu hoàn chỉnh có thể được viết như sau:

$$p(t|w, \sigma^2) = (2\pi\sigma^2)^{-N/2} \exp \left\{ -\frac{1}{2\sigma^2} \|t - \Phi w\|^2 \right\} \quad (1.4)$$

Suy luận Bayesian tiến hành bằng cách tính toán, từ quy tắc Bayes, hậu nghiệm cho trước dữ liệu:

$$p(w, \alpha, \sigma^2 | t) = \frac{p(t|w, \alpha, \sigma^2) p(w, \alpha, \sigma^2)}{p(t)} \quad (1.5)$$

Sau đó, khi cho một điểm test mới, x_* , các dự đoán được thực hiện cho đầu ra tương ứng t_* , theo phân phối dự đoán:

$$p(t_* | t) = \int p(t_* | w, \alpha, \sigma^2) p(w, \alpha, \sigma^2 | t) dw d\alpha d\sigma^2 \quad (1.6)$$

Dựa trên phân phối hậu nghiệm trên các trọng số, điều kiện trên các giá trị cực đại α_{MP} và σ_{MP}^2 . Sau đó có thể tính phân phối dự đoán, từ (1.6) cho một dữ liệu mới x_* .

$$p(t_* | t, \alpha_{MP}, \sigma_{MP}^2) = \int p(t_* | w, \sigma_{MP}^2) p(w | t, \alpha_{MP}, \sigma_{MP}^2) dw \quad (1.7)$$

Vì cả hai số hạng trong tích phân là Gaussian, dễ dàng tính toán, cho:

$$p(t_*|t, \alpha_{MP}, \sigma_{MP}^2) = N(t_*|y_*, \sigma_*^2) \quad (1.8)$$

1.3.2 Phân lớp với mô hình Bayesian thưa

Áp dụng hàm liên kết sigmoid logistic $\sigma(y) = 1/(1 + e^{-y})$ cho $y(x)$ và thông qua phân phối Bernoulli cho $P(t|x)$, viết likelihood như sau:

$$P(t|w) = \prod_{n=1}^N \sigma\{y(x_n; w)\}^{t_n} [1 - \sigma\{y(x_n; w)\}]^{1-t_n} \quad (1.9)$$

trong đó, theo mô tả xác suất, các đầu ra $t_n \in \{0, 1\}$.

1.4 Độ phức tạp tính toán của các phương pháp

1.4.1 Độ phức tạp tính toán của SVM

Khi test mẫu mới x , thủ tục tốn kém nhất là SVM so sánh nó với toàn bộ các SV thông qua các tính toán nhân K . Tính toán này tỉ lệ tuyến tính với số SV N_S . Trong nhiều trường hợp, số lượng lớn các SV này là nguyên nhân chính làm cho hàm quyết định (1.1) trở nên chậm chạp hơn.

Như vậy, giảm so sánh này sẽ làm giảm sự chậm chạp, cũng có nghĩa là làm tăng tốc độ pha test của SVM. Để tăng tốc SVM, có thể tiếp cận rút ngắn thời gian huấn luyện hoặc đẩy nhanh pha test. Theo quan điểm của chúng tôi, đẩy nhanh pha test sẽ có ý nghĩa hơn trong thực tiễn vì sẽ tạo ra được một máy dự đoán nhanh.

1.4.2 Độ phức tạp tính toán của RVM

Phép toán nghịch đảo ma trận trong tính Σ yêu cầu các phép toán $O(N^3)$. Việc tính toán hạng mục lỗi yêu cầu các phép toán $O(N^2)$. Các ma trận Φ và Σ có hạng đầy đủ, do đó yêu cầu độ phức tạp không gian ban đầu là $O(N^2)$. Những vấn đề này làm thời gian huấn luyện của RVM chậm, hạn chế tính thực tiễn của thuật toán RVM cơ bản đối với các bài toán có kích thước vừa và lớn.

1.5 Học sâu và mô hình RBF lai

Gần đây, học sâu (DL) đã trở thành xu hướng phát triển nhanh trong phân tích dữ liệu lớn và đã được áp dụng rộng rãi và thành công trong nhiều lĩnh vực.

1.5.1 Mô hình học sâu trong phân lớp

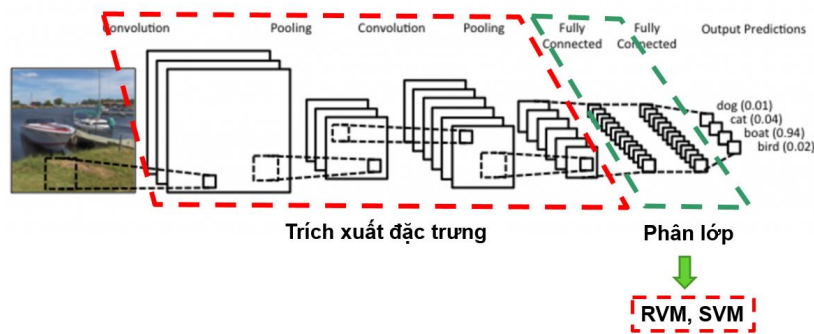
CNN có thể trích xuất đặc trưng cho thấy hiệu năng phân loại tốt. Các đặc trưng học được của CNN có thể sử dụng để huấn luyện thay cho các đặc trưng “nông” được thiết kế thủ công truyền thống.

1.5.2 Mô hình lai CNN-SVM

Kiến trúc của mô hình lai CNN-SVM được thiết kế bằng cách thay thế layer đầu ra cuối cùng của CNN bằng một bộ phân loại SVM. Trong mô hình này, CNN hoạt động như một bộ trích xuất đặc trưng huấn luyện và SVM hoạt động như một bộ nhận dạng.

1.5.3 Mô hình lai đề xuất CNN-RVM

Chúng tôi đề xuất mô hình lai CNN-RVM với kiến trúc tương tự như mô hình lai CNN-SVM (Hình 1.2). CNN hoạt động như một bộ trích xuất đặc trưng để huấn luyện và RVM đóng vai trò như một bộ học nhận dạng.



Hình 1.2: Mô hình lai CNN-RVM phân loại ảnh

Các mô hình lai CNN-SVM, CNN-RVM là mô hình phân loại triển vọng do:

- Tự động trích xuất các đặc trưng giúp tiết kiệm công sức, thời gian.
- Mô hình lai CNN-SVM, CNN-RVM kết hợp các ưu điểm của CNN và SVM, RVM, chúng đều là các mô hình phân loại phổ biến và thành công nhất.

1.6 Kết chương

Chương này của luận án tập trung vào giới thiệu tổng quan về mô hình RBF trong bài toán nhận dạng. Phần đầu tập trung vào giới thiệu mô hình RBF. Các phần sau của chương trình bày về hai mô hình RBF: mô hình RBF dựa trên biên cực đại (SVM) và mô hình RBF dựa trên suy luận xác suất thống kê (RVM). Tiếp theo là sự phân tích về độ phức tạp tính toán của các phương pháp đó. Đây là những cơ sở lý thuyết giúp ích cho định hướng nghiên cứu và xây dựng các thuật toán sẽ được trình bày ở chương tiếp theo.

