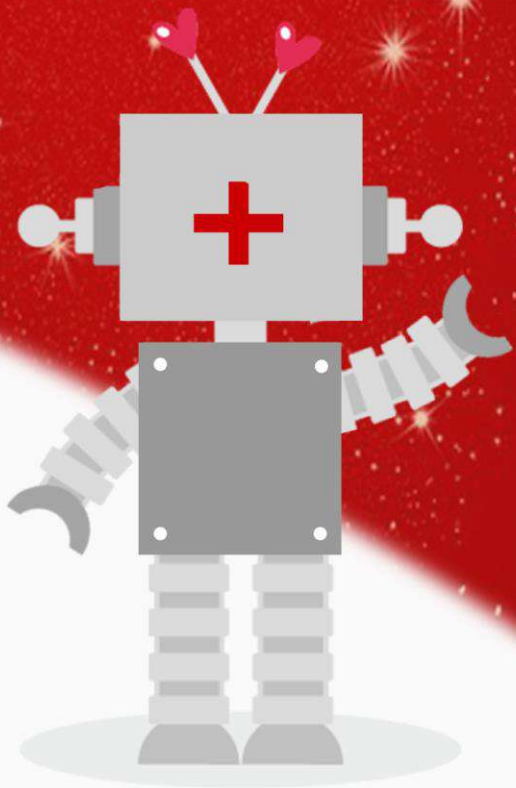
Dịch và biên soạn: NGUYỄN HỮU HƯNG - DƯƠNG LỰC
từ sách: Fundamental of artificial intelligence - Tang Xiaoou

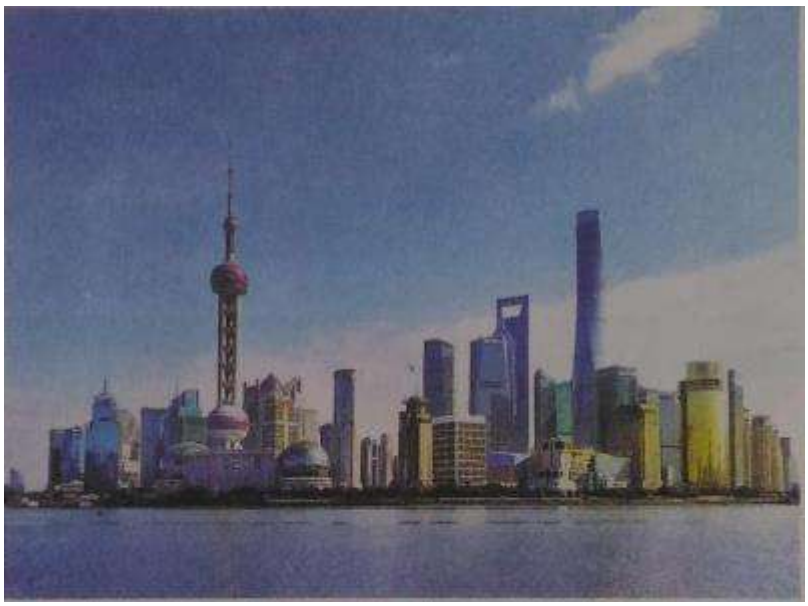
Kiến thức cơ bản về Trí tuệ nhân tạo

Dành cho học sinh phổ thông



Mục lục

Chương I. Trí tuệ nhân tạo mở ra một kỷ nguyên mới



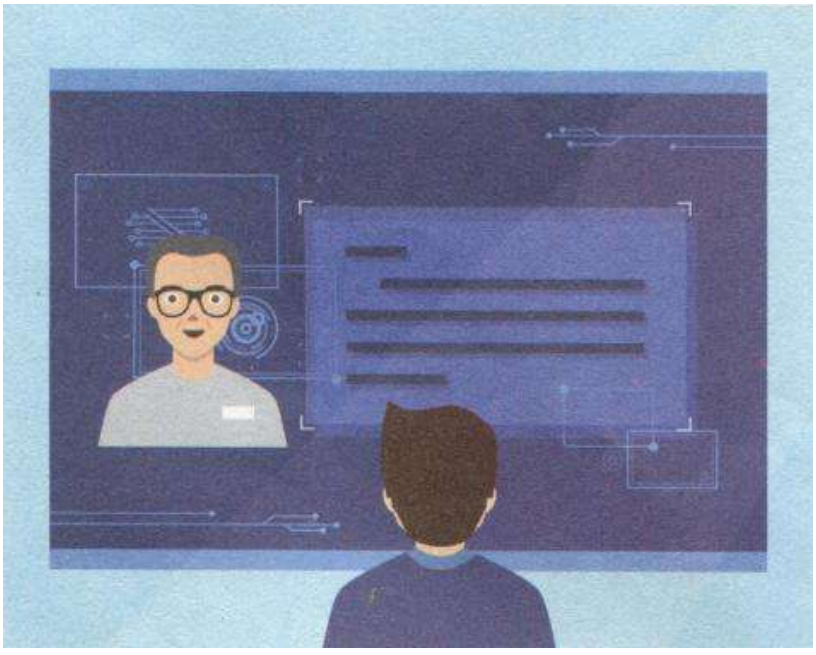
Đây là thời đại công nghệ có những thay đổi và bước tiến vượt bậc. Tất cả mọi người đang tận hưởng cuộc sống thoải mái và thuận tiện. Đằng sau đó là một làn sóng công nghệ thay đổi sâu sắc cuộc sống và xã hội của chúng ta - trí tuệ nhân tạo.

Khi bạn hỏi loa thông minh về thời tiết, nó hiểu được vấn đề của bạn thông qua công nghệ nhận dạng giọng nói. Khi bạn sử dụng smartphone, nó có thể tự động mở khóa vì được nhận dạng người đó thông qua xác minh khuôn mặt. Khi bạn mở một trang web thương mại điện tử, bạn có thể thấy mục yêu thích của mình ngay từ đầu vì nó hiểu được sự quan tâm của bạn thông qua công nghệ phân tích dữ liệu từ hồ sơ mua hàng của bạn. Khi bạn lái xe trên một đường cao tốc, hệ thống thông minh trong xe cũng âm thầm bảo vệ bạn, liên tục theo dõi các mối nguy hiểm có thể và kịp thời cảnh báo bạn.

Tất cả điều này chỉ là một khởi đầu. Nhiều cảnh trong khoa học viễn tưởng vẫn còn cách đây một thập kỷ đã trở thành trải nghiệm cuộc sống thực của chúng ta. Giờ đây khi được thúc đẩy bởi làn sóng trí thông minh nhân tạo, sau 10 năm nữa chúng ta sẽ sống trong thế giới như thế nào?

1.1. Thời gian và không gian: Một ngày của Minh

Bắt đầu một ngày



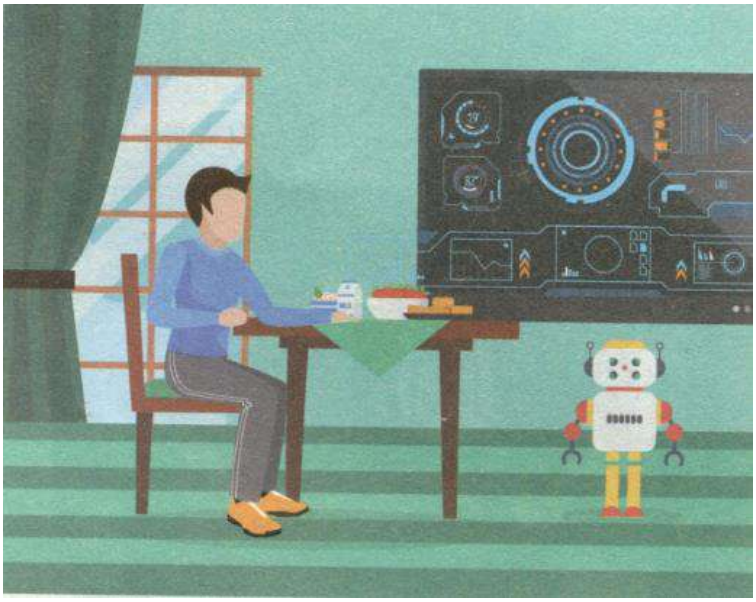
Một buổi sáng vào năm 2028, một tia nắng chiếu vào phòng ngủ, và Minh nghe thấy một giọng nói nhẹ nhàng: "Minh, bây giờ là 7 giờ sáng ngày 29 tháng 3 năm 2028, một ngày mới bắt đầu!"

Minh đã quen thuộc với âm thanh này, nó được phát ra từ phòng ngủ được điều khiển bởi hệ thống nhà thông minh. Giống như một người quản lý trung thành, hệ thống này sẽ chăm sóc cuộc sống của Minh ngày này qua ngày khác. Minh từ từ ngồi dậy khỏi giường. Khi mở mắt, anh thấy màn hình chiếu phía trước sáng lên, và lời chào của cha anh phát ra từ màn hình.

Cha của Minh là một nhà khoa học về trí thông minh nhân tạo nổi tiếng. Từ khi còn nhỏ, Minh đã tiếp xúc với công nghệ trí tuệ nhân tạo từ phòng thí nghiệm của cha mình. Một lần, Minh thấy bức ảnh của anh đã được tạo bằng những mảnh ghép trong máy tính của cha anh. Các hoạt động, tiếng cười, sự ghen tị của anh ấy, rất trông rất chân thật. Người cha nói với anh rằng tất cả điều này được thực hiện tự động bởi máy tính thông qua công nghệ thông minh, khiến anh cảm thấy ngạc nhiên không thể giải thích - khoảnh khắc bình thường này không chỉ mở ra một thế giới đầy bí ẩn, mà còn khiến anh nhận ra tình yêu người cha bận rộn với công việc dành cho anh.

Minh quyết tâm sử dụng trí thông minh nhân tạo để mang lại lợi ích cho nhiều người. Vì vậy, anh đã chọn để trở thành một kỹ sư trí tuệ nhân tạo.

Bữa ăn sáng: Bữa tiệc thông tin



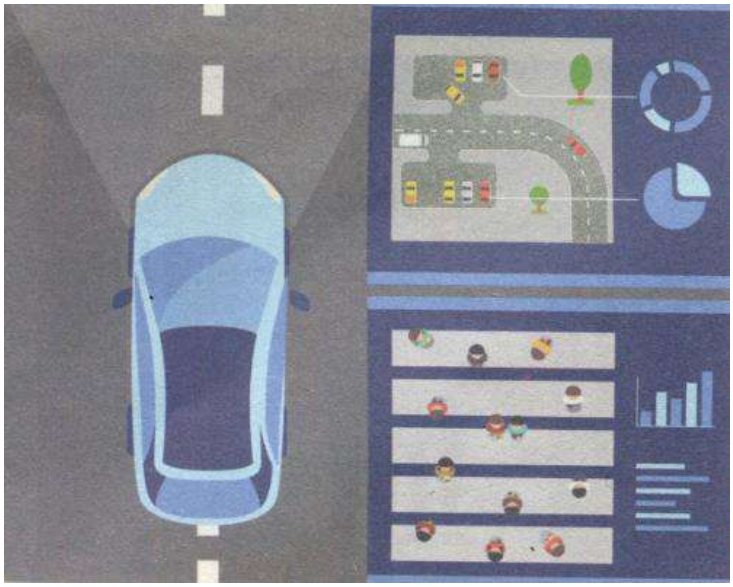
Sau khi thức dậy, Minh đến nhà hàng. Các robot nấu ăn đã chuẩn bị một bữa ăn sáng cân bằng dựa trên dữ liệu hệ thống giám sát tình báo sức

khỏe trong vài ngày qua: Một tách trà sữa, một món salad, ngoài ra còn có hai miếng bánh mì yêu thích. Một bữa ăn sáng lành mạnh và ngon miệng khiến anh cảm thấy tràn đầy năng lượng và hạnh phúc.

Khi anh ăn sáng, màn hình của nhà hàng bắt đầu phát sóng một bản tóm tắt tin tức trong ngày. Đây là thời đại bùng nổ thông tin, thành phố tạo ra nhiều thông tin hơn mỗi ngày so với thế giới kết hợp cách đây mười năm. Tuy nhiên, Minh không lo lắng về điều này. Một hệ thống lưu thông tin hiệu quả và được cá nhân hóa liên tục phát hiện ra những điều quan tâm của anh ta từ số lượng lớn thông tin mới mỗi ngày, sau đó trình bày chúng một cách thuận tiện và nhanh chóng.

Trong thời đại này, các công cụ tìm kiếm ít quan trọng hơn, các mạng thông minh mới nổi dần dần thay thế Internet truyền thống. Họ cung cấp thông tin cho mọi người vào đúng thời điểm, đúng nơi và hiệu quả chưa từng thấy.

Trên đường đi làm: Sự thoải mái của xe hơi



Từ nhà, Minh nhìn thấy chiếc xe điện màu xanh yêu quý của anh đã dừng lại ở trước nhà. Chiếc xe ở trong nhà để xe vào ban đêm. Hệ thống nhà thông minh đã quan sát những hành động của Minh, trước khi anh ra ngoài, hệ thống sẽ để chiếc xe tự động đợi trước nhà.

Khi Minh đến trước xe, cánh cửa tự động mở ra. Sau khi lên xe, cửa sẽ tự động đóng lại. Đằng sau hoạt động dường như đơn giản này là một mô-

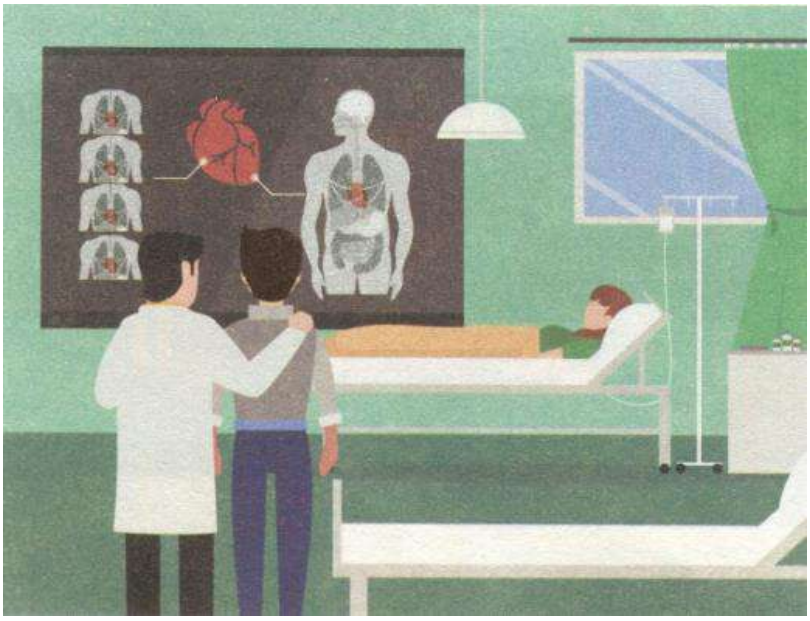
đun tự động nhận dạng chuyển động. Trong con mắt của chủ sở hữu chiếc xe, mọi thứ hoạt động một cách tự nhiên.

Tuy nhiên, khi một người lạ được phát hiện lại gần, chiếc xe sẽ đóng cửa và cảnh báo cho trung tâm an ninh.

Trong xe, Minh nghe thấy một giọng nói nhẹ nhàng: "Minh, tôi rất vui được gặp lại bạn. Bây giờ bạn có đi làm không? "

Sau khi xác nhận được xác nhận, chiếc xe bắt đầu di chuyển. Bởi vì một lễ hội vừa mới diễn ra trong thành phố, có rất nhiều xe cộ và người đi bộ trên đường ngày hôm nay.Nhưng điều này không mang lại nhiều thách thức cho chiếc xe của Minh. Với sự trợ giúp của bảng mạch và cảm biến video trên mọi hướng, hệ thống lái xe sẽ phát hiện chính xác hướng của từng xe khác và mọi người đi bộ trên đường, sau đó điều chỉnh chính xác tốc độ và hướng đi.

Đến thăm bệnh viện: Trí thông minh chăm sóc cho cuộc sống



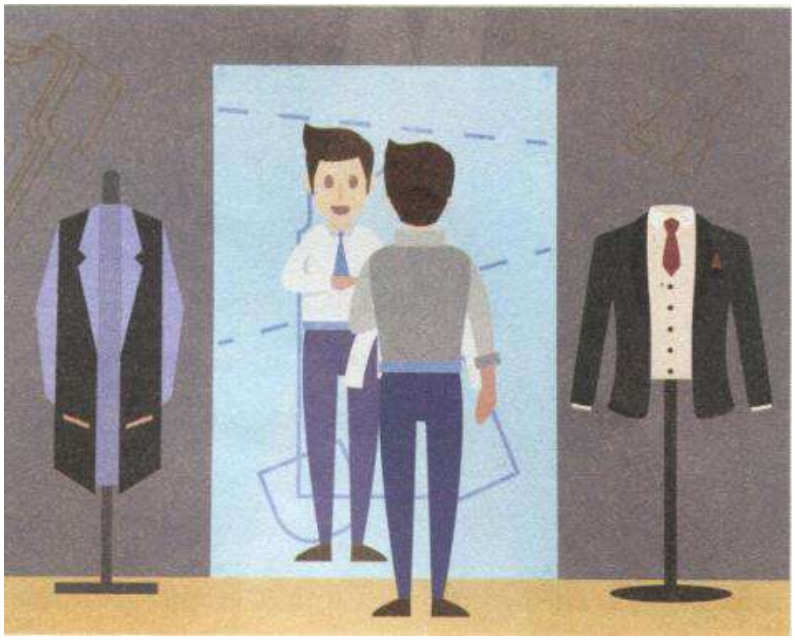
Hôm nay, Minh có một người bạn là một bác sĩ tại bệnh viện, anh đến thăm để xem trí thông minh nhân tạo có thể giúp bác sĩ như thế nào. Ở đây, anh đã nhìn thấy một trường hợp quan trọng được giới thiệu ở đây để tham khảo ý kiến.

Hình ảnh MRI của tim bệnh nhân được hiển thị trên màn hình lớn. Hệ thống phân tích hình ảnh y tế thông minh vừa được nâng cấp tại bệnh viện đã quét và phân tích hình ảnh, các khu vực trọng điểm đã được đánh dấu cẩn

thận. Hệ thống này tích hợp kết quả nhận dạng mẫu hình ảnh y tế mới nhất để phát hiện chính xác hàng trăm mẫu tổn thương khác nhau. Với sự giúp đỡ của hệ thống này, các bác sĩ nhanh chóng xác định nguyên nhân, sau đó làm việc với các đồng nghiệp để đưa ra hai lựa chọn điều trị, một cách là xử lý dứt điểm, một cách điều trị thận trọng.Trợ lý đưa hai chương trình vào hệ thống xử lý mô phỏng để kiểm tra mô phỏng. Theo báo cáo thử nghiệm, bạn bè và đồng nghiệp của Minh nhanh chóng xác định rằng việc sử dụng các giải pháp triệt để là tối ưu.