Chương 2

Các thuật toán huấn luyện nhanh các mô hình RBF thưa

2.1 Các thuật toán huấn luyện mô hình RBF thưa

2.1.1 Các thuật toán tăng tốc SVM

Đối với cả SVM, thủ tục tốn kém nhất trong test một vector đối tượng mới x là so sánh nó với toàn bộ SV thông qua hàm nhân K . Để giảm chi phí tính toán này, hay để tăng tốc pha test, phương pháp tập rút gọn cố gắng thay thế N_S , số SV gốc, bằng N_Z , một số lượng nhỏ hơn các vectơ mới, gọi là tập RV. Hàm quyết định sau đó trở thành ($T = 1$ trong trường hợp hai lớp):

$$f'_t(x) = \sum_{i=1}^{N_Z} \beta_{ti} K(z_i, x) + b_t, t = 1, \dots, T \quad (2.1)$$

2.1.2 Các thuật toán tăng tốc RVM

Có hai chiến lược để giảm độ phức tạp RVM được đề xuất trước đây:

- + Thực hiện phép nghịch đảo ma trận gần đúng thay vì chính xác với chi phí tính toán thấp hơn.
- + Lặp lấy mẫu con của dữ liệu, do đó giảm N .

2.2 Thuật toán Stochastic SimpSVM

2.2.1 Thuật toán SimpSVM-GD

Kết hợp SV đa trọng số

Giả sử chúng ta muốn thay thế hai SV đa trọng số (x_i, α_{ti}) và (x_j, α_{tj}) bằng một vectơ mới (z, β_t) , $t = 1, \dots, T$, nghiệm tối ưu 2-norm cho toàn bộ SVM sẽ

là nghiệm mà nó cực tiểu

$$\min \leftarrow \sum_{t=1}^T \|\beta_t \Phi(z) - (\alpha_{t_i} \Phi(x_i) + \alpha_{t_j} \Phi(x_j))\|^2 \quad (2.2)$$

Điều chỉnh toàn bộ các RV

Để giữ cho nghiệm giảm lược giống như nghiệm gốc nhất có thể, chúng ta có thể điều chỉnh tổng thể toàn bộ các RV dựa theo chuẩn của các siêu phẳng của các nghiệm bằng cách cực tiểu độ khác biệt giữa chúng:

$$\min \leftarrow \rho = \sum_{t=1}^T \left\| \sum_{i=1}^{N_S} \alpha_{t_i} \Phi(x_i) - \sum_{i=1}^{N_Z} \beta_{t_i} \Phi(z_i) \right\|^2 \quad (2.3)$$

Áp dụng giảm bậc gradient để cực tiểu ρ đối với toàn bộ RVs $z_i, i = 1, \dots, N_Z$, các hướng tìm kiếm cho nhân Gauss RBF là:

$$\frac{\partial \rho_{RBF}}{\partial z_i} = \sum_{t=1}^T \left(\sum_{j=1}^{N_Z} -2\gamma \beta_{t_i} \beta_{t_j} K(z_i, z_j) (z_i - z_j) - \sum_{j=1}^{N_S} -2\gamma \beta_{t_i} \alpha_{t_j} K(z_i, x_j) (z_i - x_j) \right) \quad (2.4)$$

Ở mỗi vòng lặp, cập nhật toàn bộ các z_i như sau:

$$z_i^{(t+1)} = z_i^{(t)} - \eta \frac{\partial \rho}{\partial z_i^{(t)}}, i = 1, \dots, N_Z \quad (2.5)$$

Tính toán lại các hệ số

Các hệ số tối ưu của RVs sau đó được tính toán lại bằng cách giải các phương trình sau cho tất cả T SVM hai lớp:

$$\beta_t = (K^{zz})^{-1} K^{zx} \alpha_t, t = 1, \dots, T \quad (2.6)$$

Giải thuật giảm lược

Giải thuật giảm lược lặp đi lặp lại việc chọn hai SVs x_i và x_j và thay thế chúng bằng một vectơ mới được xây dựng z .

2.2.2 Thuật toán đề xuất Stochastic SimpSVM-SMD

Ở mỗi lần lặp cập nhật z_i , thay vì phải tính toán gradient thực của F^* bằng cách sử dụng toàn bộ các SVM đơn, chúng tôi ước lượng gradient này trên cơ sở một SVM đơn thứ m được lựa chọn ngẫu nhiên

$$z_i^{(t+1)} = z_i^{(t)} - \eta^{(t)} \frac{\partial F_m}{\partial z_i^{(t)}}, i = 1, \dots, N_z \quad (2.7)$$

Tập dữ liệu	dna		satimage		shuttle		usps	
	#SV	Acc(%)	#SV	Acc(%)	#SV	Acc(%)	#SV	Acc(%)
100%	843	95.62	1215	89.75	4191	99.03	1670	94.77
50%	422	95.62	608	89.75	2096	99.03	835	94.77
10%	84	95.53	122	89.45	419	99.03	167	94.67
5%	42	95.19	61	89.25	210	99.03	84	93.92
1%	8	95.03	12	78.00	42	99.04	45	89.59

Bảng 2.1: Chính xác dự đoán của SimpSVM với tốc độ tăng tốc khác nhau trên các tập dữ liệu

Quá trình cập nhật sẽ được thực hiện cho đến khi hàm mục tiêu hội tụ về giá trị nhỏ nhất.

Cách tiếp cận này có độ phức tạp tính toán tỉ lệ tuyến tính với số lượng SVM (hay số lớp) của bài toán, cho phép phân loại dữ liệu hiệu quả với các lớp lớn.

2.2.3 Thuật toán đề xuất Stochastic SimpSVM-SVD

Ở mỗi lần lặp (2.5), thay vì điều chỉnh toàn bộ các RV, chúng ta điều chỉnh một vectơ z_i được lựa chọn ngẫu nhiên nhưng với toàn bộ các hướng. Quy tắc cập nhật sau đó là:

$$z_i^{(t+1)} = z_i^{(t)} - \eta \frac{\partial \rho}{\partial z_i^{(t)}}, i \in [1, \dots, N_Z] \quad (2.8)$$

trong đó η là kích thước bước lặp. Quá trình cập nhật sẽ được thực hiện cho đến khi hàm mục tiêu hội tụ về giá trị nhỏ nhất. Kết quả nghiên cứu này đã được công bố trong công trình [CT5].

2.2.4 Thực nghiệm

Trước tiên, chúng tôi trình bày lại thực nghiệm chỉ ra hiệu năng của thuật toán SimpSVM so với thuật toán SVM gốc. Thuật toán SimpSVM chạy với các giá trị khác nhau của N_Z , chỉ thị tốc độ tăng tốc trong pha test khác nhau của SimpSVM giảm lược trên 4 tập dữ liệu chuẩn: dna, satimage, shuttle, usps. So sánh độ chính xác của SVM gốc và SimpSVM giảm lược trên dữ liệu test được thể hiện trong Bảng 2.1. Các kết quả cho thấy SimpSVM giảm lược với chỉ 10% số SV có hiệu năng gần tương đương SVM gốc. Đặc biệt, trên tập dữ liệu "dna" và "shuttle", độ tăng tốc có thể lên đến 100 lần mà không bị mất độ chính xác dự đoán.