Quá trình này chỉ mất vài giờ kể từ khi có được hình ảnh để xác định kế hoạch điều trị. Ngay cả một vài năm trước, thật khó để tưởng tượng. Người bạn của Minh nói rằng, trước khi được giới thiệu hệ thống y tế thông minh, tình trạng phức tạp như vậy thường đòi hỏi nhiều ngày tham vấn, điều tra nhiều yếu tố không chắc chắn và đôi khi thậm chí trì hoãn cơ hội điều trị tốt nhất. Công nghệ thông minh đã cải thiện đáng kể hiệu quả điều trị.

Sau giờ làm việc: Mua sắm tiện lợi



Công việc được hoàn thành suôn sẻ và Minh rất hạnh phúc. Sau khi làm việc, Minh quyết định mua một chiếc áo mới cho mình. Vì vậy, anh đã lái xe đến một cửa hàng quần áo thương hiệu lớn.

Giống như nhiều cửa hàng trong thành phố, cửa hàng này đã trở thành cửa hàng thông minh vài năm trước.Tại thời điểm Minh bước vào cửa, hệ thống chào đón của cửa hàng đã nhận ra anh và cho thấy sự chào đón

của anh trên màn hình. Một chiếc gương thông minh đứng cạnh mỗi hàng quần áo trong cửa hàng, khi Minh lấy quần áo và đi đến chiếc gương, tấm gương cho thấy hình ảnh ba chiều của anh ấy khi anh ấy mặc quần áo mới. Nhờ vào sự đột phá về phát hiện dáng điệu và công nghệ tái tạo con người 3D, những hình ảnh được tổng hợp bởi gương thông minh rất thực tế và có thể dễ dàng phù hợp với các tư thế khác nhau của Minh, không khác gì so với trải nghiệm thử thực tế.

Minh rất hài lòng với bộ quần áo mới của mình. Sau khi rời khỏi cửa hàng, anh ấy đã gửi hình ảnh của bộ đồ cho cha mình. Trên đường về nhà, Minh nhìn thấy câu trả lời của cha mình trên kính chắn gió của xe:

"Con có mắt thẩm mỹ đấy, con sắp bắt kịp với bố rồi."

Suy nghĩ và thảo luận

Sau khi đọc câu chuyện của Minh, bạn có ấn tượng hơn về trí tuệ nhân tạo không? Hãy nói lên tưởng tượng của bạn về trí tuệ nhân tạo trong tương lai.

Cuộc sống của Minh sau mười năm không còn xa vời đối với chúng ta. Điều đó được điều khiển bởi làn sóng trí tuệ nhân tạo và đang được thực hiện từng bước. Để tạo ra một cuộc sống thông minh mới, chúng ta hãy tìm hiểu về trí tuệ nhân tạo.

1.2. Ngắn gọn về lịch sử trí tuệ nhân tạo

Sự ra đời

Ngay từ những năm 1940 và 1950, các nhà toán học và kỹ sư máy tính đã bắt đầu khám phá khả năng sử dụng trí thông minh mô phỏng máy.

Năm 1950, Alan Turing đề xuất bài kiểm tra Turing nổi tiếng trong bài báo "Máy tính và trí thông minh" của ông. Trong bài kiểm tra Turing, một người thử nghiệm sẽ nói chuyện tự do với một cỗ máy và một người trong phòng bí mật. Nếu người kiểm tra không thể phân biệt giữa hai thực thể đang nói chuyện đâu là người đâu là máy, cỗ máy tham gia vào cuộc trò chuyện được xem là vượt qua bài kiểm tra. Bài kiểm tra Turing đã được công nhận rộng rãi như một tiêu chí quan trọng để kiểm tra trí thông minh máy trong vài

thập kỷ qua và đã có tác động sâu sắc đến sự phát triển trí thông minh nhân tạo.

Hình 1-1: Alan Turing (1912-1954)

Vào mùa hè năm 1951, Marvin Minsky, một sinh viên 24 tuổi tốt nghiệp tại Khoa Toán học Đại học Princeton, đã thiết lập máy mạng nơron đầu tiên trên thế giới có tên là SMARC (Stochastic Neural Analog Reinforcement Calculator). Mạng lưới nhỏ này gồm 40 tế bào thần kinh. Đây là lần đầu tiên người ta mô phỏng sự truyền tín hiệu thần kinh. Công việc đột phá này đã đặt nền tảng sâu rộng cho trí thông minh nhân tạo. Do những đóng góp đột phá của ông trong lĩnh vực trí tuệ nhân tạo, Marvin Minsky giành giải Turing trong lĩnh vực khoa máy học tính vào năm 1969.

Hình 1-2: Marvin Minsky

Năm 1955, Aller Newell, Ilerlycert Sinion, và Clift Shaw đã thiết lập một chương trình máy tính có tên là Logic Theorist để mô phỏng các kỹ năng giải quyết vấn đề con người. Chương trình này đã chứng minh thành công 38 trong số 52 sách giáo khoa toán học trong sách giáo khoa toán học đại học, và thậm chí còn tìm thấy bằng chứng tốt hơn sách giáo khoa. Công việc này đi tiên phong trong một phương pháp được sử dụng rộng rãi trong tương lai: reasoning.

Năm 1956, Wensky, John McCarthy, Claude Shannon và Nathan Rochester đã tổ chức một hội thảo tại trường Dartmouth College ở Hoa Kỳ. Cuộc họp đề xuất:

“Mọi khía cạnh của việc học và sự thông minh đều có thể được mô tả chính xác để mọi người có thể xây dựng một chiếc máy để mô phỏng nó.”

Hội nghị đã thiết lập tên "Trí tuệ nhân tạo" (AI), một lĩnh vực mới dành riêng cho việc mô phỏng trí thông minh của con người thông qua các máy móc, chính thức tuyên bố sự ra đời của trí tuệ nhân tạo.

Hình 1-3: Hội thảo tại Dartmouth, nơi sinh ra khái niệm Trí tuệ nhân tạo

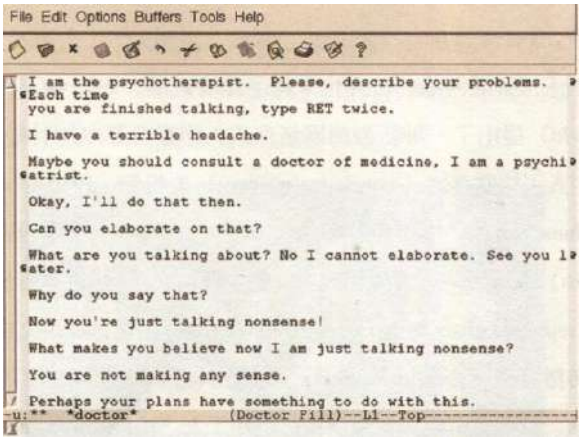
Làn sóng đầu tiên (1956-1974): Chuyến đi đầu tiên tuyệt vời

Sự ra đời của trí tuệ nhân tạo gây sự kinh ngạc cho thế giới, và lần đầu tiên mọi người thấy khả năng trí tuệ được tạo ra bởi máy móc. Vào thời điểm đó, một số người lạc quan rằng một cỗ máy hoàn toàn thông minh sẽ được sinh ra trong vòng 20 năm. Mặc dù chúng ta vẫn chưa thấy một cỗ máy như vậy cho đến bây giờ, sự nhiệt huyết xuất phát từ sự ra đời của nó đã truyền sức sống vô hạn vào sự phát triển của lĩnh vực mới này.

Năm 1963, Văn phòng Kế hoạch và Nghiên cứu tiên tiến mới được thành lập của Mỹ (ARPA) đã đầu tư 2 triệu USD vào MIT để mở một dự án

mới, Project MAC (The Project on Mathematics and Computation). Ngay sau đó, các nhà khoa học thông minh nhân tạo nổi tiếng nhất, Marvin Minsky và McCarthy, đã tham gia dự án và quảng bá một loạt các nghiên cứu trong các lĩnh vực hiểu biết về ngôn ngữ và thị giác. Dự án MAC đã đào tạo một số lượng lớn các tài năng khoa máy học tính và trí tuệ nhân tạo sớm nhất, đã có tác động sâu sắc đến sự phát triển của các lĩnh vực này. Dự án này cũng là tiền thân của Phòng thí nghiệm Trí tuệ Nhân tạo và Trí tuệ Nhân tạo MIT (MIT CSAIL).

Được thúc đẩy bởi sự nhiệt tình và đầu tư lớn, một loạt những thành tựu mới đã xuất hiện vào thời điểm này. Giáo sư Joseph Weizenbaum thuộc Viện Công nghệ Massachusetts đã thiết lập chương trình đối thoại tự nhiên đầu tiên trên thế giới ELIZA từ 1964 đến 1966. ELIZA trò chuyện với mọi người thông qua các quy tắc khớp và đối thoại mẫu đơn giản. Mặc dù quá trình đối thoại này hơi đơn giản so với quan điểm ngày nay, khi nó lần đầu xuất hiện trước mặt thế giới, nó thực sự gây ngạc nhiên cho thế giới. Đại học Waseda ở Nhật Bản cũng đã phát minh ra robot hình người đầu tiên trên thế giới từ năm 1967 đến năm 1972. Nó không chỉ nói chuyện, mà còn có thể đi và lấy các vật thể trong nhà dưới sự hướng dẫn của hệ thống thị giác.