Dữ liệu thực nghiệm

Chúng tôi đã chọn các tập dữ liệu đa lớp khác nhau có sẵn công khai từ thư viện học máy UCI bao gồm: dna, letter, satimages, shuttle, vowel, pendigits,

usps, mnist.

Đánh giá hiệu năng

Đánh giá hiệu năng của các thuật toán tập trung vào hai phương diện quan trọng của các giải thuật, đó là hiệu năng phân lớp và thời gian huấn luyện.

Hiệu năng phân lớp được đánh giá bằng cách tính toán tỉ lệ mẫu phân lớp đúng trên tổng số mẫu (độ chính xác) sử dụng công thức:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (2.9)$$

Lựa chọn tham số

Chúng tôi sử dụng phương pháp tìm kiếm lưới để để lựa chọn các tham số huấn luyện C, γ sao cho các bộ phân lớp SVM gốc được huấn luyện có độ chính xác dự đoán tốt trên tập dữ liệu test độc lập.

Kết quả thực nghiệm

Trong thực nghiệm thứ nhất, chúng tôi chạy các giải thuật SimpSVM đề xuất với các giá trị N_z khác nhau, chỉ ra tốc độ tăng tốc của các giải thuật SimpSVM đề xuất trong pha test trên các tập dữ liệu. Sau đó chúng tôi so sánh độ chính xác của các giải thuật SimpSVM với các độ tăng tốc N_z đó trên các tập dữ liệu test. Kết quả thực nghiệm trong Bảng 2.2 cho thấy các giải thuật SimpSVM có độ chính xác phân lớp khá tương đồng nhau theo các thiết đặt độ tăng tốc N_z . Trong hầu hết các tập dữ liệu với các N_z khác nhau, SimpSVM-SVD có độ chính xác cao hơn so với SimpSVM-SGD.

Trong thực nghiệm thứ hai, chúng tôi so sánh thời gian huấn luyện xây dựng mô hình giảm lược của các giải thuật SimpSVM trên các tập dữ liệu với N_z bằng một nửa N_S (hay tốc độ trong pha test của giải thuật sẽ nhanh gấp đôi). So sánh này được chỉ ra trong Hình 2.1. Có thể thấy trên 6 tập dữ liệu này, SimpSVM-SGD có thời gian huấn luyện xây dựng mô hình giảm lược nhỏ hơn so với SimpSVM-GD, tuy nhiên nó mất mát một chút độ chính xác dự đoán. SimpSVM-SVD có thể giảm thời gian cho việc giảm lược SVM tốt trong khi hiệu năng dự đoán gần như không thay đổi. Đặc biệt, trên các tập dữ liệu "letter", "shuttle", "vowel" và "mnist", nó có thời gian chỉ bằng một phần năm so với SimpSVM hay nói cách khác nó chạy nhanh hơn 5 lần so với SimpSVM mà không mất mát về độ chính xác.

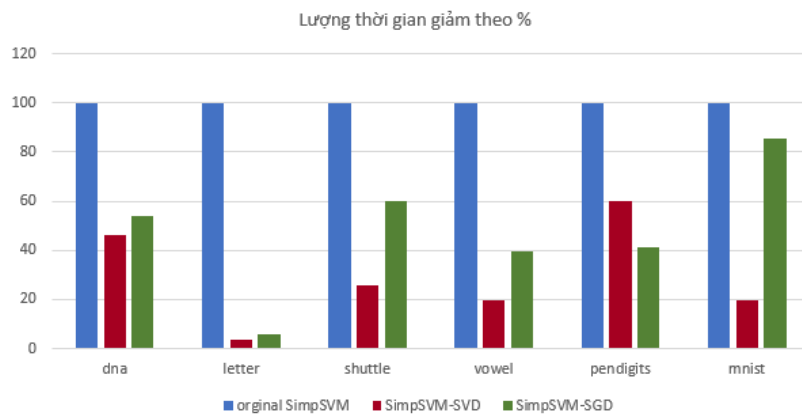
2.3 Thuật toán FastRVM

2.3.1 Thuật toán RVM2

Thuật toán cực đại likelihood biên:

Tập dữ liệu		%SVs				
		100%	50%	25%	10%	5%
Dna	# SV	654	327	164	65	33
	SimpSVM	94.44	94.52	94.18	94.44	94.86
	SimpSVM-SVD	94.44	94.44	94.18	93.59	91.91
	SimpSVM-SGD	94.44	94.01	93.42	93.09	90.73
Letter	# SV	6014	3007	1504	601	301
	SimpSVM	97.04	97.00	96.74	91.54	80.86
	SimpSVM-SVD	97.04	97.02	96.68	87.98	71.40
	SimpSVM-SGD	97.04	96.88	96.26	87.02	68.54
Shuttle	# SV	3208	1604	802	321	160
	SimpSVM	98.98	98.98	98.98	98.98	98.98
	SimpSVM-SVD	98.98	98.98	98.98	98.98	98.98
	SimpSVM-SGD	98.98	98.98	98.98	98.98	98.98
Vowel	# SV	393	197	98	39	20
	SimpSVM	60.82	60.82	59.52	38.31	37.45
	SimpSVM-SVD	60.82	61.26	58.66	38.74	35.71
	SimpSVM-SGD	60.82	60.82	58.44	39.83	39.18
Pendigits	# SV	948	474	237	95	47
	SimpSVM	98.48	98.48	98.46	98.26	95.14
	SimpSVM-SVD	98.48	98.48	98.37	97.94	95.74
	SimpSVM-SGD	98.48	98.40	98.34	96.71	91.51
Mnist	# SV	11554	5777	2889	1155	578
	SimpSVM	98.31	98.34	98.26	98.03	97.84
	SimpSVM-SVD	98.31	98.27	98.21	97.98	97.60
	SimpSVM-SGD	98.31	98.33	97.94	96.96	95.36

Bảng 2.2: Chính xác dự đoán của các SimpSVM với tốc độ tăng tốc khác nhau trên các tập dữ liệu



Hình 2.1: So sánh thời gian rút gọn giữa các SimpSVM

1. Nếu hồi quy khởi tạo cho σ^2 một giá trị hợp lý (ví dụ: 0,1).
2. Khởi tạo một vector cơ sở đơn ϕ_i , thiết lập, từ (??):

$$\alpha_i = \frac{\|\phi_i\|^2}{\|\phi_i^T t\|^2 / \|\phi_i\|^2 - \sigma^2} \quad (2.10)$$

Tất cả các α_m khác được đặt là vô cùng.

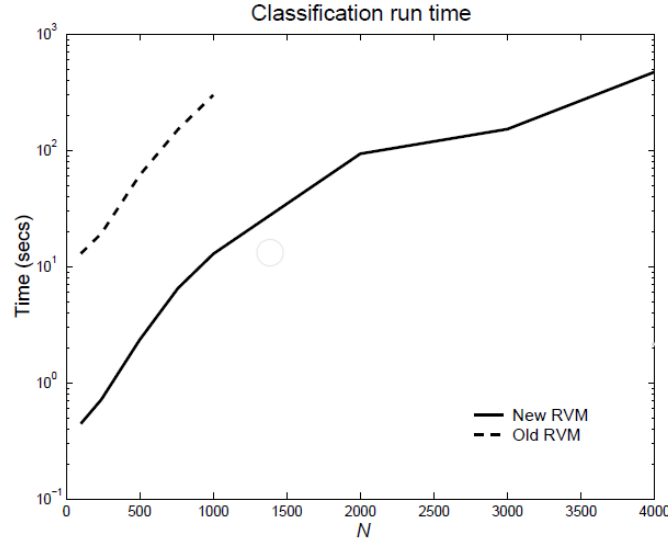
3. Tính toán Σ , μ , các giá trị khởi tạo của s_m , q_m cho toàn bộ M cơ sở ϕ_m .
 4. Chọn một vectơ cơ sở ứng viên ϕ_i từ tập hợp toàn bộ M .
 5. Tính toán $\theta_i \triangleq q_i^2 - s_i$.
 6. Nếu $\theta_i > 0$ và $\alpha_i < \infty$ (ϕ_i nằm trong mô hình), ước lượng lại α_i .
 7. Nếu $\theta_i > 0$ và $\alpha_i = \infty$, thêm ϕ_i vào mô hình với α_i cập nhật.
 8. Nếu $\theta_i \leq 0$ và $\alpha_i < \infty$, xóa ϕ_i khỏi mô hình và đặt $\alpha_i = \infty$.
 9. Nếu hồi quy và ước lượng mức nhiễu, cập nhật $\sigma^2 = \|t - y\|^2 / (N - M + \Sigma_m \alpha_m \Sigma_{mm})$.
 10. Tính toán/cập nhật Σ , μ (sử dụng phương pháp xấp xỉ Laplace trong phân lớp) và toàn bộ s_m và q_m .
 11. Nếu hội tụ chấm dứt, nếu chưa quay lại 4.
- Thuật toán trên đảm bảo tăng likelihood biên ở mỗi bước, cho đến khi đạt được một cực đại cục bộ.

2.3.2 Thuật toán đề xuất FastRVM (bRVMf)

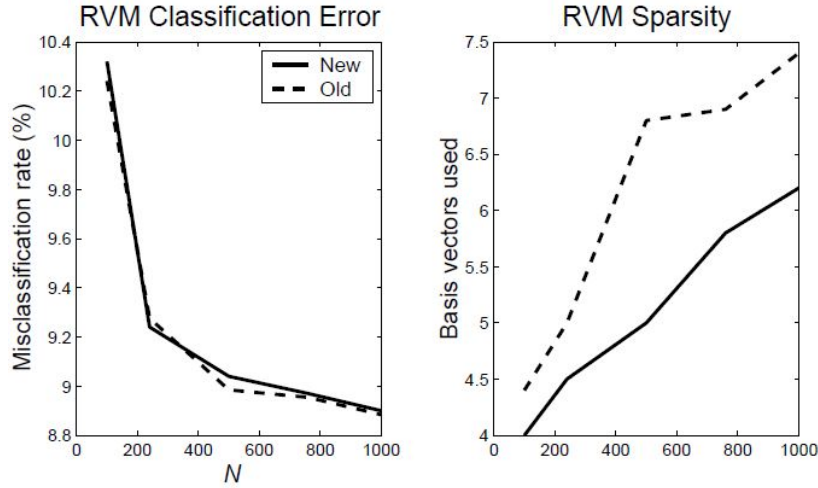
- Khởi tạo tập D_1 bằng cách chọn ngẫu nhiên B mẫu huấn luyện từ tổng toàn bộ các mẫu huấn luyện, huấn luyện D_1 bằng RVM2 thu được model M_1 . Số mẫu huấn luyện còn lại $D_r = D \setminus D_1$
- Lặp $i = 2$ đến k_{fold}
- + Chọn ngẫu nhiên $D_i = B$ số mẫu huấn luyện trong số mẫu huấn luyện còn lại D_r , $D_r = D_r \setminus D_i$
- + Dùng model M_{i-1} để tính $s_i, q_i, \theta_i (= q_i^2 - s_i)$ cho các vector cơ sở z_i trong D_i
- + Chọn ra tập $Z_i = N_B$ số z_i trong D_i có θ_i lớn nhất
- + Kết hợp M_{i-1} với Z_i , huấn luyện bằng RVM2 thu được model M_i
- Huấn luyện lại một lần cuối M_i trên toàn bộ dữ liệu huấn luyện tổng, thu được model cuối M_f

2.3.3 Thuật toán đề xuất FastRVM (bRVMdml)

- Khởi tạo tập D_1 bằng cách chọn ngẫu nhiên B mẫu huấn luyện từ tổng toàn bộ các mẫu huấn luyện, huấn luyện D_1 bằng RVM2 thu được model M_1 . Số mẫu huấn luyện còn lại $D_r = D \setminus D_1$
- Lặp $i = 2$ đến k_{fold}



Hình 2.2: So sánh thời gian huấn luyện phân lớp giữa RVM2 và RVM



Hình 2.3: So sánh lỗi phân lớp và độ thưa mô hình giữa RVM2 và RVM

- + Chọn ngẫu nhiên $D_i = B$ số mẫu huấn luyện trong số mẫu huấn luyện còn lại $D_r, D_r = D_r \setminus D_i$
- + Dùng model M_{i-1} để tính $s_i, q_i, \theta_i (= q_i^2 \sim s_i)$ cho các vector cơ sở z_i trong D_i
- + Tính độ thay đổi trong log-likelihood ΔLi đối với $\mathcal{L}(\alpha)$ cho các vector cơ sở z_i trong D_i
- + Chọn ra tập $Z_i = N_B$ số z_i trong D_i có ΔLi lớn nhất
- + Kết hợp M_{i-1} với Z_i , huấn luyện bằng RVM2 thu được model M_i
- Huấn luyện lại một lần cuối M_i trên toàn bộ dữ liệu huấn luyện tổng, thu được model cuối M_f

2.3.4 Thực nghiệm

Trước tiên, thực nghiệm so sánh hiệu năng giữa thuật toán RVM2 và thuật toán RVM gốc. Kết quả thời gian chạy trung bình trên 10 tập dữ liệu được sinh

ngẫu nhiên như trong Hình 2.2. So sánh thời gian huấn luyện ở tập dữ liệu với số mẫu $N = 1000$, trong khi thuật toán RVM gốc cần 298 giây thì thuật toán RVM2 chỉ mất 12.84 giây. Bên cạnh đó, kết quả so sánh về lỗi khái quát hóa và độ thưa của RVM2 và RVM gốc được thể hiện trong Hình 2.3. Điều đó cho thấy RVM2 là một thuật toán FastRVM rất nhanh so với thuật toán RVM gốc trong khi độ chính xác và độ thưa của mô hình đạt được gần như là tương đương.

Dữ liệu thực nghiệm

Chúng tôi đã chọn các tập dữ liệu đa lớp khác nhau có sẵn công khai từ thư viện học máy UCI bao gồm: dna, satimages, pendigits, usps.