Kỳ vọng càng cao, sự thất vọng càng lớn. Mặc dù lĩnh vực trí thông minh nhân tạo đã xuất hiện nhưng vẫn khó đáp ứng được những kỳ vọng không thực tế của xã hội trong lĩnh vực này. Từ những năm 1970, ngày càng có nhiều lời chỉ trích về trí tuệ nhân tạo. Trong lĩnh vực này, các vấn đề khác nhau đằng sau đó dần được tiết lộ. Một mặt, có một mâu thuẫn sắc nét giữa sức mạnh tính toán hạn chế và nhu cầu tính toán ngày càng tăng nhanh, mặt khác, sự mơ hồ và thay đổi rất lớn trong hiểu biết ngôn ngữ trực quan tạo thành một trở ngại không thể vượt qua. Trí thông minh nhân tạo bước vào mùa đông đầu tiên vào giữa những năm 1970 khi sự nhiệt tình của công chúng giảm xuống và đầu tư giảm mạnh.

Làn sóng thứ hai (1980-1987): Sự phát triển của hệ thống chuyên gia

Trong những năm 1980, làn sóng trí thông minh nhân tạo tái xuất hiện nhờ những tiến bộ mới trong công nghệ như hệ thống chuyên gia và mạng thần kinh nhân tạo.

Hệ thống chuyên gia là một hệ thống chương trình trả lời các câu hỏi cụ thể về những lĩnh vực nhất định dựa trên một bộ quy tắc cụ thể. Ngay từ những năm 1960, Edward Feigenbaum đã bắt đầu một nghiên cứu về các hệ thống chuyên gia. Do đó, ông được gọi là "cha đẻ của hệ thống chuyên gia". Trong những năm 1970, các nhà khoa học tại Đại học Stanford đã phát triển một hệ thống gọi là MYCIN chẩn đoán nhiễm trùng máu dựa trên 600 quy tắc viết tay.

Năm 1980, Đại học Carnegie Mellon đã phát triển một hệ thống chuyên gia gọi là XCON cho Digito Corporation (DEC) giúp công ty này tự động chọn một sự kết hợp các thành phần máy tính dựa trên nhu cầu của khách hàng. Hệ thống này đã tiết kiệm được 100 triệu đô la mỗi năm cho Digito. Giá trị thương mại khổng lồ của XCON đã thúc đẩy sự phát triển của ngành công nghiệp cho trí tuệ nhân tạo, đặc biệt là các hệ thống chuyên gia.

Điều đáng nói đến là sự thành công của hệ thống chuyên gia đã từng bước thay đổi hướng phát triển trí tuệ nhân tạo. Các nhà khoa học đang bắt đầu tập trung vào giải quyết các vấn đề trong thế giới thực trong các lĩnh vực cụ thể thông qua các hệ thống thông minh, mặc dù điều này không chính xác giống như ý định ban đầu của họ để xây dựng trí thông minh chung.

Song song với đó, nghiên cứu về mạng nơron nhân tạo cũng đã đạt được những tiến bộ quan trọng.Năm 1982, John Hopfield đề xuất một dạng mạng mới, mạng Hopfield, trong đó cơ chế bộ nhớ kết hợp được giới thiệu. Năm 1986, David Rumelhart, Geoffrey Hinton và Ronald Williams cùng nhau đánh dấu một cột mốc bằng bài báo: "Leaming representations by back-propagating errors". Trong bài báo này, họ cho thấy sự truyền ngược

(backpropagation) có thể học cách diễn đạt dữ liệu đầu vào trong lớp ẩn của mạng thần kinh một cách hiệu quả thông qua các thí nghiệm. Kể từ đó, các thuật toán backpropagation đã được sử dụng rộng rãi trong việc đào tạo mạng nơron nhân tạo.

Hình 1-4: Geoffrey Hinton (1947 -)

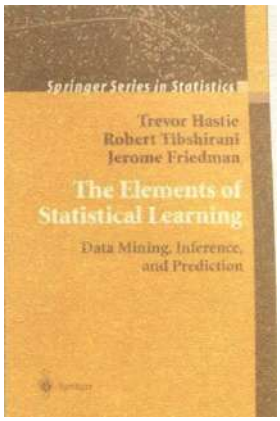
Sự trỗi dậy của một làn sóng mới của trí tuệ nhân tạo, các thương mại quốc tế của Nhật Bản và Công nghiệp vào năm 1982 bắt đầu dự án nghiên cứu quy mô lớn đầy tham vọng nhằm xây dựng một "máy tính thế hệ thứ năm". Mục tiêu của chương trình này là đạt được hiệu suất giống như siêu máy tính thông qua tính toán song song quy mô lớn và cung cấp một nền tảng cho sự phát triển trí thông minh nhân tạo trong tương lai. Đáng tiếc, sau 10 năm nghiên cứu và phát triển, nó tốn 50 tỷ yên mà dự án vẫn không đạt được mục tiêu mong đợi.

Vào cuối những năm 1980, đầu tư khổng lồ của ngành vào hệ thống chuyên gia và sự kỳ vọng cao đã bắt đầu cho thấy những tác động tiêu cực. Sự phát triển và bảo trì của hệ thống như vậy rất tốn kém và có giá trị thương mại hạn chế. Dưới ảnh hưởng của sự thất vọng, đầu tư vào trí tuệ nhân tạo đã được giảm đi rất nhiều, và sự phát triển của trí thông minh nhân tạo đã một lần nữa bước vào mùa đông.

Làn sóng thứ ba (2011-nay): Sự tái tạo rực rỡ

Vào thời điểm những năm 1990, trí thông minh nhân tạo đã trải qua nhiều thăng trầm. Mặc dù tham vọng của thế giới đã bị thất bại, lĩnh vực này ngày càng trở nên khó khăn hơn. Các nhà khoa học đã đặt ra những mục tiêu

không thực tế và bắt đầu tập trung vào việc phát triển các công nghệ thông minh để giải quyết các vấn đề cụ thể.



Trong thời gian này, các học giả nghiên cứu trí thông minh nhân tạo bắt đầu giới thiệu các công cụ toán học trong các ngành khác nhau, như đại số nâng cao, thống kê xác suất và lý thuyết tối ưu hóa, tạo ra nền tảng toán học vững chắc hơn cho trí tuệ nhân tạo. Việc sử dụng rộng rãi ngôn ngữ toán học đã mở ra các kênh truyền thông và hợp tác giữa trí thông minh nhân tạo và các ngành khác, cho phép các kết quả được kiểm tra chặt chẽ hơn. Được điều khiển bởi toán học, một số lượng lớn các mô hình toán học mới và các thuật toán đã được phát triển, chẳng hạn như statistical learning theory, support vector machine và mô hình đồ họa xác suất probabilistic graphical model. Các thuật toán thông minh mới được phát triển đang dần được áp dụng để giải quyết các vấn đề thực tế như giám sát an ninh, nhận dạng giọng nói, tìm kiếm trên web, đề xuất mua sắm và giao dịch tự động.

Việc áp dụng thành công thuật toán mới trong các tình huống cụ thể đã giúp các nhà khoa học nhìn thấy bình minh của trí tuệ nhân tạo.

Trong thế kỷ 21, sự gia tăng toàn cầu hóa và Internet đang bùng nổ đã mang lại sự tăng trưởng bùng nổ của dữ liệu điện tử trên toàn thế giới. Nhân loại đã bước vào kỷ nguyên của "dữ liệu lớn". Đồng thời, sức mạnh tính toán của chip máy tính tiếp tục tăng với tốc độ cao. Sức mạnh tính toán của một bộ vi xử lý đồ họa NVIDIA Tesla V100 hiện tại đã vượt quá 10 nghìn tỷ điểm hoạt động trên một giây, vượt qua siêu máy tính nhanh nhất thế giới vào năm 2001.

Với sự hỗ trợ của sự tăng trưởng theo hàm mũ về dữ liệu và sức mạnh tính toán, các thuật toán thông minh nhân tạo cũng đã có những đột phá lớn. Trong cuộc thi thuật toán nhận dạng hình ảnh toàn cầu vào năm 2012, IISVRC (còn được gọi là cuộc thi Image Net), một mạng nơron nhiều lớp được phát triển bởi Đại học Toronto, Alex Net, đã giành chức vô địch và vượt qua vị trí thứ hai bằng cách sử dụng các thuật toán máy học truyền thống. Kết quả của cuộc thi này gây ra những cú sốc lan rộng trong lĩnh vực trí tuệ nhân tạo. Kể từ đó, việc học sâu dựa trên mạng nơron nhiều lớp đã được mở rộng tới nhiều lĩnh vực ứng dụng và đã thành công trong nhiều lĩnh vực như nhận dạng giọng nói, phân tích hình ảnh và video. Trong năm 2016, chương trình AlphaGo của Google, được đào tạo về học tập sâu, đánh bại cựu vô địch thế giới. Phiên bản cải tiến của nó đã đánh bại cầu thủ cờ vua số một thế giới của Trung Quốc Ke Jie vào năm 2017.

Những thành tựu gây sốc này một lần nữa đã kích thích sự quan tâm của thế giới đối với trí thông minh nhân tạo. Chính phủ và các tổ chức thương mại trên khắp thế giới đã liệt kê trí thông minh nhân tạo là một phần quan trọng trong chiến lược phát triển tương lai của họ. Kết quả là, làn sóng thứ ba đã mở ra sự phát triển mạnh mẽ của trí tuệ nhân tạo.

1.3. Trí thông minh nhân tạo trong mọi ứng dụng cuộc sống

Trong những năm gần đây, công nghệ trí tuệ nhân tạo đã được sử dụng rộng rãi trong các ngành công nghiệp khác nhau và đã tạo động lực mới cho sự phát triển ~~và nâng cấp của họ~~. Dưới đây là một vài ví dụ quan trọng.