Đánh giá hiệu năng

Đánh giá hiệu năng của các thuật toán tập trung vào hai phương diện quan trọng của các giải thuật, đó là hiệu năng phân lớp và thời gian huấn luyện.

Lựa chọn tham số

Chúng tôi sử dụng tìm kiếm lưới để lựa chọn tham số huấn luyện sao cho bộ phân lớp RVM đạt được độ chính xác dự đoán cao nhất trên các tập dữ liệu test.

Kết quả thực nghiệm

Kết quả thực nghiệm so sánh độ chính xác phân lớp và thời gian huấn luyện của các thuật toán FastRVM trên các tập dữ liệu có sẵn được chỉ ra trong Bảng 2.3. Như có thể thấy trong bảng đó:

- + Hai thuật toán FastRVM đề xuất thu được độ chính xác phân lớp tương tự như của RVM2 do đó thể hiện chúng là hướng đi đúng.

- + Số lượng RV cũng tương tự như RVM2 nên vẫn giữ được độ thưa của lời giải hay độ nhanh trong pha test của RVM.

- + Thời gian để hội tụ (hay huấn luyện) nhỏ hơn nhiều so với RVM2 trong toàn bộ các trường hợp. Trên các tập dữ liệu chuẩn đó, các thuật toán FastRVM đề xuất có thể chạy nhanh hơn gấp hàng chục lần so với thuật toán RVM2 trong khi chúng ta cần nhắc lại rằng RVM2 đã là một thuật toán RVM nhanh.

Từ đó chúng tôi các thuật toán FastRVM đề xuất có thể tăng tốc pha huấn luyện RVM rất nhanh trong khi giữ được độ thưa của nghiệm (hay độ nhanh của pha test) và bảo toàn được nhiều nhất có thể độ chính xác dự đoán.

Một vấn đề xem xét khác ở đây là ứng dụng RVM cho các tập dữ liệu lớn hơn, trong đó ma trận thiết kế Φ sẽ không chứa được trong bộ nhớ và không thể sử dụng thuật toán RVM2 ban đầu. Trong khi đó, hai thuật toán FastRVM đề xuất nhờ lấy mẫu tập con dữ liệu, ma trận thiết kế Φ dựa trên tập con dữ

Tập dữ liệu		RVM2	bRVmf	bRVmdml
DNA	Acc(%)	92.66	93.09	92.16
	Times	436	48	49
	RVs	168	146	157
Pendigits	Acc(%)	97.91	97.88	97.68
	Times	14490	666	604
	RVs	141	136	136
Satimages	Acc(%)	89.60	88.90	88.95
	Times	2176	42	45
	RVs	111	87	90
USPS	Acc(%)	93.97	94.22	94.37
	Times	45316	2763	2681
	RVs	321	309	307

Bảng 2.3: So sánh kết quả của các thuật toán FastRVM

liệu nhỏ hơn có thể chứa được trong bộ nhớ, thuật toán thực hiện được nên vẫn có thể vượt qua được các tập dữ liệu lớn hơn.

2.4 Kết chương

Chương này tập trung vào việc tìm hiểu các thuật toán huấn luyện nhanh các mô hình RBF thưa và đề xuất một số thuật toán mới tăng tốc SVM và RVM. Các thuật toán đề xuất stochastic SimpSVM nhằm mục đích đẩy nhanh việc huấn luyện, xây dựng mô hình SimpSVM giản lược với ít hơn đáng kể các SVs, giúp tăng tốc pha test của SVM. Các thuật toán đề xuất FastRVM nhằm mục đích rút ngắn thời gian huấn luyện xây dựng mô hình RVM đồng thời tăng cường khả năng làm việc của RVM trên các tập dữ liệu lớn hơn. Hiệu năng của các thuật toán mới này được xác định thông qua thực nghiệm. Cụ thể, các phương pháp đề xuất được thực nghiệm trên các bộ dữ liệu chuẩn đang được dùng phổ biến hiện nay. So sánh kết quả phân lớp sử dụng các thuật toán gốc và các thuật toán cải tiến mới cho thấy các thuật toán cải tiến mới hoặc tăng tốc được pha test của thuật toán hoặc tăng tốc được pha huấn luyện của thuật toán với độ chính xác được đảm bảo, không thay đổi nhiều. Các nghiên cứu được nêu trong chương này đã được tổng hợp và công bố trong các công trình [CT4, CT5, CT6] tại các tạp chí và kỷ yếu hội nghị quốc tế có phản biện.

Chương 3

Ứng dụng mô hình RBF thưa cho bài toán phân loại đối tượng

3.1 Bài toán nhận dạng ngôn ngữ ký hiệu

Các ký hiệu trong ngôn ngữ Auslan được ghi lại bằng cách sử dụng các ngón tay dụng cụ cùng với các cảm biến.

3.1.1 Mô tả tập dữ liệu

Tập dữ liệu bao gồm các mẫu là chuỗi các vectơ đặc trưng 22 chiều. Toàn bộ các mẫu dữ liệu đều được gắn nhãn với các từ ký hiệu tương ứng.

3.1.2 Trích xuất đặc trưng

Từ 22 kênh thông tin, chúng tôi sử dụng phương pháp trích xuất các đặc trưng "global" và "meta".

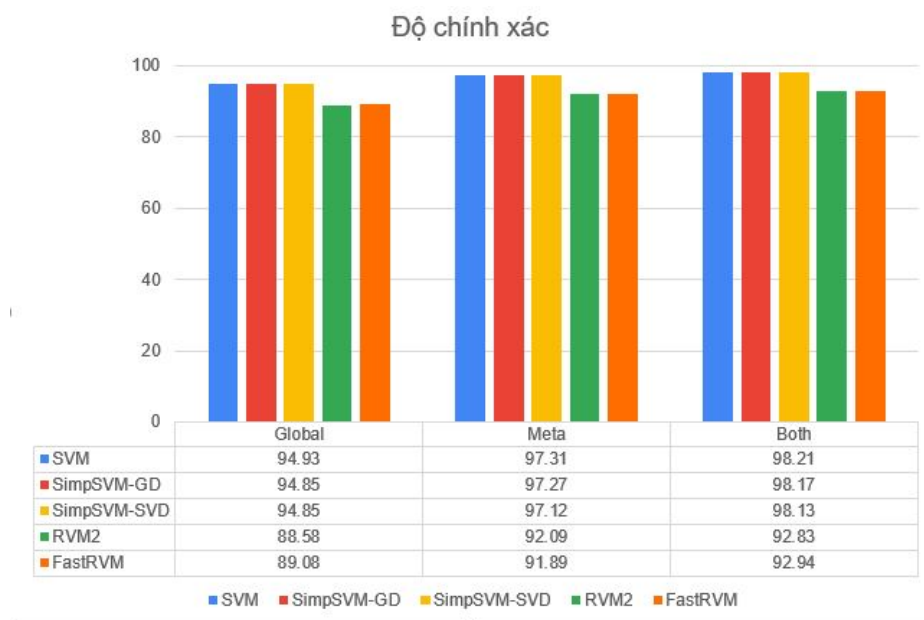
3.1.3 Lựa chọn tham số

Sử dụng phương pháp tìm kiếm lưới để tìm các giá trị C , γ phù hợp.

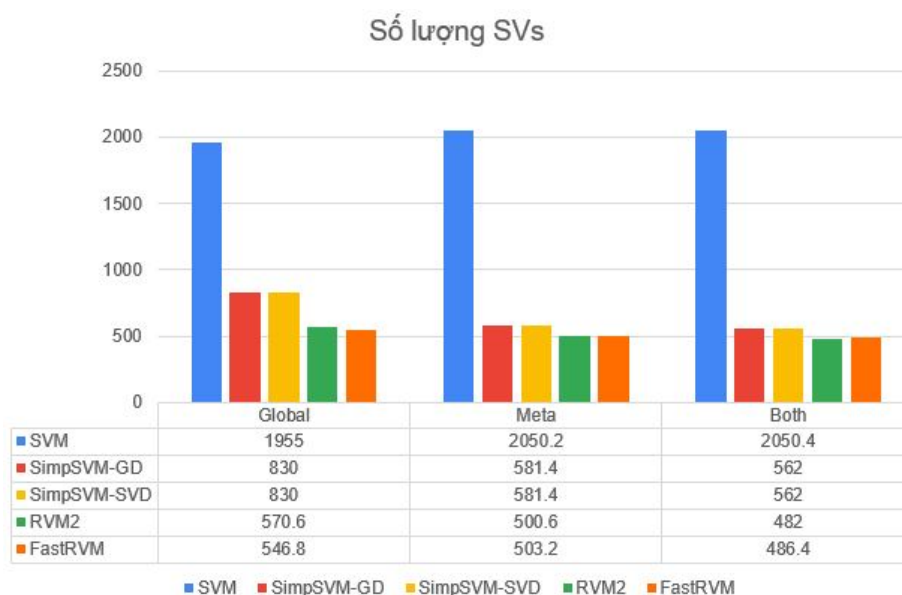
3.1.4 Phân lớp ký hiệu

Thực hiện phương pháp kiểm tra chéo 5-fold.

- Về hiệu năng nhận dạng: trong Hình 3.1 và Hình 3.2, chúng tôi so sánh hiệu năng dự đoán và số hàm cơ sở của các mô hình SVM, SimpSVMs và RVMs trên 3 kiểu đặc trưng: global, meta và kết hợp. Các kết quả chỉ ra rằng khi số lượng các đặc trưng tăng, toàn bộ ba phương pháp đều tăng độ chính xác. Với số vectơ yêu cầu nhỏ hơn nhiều, trong pha test, các phương pháp SimpSVMs và RVMs sẽ nhanh hơn đáng kể bởi vì số lượng tính toán ít hơn.



Hình 3.1: Độ chính xác của các mô hình SVM, SimpSVM và RVM trên 3 kiểu đặc trưng

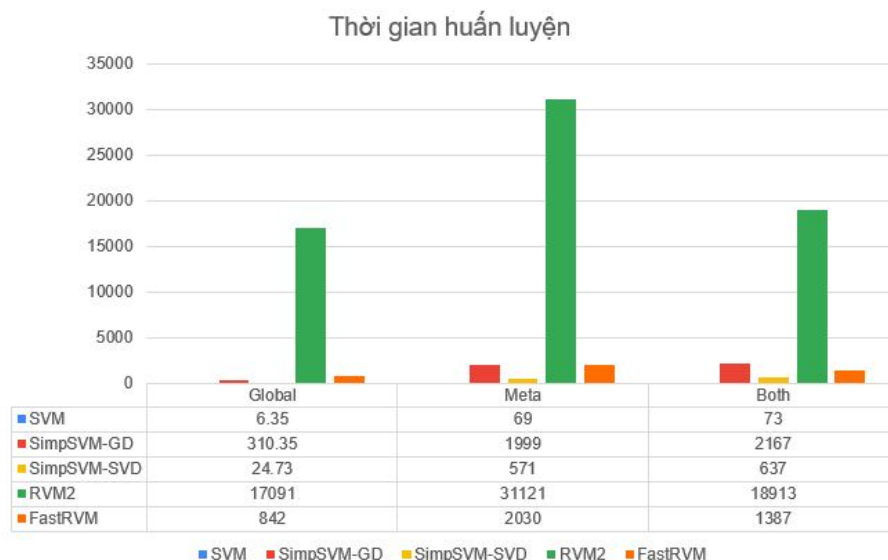


Hình 3.2: Số hàm cơ sở của các mô hình SVM, SimpSVM và RVM trên 3 kiểu đặc trưng

- Về thời gian huấn luyện: Hình 3.3 cho thấy các thuật toán đề xuất Stochastic SimpSVM, FastRVM đều có thời gian huấn luyện nhanh hơn so với các thuật toán gốc đến hàng chục lần trong khi độ chính xác không mất mát nhiều và số SVs cuối gần như tương đương.

3.2 Bài toán nhận dạng cử chỉ người

Gần đây, nhận dạng cử chỉ/hành động của con người đã thu hút nhiều sự chú ý khi các thiết bị ghi lại chuyển động trở nên ít tốn kém hơn và bộ công cụ phát triển phần mềm (SDK) được các công ty khổng lồ hỗ trợ rộng rãi.



Hình 3.3: Thời gian huấn luyện của các mô hình SVM, SimpSVM và RVM trên 3 kiểu đặc trưng

3.2.1 Tập dữ liệu

Bộ dữ liệu Microsoft Research Cambridge-12 Kinect bao gồm chuỗi các chuyển động của con người, được biểu thị dưới dạng các vị trí của bộ phận cơ thể và cử chỉ liên quan được hệ thống nhận dạng.

3.2.2 Trích xuất đặc trưng

Từ hai mươi tọa độ 3D của các khớp xương, chúng tôi sử dụng ba loại thông tin trích xuất: vận tốc điểm tương đối, góc khớp và vận tốc góc khớp.

3.2.3 Lựa chọn tham số

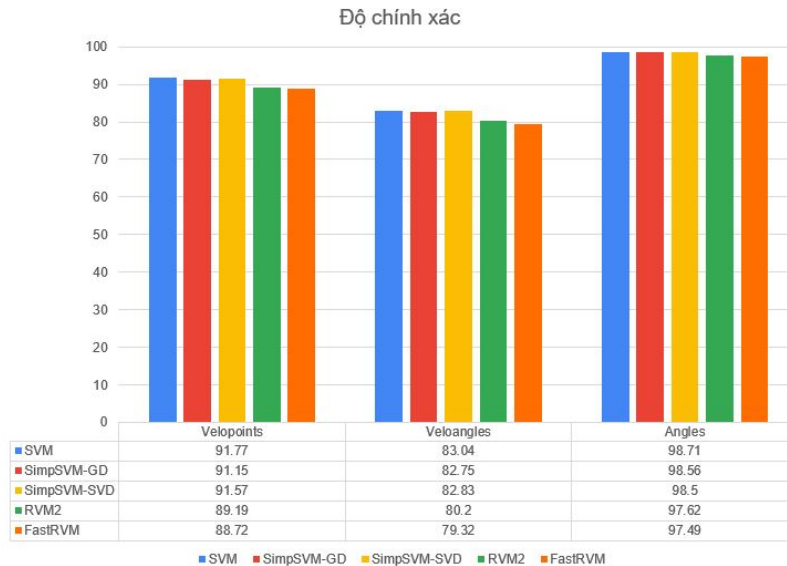
Sử dụng phương pháp tìm kiếm lưới để tìm các giá trị C , γ phù hợp.