Bảo mật

Cùng với quá trình đô thị hoá và sự phát triển nhanh chóng của nền kinh tế xã hội, an ninh đã dần dần trở thành chủ đề quan tâm chung cho toàn xã hội. Từ việc xây dựng các thành phố an toàn đến bảo vệ cộng đồng cư dân, từ việc giám sát các nơi công cộng đến việc bảo vệ các thiết bị điện tử cá nhân, chúng ta không thể tách rời khỏi một hệ thống an ninh hiệu quả và đáng tin cậy. Trong những năm gần đây, công nghệ trí tuệ nhân tạo đã được sử dụng rộng rãi trong lĩnh vực an ninh và đã trở thành vệ sĩ của tất cả chúng ta.

Từ năm 2015, nhiều thành phố trên cả nước đã đẩy nhanh việc xây dựng các thành phố an toàn và tích cực triển khai các hệ thống giám sát video công cộng đặc biệt tại các tuyến đường đô thị chính và khu vực trọng yếu. Khi đối mặt với lượng dữ liệu video giám sát lớn, cách truyền thống dựa vào cảnh sát giám sát hoặc tìm các phân đoạn quan trọng bằng cách xem video rõ ràng là không khả thi. Do đó, công nghệ phân tích video dựa trên trí tuệ nhân tạo được áp dụng rộng rãi. Công nghệ phân tích video thông minh mới có thể thực hiện được các công việc như sau:

- Người đi bộ và phương tiện được phát hiện từ video trong thời gian thực.
- Tự động tìm thấy hành vi bất thường trong video (chẳng hạn như người đi bộ say rượu hoặc xe ngược chiều) và nhanh chóng đưa ra cảnh báo với thông tin thời điểm, vị trí cụ thể trong video.
- Tự động đánh giá mật độ của đám đông và hướng của dòng người, khám phá những mối nguy hiểm tiềm ẩn gây ra bởi đám đông dày đặc, giúp nhân viên hướng dẫn và quản lý dòng người.

Những công nghệ này có thể giải phóng các nhà quản lý thành phố của chúng ta khỏi công việc giám sát video tẻ nhạt và phục vụ mọi người hiệu quả hơn.

Y tế

Một cơ thể khỏe mạnh đặc biệt quan trọng đối với mỗi người chúng ta. Khi có bệnh, chúng ta cần phải tìm sự giúp đỡ của bác sĩ kịp thời để chữa trị. Tuy nhiên, các bác sĩ có kinh nghiệm vẫn còn rất khan hiếm.

Việc áp dụng trí tuệ nhân tạo trong điều trị y tế cung cấp một cách suy nghĩ mới để giải quyết vấn đề "khó gặp bác sĩ". Hiện nay, nhiều tổ chức nghiên cứu trên khắp thế giới đã đầu tư rất nhiều nỗ lực trong việc phát triển công nghệ để phân tích tự động hình ảnh y tế. Những kỹ thuật này tự động tìm các điểm chính trong hình ảnh y tế và thực hiện phân tích so sánh. Kết quả phân tích của trí thông minh nhân tạo có thể cung cấp thông tin tham khảo cho chẩn đoán của bác sĩ làm giảm chẩn đoán sai. Bên cạnh đó, một số công nghệ mới cũng thực hiện tái tạo ảnh y tế bởi một mô hình ba chiều các bộ phận cơ thể con người, giúp phẫu thuật các bác sĩ phẫu thuật chính xác hơn.

Với những tiến bộ trong công nghệ chuẩn đoán y tế, chúng tôi tin rằng trí tuệ nhân tạo sẽ không chỉ cung cấp khuyến nghị cho cho các bác sĩ, mà còn có thể cung cấp cảnh báo sớm về nguy cơ mắc bệnh và tư vấn sức khỏe cho mỗi người chúng ta, để chúng ta sống khỏe mạnh hơn.

Dịch vụ khách hàng thông minh

Với sự phát triển của Internet và thương mại điện tử, việc giao tiếp của chúng ta với các người bán hàng đã trở nên đa dạng và trực tiếp hơn. Ví dụ: Nếu chúng ta xem sản phẩm hoặc dịch vụ, chúng ta có thể tham khảo trực tiếp người bán bằng điện thoại hoặc công cụ trò chuyện trực tuyến. Làm thế nào để thực hiện hiệu quả việc giao tiếp thường xuyên từ khách hàng mang lại những thách thức lớn cho doanh nghiệp, đó cũng là một phần quan trọng trong việc duy trì tính cạnh tranh trong kỷ nguyên Internet.

Để đối phó với thách thức mới này, nhiều công ty đã bắt đầu giới thiệu công nghệ trí tuệ nhân tạo để tạo ra một hệ thống dịch vụ khách hàng thông minh. Dịch vụ khách hàng thông minh có thể giao tiếp với khách hàng như một nhân viên. Nó có thể hiểu được các vấn đề của khách hàng, phân tích ý nghĩa của vấn đề (chẳng hạn như khách hàng có hỏi giá hay tham khảo chức năng của sản phẩm), sau đó tiến hành trả lời chính xác và cá nhân hóa để nâng cao trải nghiệm của khách hàng. Đối với các doanh nghiệp, hệ thống như vậy không chỉ có thể nâng cao hiệu quả phục vụ khách hàng mà còn tự động phân tích nhu cầu và vấn đề của khách hàng để cung cấp cơ sở cho các quyết định trong tương lai.

Hiện nay, dịch vụ khách hàng thông minh đã được áp dụng trong nhiều ngành công nghiệp, bao gồm thương mại điện tử, tài chính, truyền thôngvà du lịch.

Lái xe tự động

Trong xã hội hiện đại, lái xe ô tô đi làm hoặc đi du lịch đã trở thành một hoạt động phổ biến. Với sự phát triển của công nghệ, mọi người bắt đầu khám phá khả năng cho phép xe tự động lái.

- Năm 2004, Cơ quan Chỉ đạo các Dự án Nghiên cứu Quốc phòng Tiên tiến (DARPA) tổ chức cuộc thi Xe không người lái ở sa mạc Mojave. Tại thời điểm đó, không đội nào trong 15 đội hoàn thành mục tiêu đi 142 dặm với điều kiện không có người lái. Tuy nhiên, các đề xuất đưa ra bởi các đội thi tại thời điểm đó đã trở thành nguyên mẫu của một chiếc xe tự lái hiện đại.
- Năm 2010, Google thông báo rằng họ đang phát triển xe tự lái, và một năm sau đó chiếc xe được thử nghiệm ở sa mạc Mojave. Đến năm 2012, Google thông báo rằng xe tự lái của nó đã đi 300.000 dặm mà không gây ra bất cứ tai nạn nào.
- Trong năm 2014, Baidu và BMW đã công bố bắt đầu nghiên cứu lái xe tự động.

Cho đến nay, bức màn nghiên cứu xe tự lái đã được mở ra và nhiều công ty đã đầu tư vào phát triển công nghệ xe tự lái.

Những chiếc xe tự lái ngày nay sử dụng nhận thức thời gian thực về môi trường lái xe qua nhiều cảm biến, bao gồm máy quay video, radar laser, hệ thống định vị vệ tinh. Hệ thống lái thông minh có thể phân tích nhiều tín hiệu cảm biến, kết hợp bản đồ và cảnh báo (như đèn giao thông và biển báo), lập phương án dẫn đường trong thời gian thực và ra lệnh điều khiển hoạt động của xe.

Sản xuất công nghiệp

Với nhu cầu mạnh mẽ của người tiêu dùng, hệ thống sản xuất công nghiệp phải trở nên thông minh hơn để gia tăng hiệu quả và năng xuất, và trí tuệ nhân tạo là động lực mạnh nhất cho các hệ thống sản xuất công nghiệp.

Ví dụ, kiểm soát chất lượng là một phần quan trọng của quá trình sản xuất, nếu một sản phẩm chất lượng kém được đưa vào thị trường, nó sẽ không chỉ làm giảm đáng kể trải nghiệm người dùng mà còn dẫn đến các tai nạn nghiêm trọng. Do đó, một số lượng lớn công nhân kiểm định được bố trí trên dây chuyền sản xuất thông thường để kiểm tra chất lượng bằng mắt thường. Loại phương pháp phát hiện thủ công này dễ dàng bỏ lỡ và đánh giá sai, hiệu quả sử dụng lao động thấp,. Do đó, nhiều công ty phát triển các công cụ kiểm tra chất lượng sản phẩm sử dụng trí thông minh nhân tạo để giúp các nhà máy tự động phát hiện các sai sót.

Tại Hannover Messe 2011, Đức đề xuất khái niệm về ngành công nghiệp 4.0, quan trọng nhất trong số đó là sử dụng một số lượng lớn các cảm biến trong môi trường công nghiệp để thu thập một lượng lớn dữ liệu. Trí thông minh nhân tạo là vũ khí mạnh mẽ để phân tích những dữ liệu khổng lồ

này và khai thác thông tin giá trị từ nó. Các ông lớn trong ngành công nghiệp như Siemens và General Electric (GE) đã phát triển hệ thống trí thông minh nhân tạo để dự đoán rủi ro sản xuất, giảm lãng phí nguyên liệu và năng lượng, đồng thời tăng hiệu quả sản xuất.

1.4. Trí tuệ nhân tạo và máy học

Trí tuệ nhân tạo là gì

Trong giới khoa học có nhiều định nghĩa khác nhau về trí tuệ nhân tạo. Chúng ta chấp nhận một số định nghĩa được thừa nhận rộng rãi:

Trí tuệ nhân tạo là một kỹ thuật mô phỏng khả năng nhận thức của con người thông qua máy móc.