3.2.4 Phân lớp cử chỉ, hành động

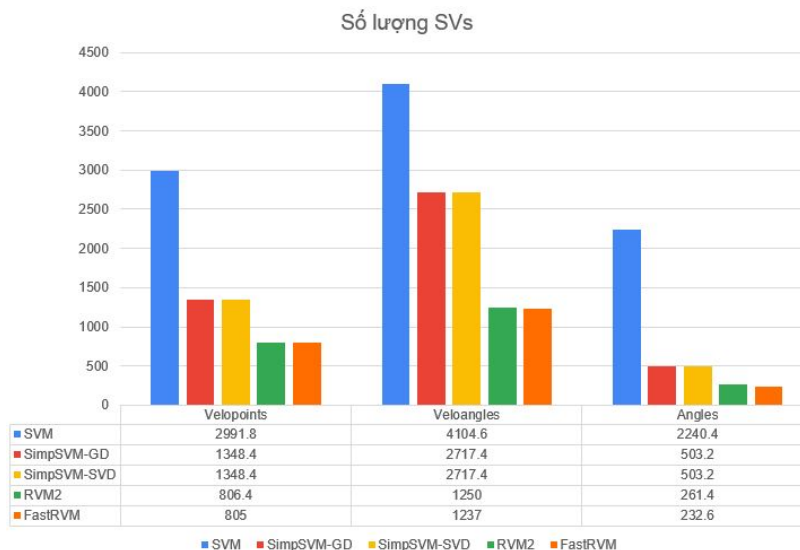
Từ các kết quả thực nghiệm, chúng ta có thể nhận thấy những điểm sau:

- Về độ chính xác nhận dạng: Hình 3.4 cho thấy RVM2 kém chính xác nhất trên cả ba bộ đặc trưng, có thể do RVM2 loại bỏ một tỷ lệ lớn các hàm cơ sở làm giảm độ chính xác dự đoán nhiều hơn. Tuy nhiên, trong Hình 3.5, số lượng SVs nhỏ nhất được yêu cầu của RVM2, so với SVM dẫn đến RVM2 sẽ dự đoán rất nhanh trong pha test. Ngoài ra, SimpSVM giúp giảm đáng kể số lượng SV của SVM ban đầu trong khi vẫn giữ được độ chính xác của nó.

- Về thời gian huấn luyện: Hình 3.6 cho thấy các thuật toán đề xuất Stochastic SimpSVM, FastRVM có thời gian huấn luyện nhanh hơn so với các thuật toán gốc trong khi độ chính xác được bảo toàn và số SVs cuối gần như tương đương.



Hình 3.4: Độ chính xác của các mô hình học được bởi SVM, SimpSVM và RVM trong phân lớp cử chỉ, hành động



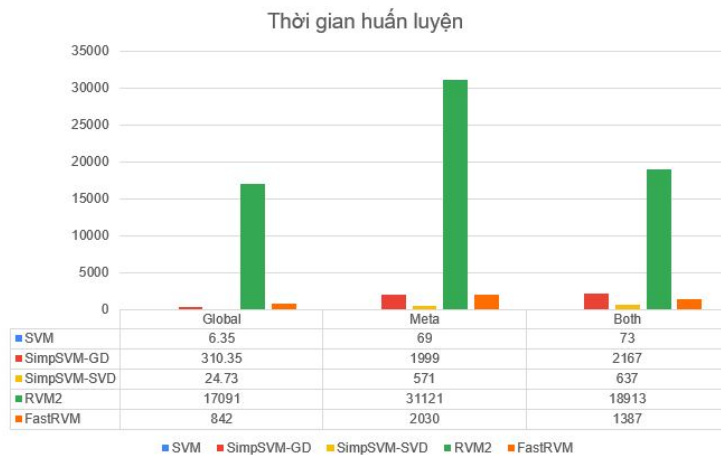
Hình 3.5: Số hàm cơ sở của các mô hình học được bởi SVM, SimpSVM và RVM trong phân lớp cử chỉ, hành động

3.3 Bài toán nhận dạng ảnh hoa

Mục đích của nhận dạng ảnh hoa là xác định loài hoa phù hợp cho ảnh dựa trên các đặc trưng của nó.

3.3.1 Tập dữ liệu

Chúng tôi sử dụng 2 tập dữ liệu ảnh hoa Oxford 17-Flowers và 102-Flowers của Đại học Oxford. Các ảnh đa dạng về kích cỡ, góc chụp và thay đổi ánh sáng. Trong các lớp ảnh cũng có sự biến thiên lớn của ảnh trong cùng một lớp hay ảnh gần giống với ảnh ở lớp khác. Một số ví dụ ảnh hoa như trong Hình 3.7.



Hình 3.6: Thời gian huấn luyện của các mô hình học được bởi SVM, SimpSVM và RVM trong phân lớp cử chỉ, hành động



Hình 3.7: Minh họa ví dụ về ảnh các loài hoa trong 2 tập dữ liệu ảnh hoa Oxford

3.3.2 Trích xuất đặc trưng

Chúng tôi sử dụng một CNN tên là ResNet18 để trích xuất các đặc trưng của ảnh hoa bằng cách chỉ giữ lại các layer ConvNet trong CNN và bỏ đi các layer FC. Sau đó dùng đầu ra của các layer ConvNet còn lại như là các đặc trưng đầu vào để phân lớp bằng các mô hình RBF thưa SVM, RVM...

3.3.3 Lựa chọn tham số

Sử dụng phương pháp tìm kiếm lưới để tìm các giá trị C , γ phù hợp.

3.3.4 Phân lớp ảnh hoa

Các kết quả thực nghiệm cụ thể được báo cáo lại trong Bảng 3.1. Từ các kết quả thực nghiệm đó, chúng ta có thể nhận thấy những điểm sau:

- Về độ chính xác nhận dạng: các mô hình lai RBF nói chung đặc biệt là mô

Methods	Oxford 17-Flowers		
	Acc	SVs	Times
CNN	96.32	#N/A	#N/A
CNN-SVM	97.06	789	0.69
CNN-SimpSVM-GD	96.69	395	386.69
CNN-SimpSVM-SVD	96.69	395	167.69
CNN-RVM2	95.99	137	193
CNN-FastRVM	96.32	136	27.25

Bảng 3.1: Hiệu năng phân loại ảnh hoa của các phương pháp

hình CNN-SVM cho kết quả nhận dạng tốt hơn mô hình CNN nguyên bản do kết hợp được ưu điểm của các đặc trưng được trích xuất tự động bởi mạng học sâu CNN với sức mạnh của bộ phân lớp SVM. Mô hình RVM vẫn là các máy kém chính xác nhất trên tập đặc trưng học được qua mạng học sâu ResNet của nhiệm vụ phân lớp ảnh hoa có thể do RVM loại bỏ một tỷ lệ đáng kể các hàm cơ sở, do đó làm giảm độ chính xác dự đoán nhiều hơn. Tuy nhiên, số lượng SV nhỏ nhất được yêu cầu của RVM, so với SVM dẫn đến RVM sẽ dự đoán rất nhanh trong pha test. Ngoài ra, SVM giản lược (SimpSVM) giúp giảm đáng kể số lượng SV của SVM ban đầu trong khi vẫn bảo toàn độ chính xác của nó (tỷ lệ lỗi tăng không đáng kể).