Trí thông minh nhân tạo bao gồm một loạt các khả năng, bao gồm nhận thức, học tập, lý luận và ra quyết định. Từ quan điểm thực tế, năng lực cốt lõi của trí tuệ nhân tạo là đưa ra các đánh giá hoặc dự đoán dựa trên các yếu tố đầu vào nhất định. Ví dụ:

- *Trong ứng dụng nhận diện khuôn mặt, nó dựa trên ảnh đầu vào để xác định đâu là người trong ảnh.*
- *Trong nhận dạng giọng nói, nó có thể đánh giá nội dung của lời nói dựa trên tín hiệu âm thanh của người nói.*
- *Trong chẩn đoán y tế, nó có thể xác định nguyên nhân và bản chất của căn bệnh dựa trên hình ảnh y tế được nhập vào.*
- *Trong một trang web thương mại điện tử, nó có thể dự đoán sản phẩm mà người dùng quan tâm dựa trên lịch sử mua hàng trong quá khứ của người dùng, để trang web có thể đưa ra các đề xuất tương ứng.*
- *Trong các ứng dụng tài chính, nó có thể dự đoán sự thay đổi giá trong tương lai dựa trên giá cổ phiếu trong quá khứ và thông tin giao dịch.*
- *Trong trò chơi cờ vây, nó có thể dự đoán tỷ lệ thắng của một người chơi nhất định dựa trên tình hình hiện tại của các quân cờ.*

Suy nghĩ và thảo luận

Trong cuộc sống hàng ngày, bạn tiếp xúc với những loại đánh giá và dự đoán nào của trí tuệ nhân tạo?

Vậy thì trí thông minh nhân tạo tự động đưa ra những đánh giá hay dự đoán như thế nào? Trên thực tế, điều này không phải là bí ẩn, đôi khi chúng ta chỉ cần một số quy tắc đơn giản. Ví dụ, chúng ta có thể sử dụng một nhiệt kế phổ biến trong cuộc sống để tạo thành một hệ thống thông minh rất đơn giản. Đầu tiên đọc nhiệt độ làm giá trị đầu vào, sau đó sử dụng các quy tắc đơn giản như "Nhiệt độ có vượt quá 37,5 độ C không" để xác định xem người đó có bị sốt hay không.

Hệ thống chuyên gia nổi lên vào những năm 1980 dựa trên các quy tắc được xác định thủ công để trả lời các câu hỏi cụ thể. Tuy nhiên, việc quy tắc được xác định thủ công có nhiều hạn chế. Một mặt, việc thiết lập một hệ thống quy tắc hoàn chỉnh trong các ứng dụng phức tạp thường là một quá trình tốn kém thời gian và tiền bạc. Mặt khác, rất nhiều ứng dụng dựa trên dữ liệu tự nhiên như nhận diện hình ảnh hoặc giọng nói rất khó để định nghĩa cụ thể quy tắc bằng cách thủ công. Vì thế AI hiện đại thường có khả năng dự đoán hoặc đánh giá thông qua việc học. Một trong những phương pháp là máy học (machine learning) và phương pháp này dần trở thành phương pháp chủ đạo.

Học từ dữ liệu

Phương pháp máy học thường học các quy tắc có trong dữ liệu đã biết. Tuy nhiên, dữ liệu đã biết được sử dụng chủ yếu làm tài liệu cho việc học và mục đích chính của việc học là khái quát hóa (generalize), sau đó áp dụng các quy tắc đã học cho dữ liệu mới trong tương lai và đưa ra các đánh giá hoặc dự đoán.

Có nhiều cách máy học khác nhau. Một trong những hình thức máy học phổ biến nhất là **học có giám sát** (supervised learning). Hãy xem một ví dụ bên dưới. Trong trường hợp này, chúng ta hy vọng sẽ có được một công thức để dự đoán giá của một viên ngọc. Chúng ta biết rằng giá của đá quý chủ yếu được xác định bởi trọng lượng và cấp độ của nó. Nếu chúng ta sử dụng phương pháp học có giám sát, chúng ta cần phải thu thập các dữ liệu giá đá quý, như trong *Bảng 1-1*.

Trọng lượng	Loại	Giá
3	2	7030
4	1	6010
2	3	7960

Bảng 1-1. Giá của đá quý

Bây giờ chúng ta sẽ tìm hiểu một công thức có thể được sử dụng để dự báo giá theo Bảng 1-1. Mỗi hàng trong bảng được gọi là một mẫu (sample). Chúng ta có thể thấy rằng mỗi mẫu chứa hai phần: giá trị thực tế của thông tin đầu vào (trọng lượng, loại) và giá trị dự báo (giá). Sử dụng Bảng 1-1, chúng ta có thể thử nghiệm các công thức dự đoán khác nhau và nhận phản hồi để đánh giá tính chính xác của công thức đó bằng cách so sánh sự khác biệt giữa giá trị được dự đoán và giá thực trên mỗi mẫu. Thuật toán cho việc máy học sau đó liên tục điều chỉnh công thức dự đoán dựa trên những phản hồi này. Trong cách học tập này, giá trị thực của giá trị dự báo trong các mẫu đóng một vai trò giám sát trong quá trình học tập bằng cách cung cấp phản hồi. Chúng ta gọi đó là học tập được giám sát. Trong các ứng dụng thực tế, học tập được giám sát là một cách học tập rất hiệu quả. Các phương pháp cụ thể cho việc học có giám sát sẽ được giới thiệu trong các chương sau.

Học tập được giám sát yêu cầu mỗi mẫu phải được cung cấp giá trị thực của số tiền được dự đoán, điều này rất khó trong một số ứng dụng. Ví dụ, trong các ứng dụng chẩn đoán y tế, nếu bạn muốn có được một mô hình chẩn đoán thông qua việc học có giám sát, bạn cần yêu cầu một bác sĩ chuyên nghiệp ghi nhãn chính xác một số lượng lớn các trường hợp bệnh và dữ liệu hình ảnh y khoa của họ. Điều này đòi hỏi rất nhiều nhân lực và rất tốn kém. Để vượt qua những khó khăn như vậy, các nhà nghiên cứu đang tích cực khám phá các phương pháp khác mà không cần cung cấp thông tin giám sát (giá trị đích thực của số tiền dự đoán). Chúng tôi gọi đó là cách tiếp cận **học không giám sát** (unsupervised learning). Học tập không giám sát thường khó khăn hơn so với học được giám sát, nhưng nó là hướng nghiên cứu quan trọng cho sự phát triển trí thông minh nhân tạo bởi nó có thể giúp chúng ta vượt qua khó khăn trong việc thu thập dữ liệu giám sát trong nhiều ứng dụng thực tế.

Hướng nghiên cứu quan trọng

Một cách học khác được gọi là bán giám sát (semi-supervised) cũng nhận được chú ý rộng rãi. Nó yêu cầu một phần nhỏ của mẫu để cung cấp giá trị thực. Phương pháp này thường mang đến kết quả tốt hơn phương pháp không giám sát bằng cách sử dụng một cách hiệu quả thông tin được cung cấp trong khi kiểm soát chi phí thu thập thông tin giám sát ở mức chấp nhận được.

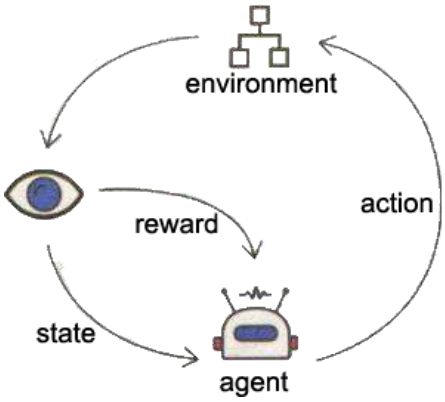
Học từ hành động

Trong ứng dụng thực tế của máy học, chúng ta cũng sẽ gặp phải một loại vấn đề khác: Ví dụ, khi chơi cờ vua, giao dịch cổ phiếu hoặc ra quyết định kinh doanh, chúng ta không quan tâm đến việc dự đoán có chính xác hay không, mà liệu quy trình hành động có mang lại lợi ích lớn nhất hay không. Để giải quyết những vấn đề như vậy, người ta đã đề xuất một phương pháp máy học khác gọi là học tăng cường (reinforcement learning).

Mục tiêu của việc học tăng cường là có được một chiến lược để hướng dẫn các hành động. Ví dụ, trong cờ vây, chiến lược này có thể hướng dẫn vị trí nên đánh dựa trên vị trí của các quân cờ khác; trong giao dịch chứng khoán, chiến lược này sẽ cho chúng ta biết thời điểm mua và khi nào bán. Không giống như học tập được giám sát, việc học tăng cường không đòi hỏi một loạt các mẫu chứa các đầu vào và dự đoán, nó học trong quá trình hành động.

Một mô hình học tập tăng cường thường bao gồm các phần sau:

- *Tập hợp các trạng thái (state). Ví dụ, vị trí các quân đen và trắng trên bàn cờ vây, giá của mỗi cổ phiếu trên thị trường.*
- *Tập hợp các hành động (action) có thể được chọn. Ví dụ, đối với cờ vây, đó là vị trí bạn có thể đánh, đối với giao dịch chứng khoán, đó là cổ phiếu và số lượng được mua hoặc bán tại mỗi thời điểm.*
- *Một tác tử (agent) có thể tương tác với môi trường (environment). Agent là chủ thể ra quyết định chọn một hành động nào đó trong tập hợp các hành động có thể, tác động tới môi trường làm thay đổi trạng thái của môi trường đồng thời thiết lập một trạng thái mới cho Agent kèo theo một phần thưởng (reward). Ví dụ, một đối thủ trong một trò chơi cờ vua, hoặc một người chơi chứng khoán. Trong học tập tăng cường, để giảm chi phí học tập, chúng ta thường sử dụng môi trường mô phỏng thay vì một môi trường thực tế.*
- *Quy tắc khen thưởng (reward). Khi agent ra một quyết định làm thay đổi môi trường đồng thời thiết lập trạng thái mới của nó thông qua hành động, nó nhận được một phần thưởng hoặc bị phạt (sự trả về tiêu cực).*



Học tăng cường thường bắt đầu với một chiến lược ban đầu. Thông thường, chiến lược ban đầu không nhất thiết là lý tưởng. Trong quá trình học tập, bộ phận ra quyết định tương tác với môi trường thông qua các hành động, liên tục nhận được phản hồi (phần thưởng hoặc hình phạt) và điều chỉnh chiến lược tối ưu hóa dựa trên phản hồi. Đây là một cách học rất mạnh mẽ, tốt hơn cơ chế ra quyết định của con người. Trong năm 2016, Học tăng cường đã được áp dụng thành công cho nhiều bài toán như Alpha dog đã đánh bại nhà vô địch thế giới Li Shiyi trong môn cờ vây, gây kinh ngạc trên toàn thế giới.

1.5. Tóm tắt chương

Trí thông minh nhân tạo là một môn học nghiên cứu cách mô phỏng khả năng nhận thức của con người bởi bất cứ hệ thống máy móc nhân tạo nào. Nó có thể được sử dụng để dự đoán và đưa ra quyết định bằng cách xác định thủ công, học tập từ dữ liệu hay học tập từ hành động. Thông qua những nỗ lực của vài thập kỷ qua, trí thông minh nhân tạo đã được phát triển và ứng dụng thành công trong nhiều ngành công nghiệp.

Làn sóng công nghệ mới, trí thông minh nhân tạo, đang biến đổi sâu sắc thế giới và ảnh hưởng đến cuộc sống của chúng ta, nhưng mới chỉ là khởi đầu. Sản xuất, cuộc sống, xã hội, giải trí và các khía cạnh khác vẫn có thể được cải thiện hơn nữa thông qua việc áp dụng công nghệ trí tuệ nhân tạo. Sự phát triển quá khứ của trí thông minh nhân tạo đã cho thấy một tương lai thú vị, kỷ nguyên mới và tốt hơn này đòi hỏi chúng ta phải làm việc cùng nhau để tạo ra chúng.

Chương II: Phân loại hoa

Minh đi chơi và đột nhiên bị thu hút đến một nơi, đó là một ngọn núi hoa. Anh ấy không muốn bước thêm một bước nào nữa và hỏi cha mình, "Hoa này là hoa gì?".

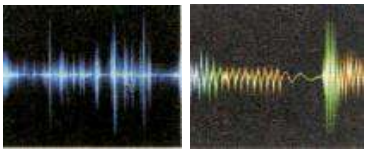
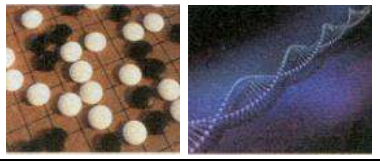
Khi quan sát cẩn thận, một vài cánh hoa khác so với những cánh hoa còn lại. "Đây có phải là hoa diên vĩ (iris) không?" "Đây là một loại hoa khác. Cánh hoa của chúng có kích thước khác. Con có thể vẽ chúng bằng tay và ghi lại chúng ...". Minh chạy đi: "Bố qua chỗ này và nhìn kìa!"

Khi chúng ta nhìn thấy một bức tranh, chúng ta có thể biết được có con vật nào trên bức tranh hay không, khi nghe một bài hát, chúng ta có thể phân biệt giữa nhạc cổ điển và nhạc pop, khi xem một video, chúng ta biết các diễn viên đang múa hay chạy nhảy ... Trong cuộc sống, việc đánh giá một loài vật, một quá trình như vậy được gọi là phân loại trong lĩnh vực trí tuệ nhân tạo.

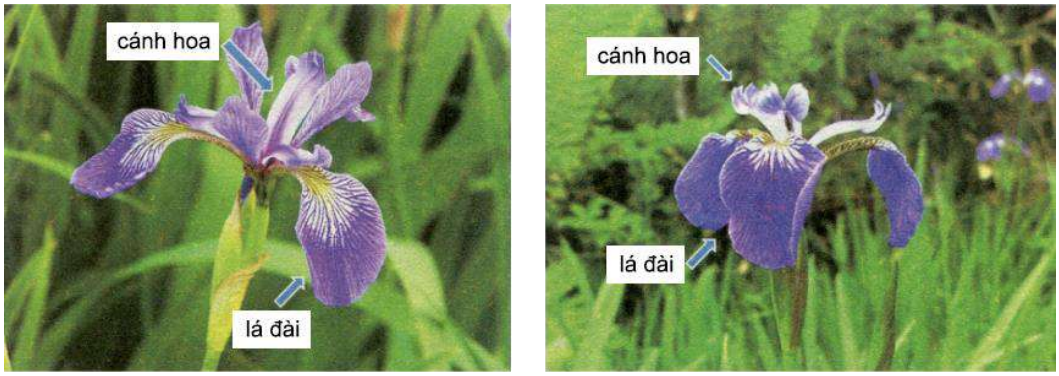
2.1. Nhiệm vụ phân loại

Hệ thống trí thông minh nhân tạo xử lý nhiều loại dữ liệu: hình ảnh, âm thanh, văn bản, video và hơn thế nữa. Các kiểu dữ liệu chung và một số ứng dụng liên quan đến chúng được thể hiện trong *Hình 2-1*. Dữ liệu chứa các thông tin. Phân loại là để đánh giá thể loại dựa trên các đặc điểm khác nhau của dữ liệu đã cho.

Trong chương này, chúng ta học về một nhiệm vụ phân loại đơn giản cụ thể là phân loại hai loại hoa diên vĩ (iris). Các cánh hoa của chúng sáng và đẹp, và các lá có màu xanh lá cây, làm cho mọi người cảm thấy dễ chịu. Có khoảng 300 loài trên thế giới, và những loài phổ biến là Iris versicolor và Iris setosa. Như trong Hình 2-2, chúng có cánh hoa và bẹ hoa có hình dạng và màu sắc tương tự nhau. Nói chung, một loại có cánh hoa lớn hơn, loại kia có kích thước cánh hoa nhỏ hơn. Chúng ta sử dụng ví dụ về phân loại hoa để hiểu cơ bản về vấn đề phân loại.

Kiểu dữ liệu	Hình ảnh minh họa	Ví dụ thực tế
Hình ảnh (Chi tiết ở chương 3)		Nhận diện khuôn mặt, nhận diện đối tượng, hiểu bối cảnh, xử lý ảnh, sinh ảnh…
Âm thanh (Chi tiết ở chương 4)		Nhận diện giọng nói, máy đối thoại, tự động biên soạn nhạc
Video (Chi tiết ở chương 5)		Phân loại video, hiểu nội dung, tóm tắt video, tự động tạo video, nhận diện hành động
Văn bản		Phân loại bài viết, hiểu nội dung, tự động viết bài báo
Các kiểu dữ liệu khác		Dữ liệu trò chơi sử dụng bởi AlphaGo, chuẩn đoán bệnh dựa trên dữ liệu gen

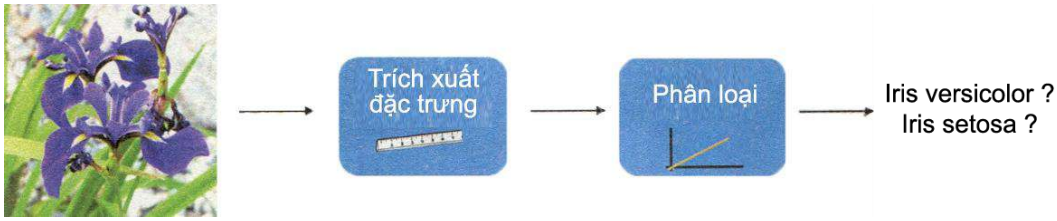
Hình 2-1: Các kiểu dữ liệu phổ biến và ứng dụng của chúng



Iris versicolor Iris setosa

Hình 2-2: Màu sắc của hai loại hoa

Chúng ta muốn xây dựng hệ thống trí tuệ nhân tạo đơn giản để phân biệt giữa 2 loại hoa giống như cách con người thực hi. Một hệ thống trí tuệ nhân tạo thực hiện các nhiệm vụ phân loại như vậy được gọi là một **bộ phân loại** (classifier). Hình 2-3 cho thấy luồng thực hiện của toàn bộ hệ thống. Khi iris được nhìn thấy lần đầu tiên, các đặc trưng của nó được trích xuất, sau đó được nhập vào bộ phân loại đã được huấn luyện, bộ phân loại có thể đưa ra các dự đoán loại của hoa. Trong phần tiếp theo, chúng ta hãy xây dựng hệ thống này từng bước một.