- Về thời gian huấn luyện: thời gian huấn luyện cũng được đo để so sánh tốc độ của các phương pháp được xem xét. Bảng 3.1 báo cáo thời gian huấn luyện của các mô hình lai CNN-SVM, CNN-SimpSVM, CNN-RVM và CNN-FastRVM. Có thể thấy, các thuật toán đề xuất Stochastic SimpSVM, FastRVM đều có thời gian huấn luyện nhanh hơn so với các thuật toán gốc từ hai, ba đến hàng chục lần trong khi độ chính xác không mất mát nhiều và số SVs cuối gần như tương đương.

3.4 Kết chương

Chương này tìm hiểu, khảo sát các nghiên cứu, phương pháp liên quan đến bài toán nhận dạng ngôn ngữ ký hiệu và bài toán nhận dạng cử chỉ, hành động. Từ đó, xây dựng giải pháp nhận dạng ngôn ngữ ký hiệu, cử chỉ, hành động dựa trên dữ liệu thu được từ các cảm biến và xử lý nhận dạng thông qua các mô hình RBF thưa (các mô hình SVM, RVM cùng với các thuật toán đề xuất stochastic SimpSVM, FastRVM trong Chương 2). Kết quả cho thấy các mô hình RBF thưa hoàn toàn khả thi và có thể đạt được các kết quả nhận dạng tốt khi ứng dụng vào trong các bài toán nhận dạng ngôn ngữ ký hiệu hay cử chỉ, hành động. Các nghiên cứu trong chương này đã được tổng hợp và công bố trong các công trình [CT1, CT2, CT3] tại các kỷ yếu hội nghị quốc tế có phản biện.

Kết luận

Luận án đã trình bày khái quát về các vấn đề cơ bản liên quan đến mô hình RBF thưa nói chung và khảo sát hai mô hình RBF thưa cụ thể: mô hình RBF dựa trên lẽ cực đại SVM và mô hình RBF dựa trên phép suy luận Bayes RVM.

Mỗi mô hình đều có những ưu điểm và hạn chế nhất định. Mô hình SVM có ưu điểm huấn luyện nhanh nhưng hạn chế là pha test chậm do số lượng lớn SVs bao gồm trong nghiệm. Mô hình RVM, ngược lại, có ưu điểm là pha test nhanh vì nghiệm rất thưa, thường chỉ bao gồm số ít các RVs nhưng thời gian huấn luyện lại chậm.

Với định hướng mục tiêu nâng cao hiệu năng các phương pháp phân loại đối tượng, luận án đã tiến hành nghiên cứu, đề xuất một số thuật toán mới cho phép nâng cao hiệu năng các phương pháp mô hình RBF. Cụ thể, các thuật toán stochastic SimpSVM giúp giảm lược nghiệm SVM để bao gồm ít hơn các SVs, từ đó giúp tăng tốc pha test của SVM trong khi vẫn bảo toàn được độ chính xác phân loại. Các thuật toán FastRVM giúp tăng tốc thời gian huấn luyện RVM nhanh hơn trong khi cũng bảo toàn được độ chính xác phân loại.

Luận án cũng đã tiến hành thực nghiệm trên các tập dữ liệu chuẩn được sử dụng rộng rãi cũng như dữ liệu của hai bài toán thực tế (nhận dạng ngôn ngữ ký hiệu và nhận dạng cử chỉ, hành động) để khẳng định các thuật toán đó đáp ứng được mục tiêu đã đặt ra.

Những đóng góp mới của luận án:

- Kết quả phân tích, so sánh và đánh giá các phương pháp huấn luyện mô hình RBF thưa đối với phân loại đối tượng và yêu cầu tính toán khác nhau. Những kết luận về sự phù hợp của các phương pháp này trong những điều kiện hoàn cảnh cụ thể.
- Đề xuất và phát triển các thuật toán stochastic SimpSVM hiệu quả nhằm đẩy nhanh việc huấn luyện, xây dựng mô hình SimpSVM giảm lược nghiệm SVM với ít hơn đáng kể các SVs, từ đó tăng tốc pha test của SVM trong khi giữ được độ chính xác phân lớp không thay đổi nhiều.
- Đề xuất và phát triển các thuật toán FastRVM hiệu quả nhằm đẩy nhanh việc huấn luyện RVM lên nhiều lần trong khi đảm bảo được độ chính xác phân loại cuối. Đồng thời, tăng cường khả năng làm việc của phương pháp RVM hiện tại trên các tập dữ liệu lớn hơn. Những phương pháp được đề

xuất sẽ đóng góp và sự phát triển chung của cộng đồng nghiên cứu về mô RBF nói riêng và trong lĩnh vực học máy hay trí tuệ nhân tạo nói chung.

- Góp phần nghiên cứu ứng dụng các mô hình RBF thưa trong nhận dạng đối tượng, cụ thể là phân loại đối tượng trong các bài toán thực tế như nhận dạng ngôn ngữ ký hiệu, nhận dạng cử chỉ, hành động...

DANH MỤC CÁC CÔNG TRÌNH ĐÃ CÔNG BỐ

1. Pham Q.T., Nguyen D.D., Nguyen T.T., *A comparison of SimpSVM and RVM on sign language recognition*, In the Proceedings of the International Conference on Machine Learning and Soft Computing (ICMLSC 2017), ISBN: 978-1-4503-4828-7, ACM, 2017, pp. 98-104 (Scopus).
2. Pham Q.T., Nguyen T.T., Hoang T.L., *The SVM, SimpSVM and RVM on sign language recognition problem*, In the Proceedings of the 7th International Conference on Information Science and Technology (ICIST 2017), ISBN: 978-1-5090-5401-5, IEEE, 2017, pp. 398-403.
3. Nguyen D.H., Pham Q.T., Nguyen D.D., *RBF Models with Shallow and Deep Feature for Skeleton-based Human Gesture Recognition*, In the Proceedings of the National Foundation for Science and Technology Development (NAFOSTED) Conference on Information and Computer Science (NICS), IEEE, 2017, pp. 72-77.
4. Pham Q.T., Nguyen T.T., Hoang T.L., *A Modification of Solution Optimization in Support Vector Machine Simplification for Classification*, In: Bhateja V., Nguyen B., Nguyen N., Satapathy S., Le DN. (eds) Information Systems Design and Intelligent Applications. Advances in Intelligent Systems and Computing, vol 672. Springer, Singapore, 2018, pp. 149-158 (ISI Web of Science, Scopus).
5. Pham Q.T., Hoang T.L., Nguyen T.T., *Improving Simplification of Support Vector Machine for Classification*, International Journal of Machine Learning and Computing, vol. 8, no. 4, pp. 372-376, 2018 (Scopus, Q3).