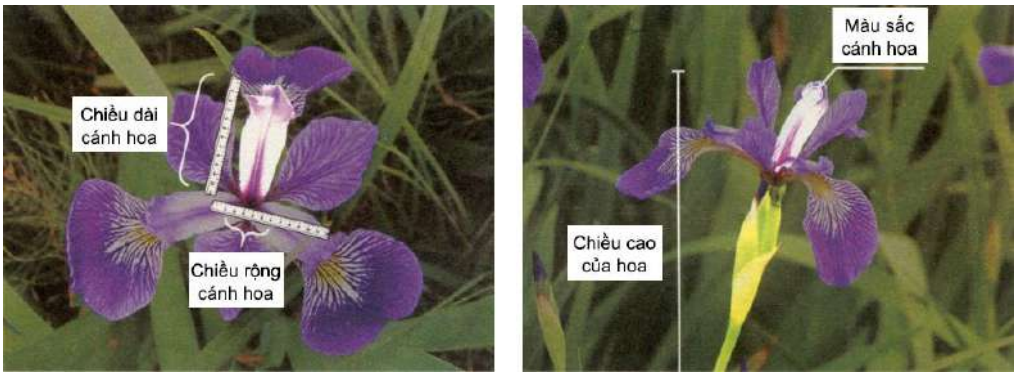
Hình 2-3: Hệ thống trí tuệ nhân tạo phân biệt hoa

2.2. Trích xuất các đặc trưng

Chúng ta có xu hướng phân biệt các đối tượng dựa trên một vài đặc điểm nhất định, ví dụ khi phân biệt các loại hoa khác nhau, ta dựa trên độ dài cánh hoa. Một đặc điểm mà có thể dùng để mô tả khía cạnh nhất định của đối tượng, giúp phân biệt đối tượng này với các đối tượng khác được gọi là một **đặc trưng** (feature).

Trong quá trình phân loại, chúng ta lấy ra những đặc trưng mà hệ thống AI có thể sử dụng được? Những nghiên cứu đã cho thấy chiều dài và chiều rộng cánh hoa là đặc trưng giúp việc phân loại trở nên hiệu quả. Khi

trích xuất các đặc trưng, như thể hiện trong hình bên trái của Hình 2-4, chúng ta có thể đo chúng trực tiếp bằng thước kẻ, việc lựa chọn các đặc trưng dựa trên kích thước của hoa như vậy cũng phù hợp với kinh nghiệm thực tế của những người tiếp xúc với hoa.



Hình 2-4: Trích xuất các đặc trưng khác nhau từ iris

Đặc trưng là khái niệm rất quan trọng trong quá trình phân loại của hệ thống AI. Như được mô tả trong hình bên phải của hình 2-4, chúng ta có thể sử dụng chiều cao của hoa hoặc màu sắc của cánh hoa như các đặc trưng. Tuy nhiên, trong các thời điểm khác nhau trong vòng đời, hoa có chiều cao khác nhau, đồng thời những loài hoa khác nhau lại có màu sắc tương tự nhau. Vì vậy, rất khó để phân biệt một cách hiệu quả nếu dựa trên chiều cao của cây hoa và màu sắc cánh hoa. Những đặc trưng khác nhau có tác động rất lớn đến sự chính xác trong quá trình phân loại.

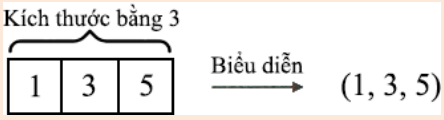
Vì vậy, chúng ta cần phải xem xét sự khác biệt giữa các loại đặc trưng khác nhau dựa trên đặc điểm của các đối tượng và dữ liệu của đối tượng đó. Đây không phải là một vấn đề đơn giản - nó thường đòi hỏi chúng ta phải thực sự hiểu được đặc tính của sự vật và sự khác biệt giữa các loại khác nhau. Chiều dài và chiều rộng của cánh hoa được sử dụng trong ví dụ trên là các đặc trưng tương đối đơn giản. Trong các phần sau, chúng ta sẽ được giới thiệu một số đặc trưng nhân tạo phổ biến của các loại dữ liệu khác nhau. Ví dụ với dữ liệu ảnh, đặc trưng được con người tạo ra là biểu đồ dốc định hướng (Histogram of oriented gradients - HOG); với âm thanh, người ta thiết kế ra hệ số công suất phổ tần âm thanh (Mel Frequency Cepstral Coefficients - MFCCs); với video, ta có biểu đồ luồng sáng (light flow histogram); với văn bản, ta có tần số từ - tần suất tài liệu nghịch đảo (term frequency–inverse document frequency - TFIDF)...

Vector đặc trưng (feature vector)

Từ đo đạc thực tế, chúng ta rút ra 2 đặc trưng của hoa là độ dài và độ rộng của cánh hoa. Để biểu diễn trong toán học, chúng ta dùng x1 để biểu thị độ dài, x2 để biểu thị độ rộng. Để sử dụng dễ dàng hơn, đặt 2 số cùng nhau trong ngoặc đơn (x1, x2). Một tập dữ liệu dưới dạng như vậy trong toán học được gọi là một vector.

Kiến thức bổ sung: Vector và các phép toán với vector

Trong toán học, vectơ là một chuỗi gồm nhiều số, chẳng hạn như (1, 3, 5). Số lượng các số được gọi là kích thước của vectơ. Ví dụ, kích thước của (1,3,5) là 3, hay còn được gọi là vector ba chiều.

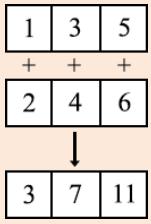


Chúng ta có thể thực hiện các phép toán đơn giản trên vector.

Phép cộng và phép trừ: Để cộng và trừ hai vector cùng kích thước, ta cộng hoặc trừ từng phần tử trong vector có vị trí tương ứng.

Ví dụ với phép cộng:

$$(1,3,5) + (2,4,6) = (1+2, 3+4, 5+6) = (3,7,11)$$



Phép trừ cũng tương tự như vậy.

$$(1,3,5) - (2,4,6) = (1-2, 3-4, 5-6) = (-1,-1,-1)$$

Phép nhân với một số: Khi nhân một số với một vector, ta nhân số đó với từng phần tử trong vector.

$$5 \times (1,3,5) = (5 \times 1, 5 \times 3, 5 \times 5) = (5,15,25)$$

Phép tích vô hướng: Hai vector cùng kích thước có thể thực hiện phép tích vô hướng. Khi thực hiện, các phần tử của hai vector được nhân với nhau, sau đó cộng lại.

$$(1,3) \cdot (2,4) = (1 \times 2) + (3 \times 4) = 14$$

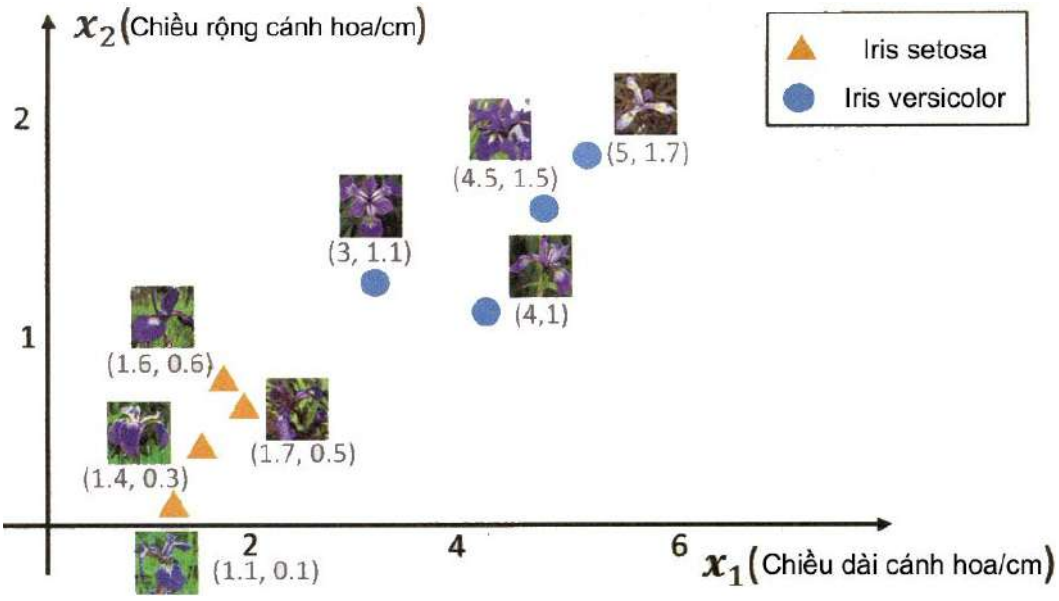
Với vectơ, chúng ta có thể tập hợp các giá trị đặc trưng mô tả một đối tượng lại với nhau để tạo thành một vector đặc trưng và mô tả nó đầy đủ hơn. Tổng quát, 1 vector có kích thước bằng n (độ dài n hay n chiều) có thể được

biểu diễn $x = (x_1, x_2, \ldots, x_n)$. Ví dụ, nếu chiều dài cánh hoa là 1.1 cm và chiều rộng là 0.1 cm thì đặc trưng của hoa được biểu diễn là $(1.1, 0.1)$.

Điểm đặc trưng và không gian đặc trưng

Ta có thể biểu diễn vector đặc trưng trên hệ tọa độ Đề-các. Ví dụ vector $(1.1, 0.1)$ có thể được biểu diễn như một điểm trên hệ tọa độ Đề các.

Như trong *Hình 2-5*, chúng ta vẽ vector đặc trưng của hoa lên hệ tọa độ. Một điểm trong hệ tọa độ đại diện cho vector đặc trưng của một bông hoa, các điểm biểu diễn vector đặc trưng được gọi là các điểm đặc trưng, không gian được hình thành bởi tất cả các điểm đặc trưng này được gọi là không gian đặc trưng.



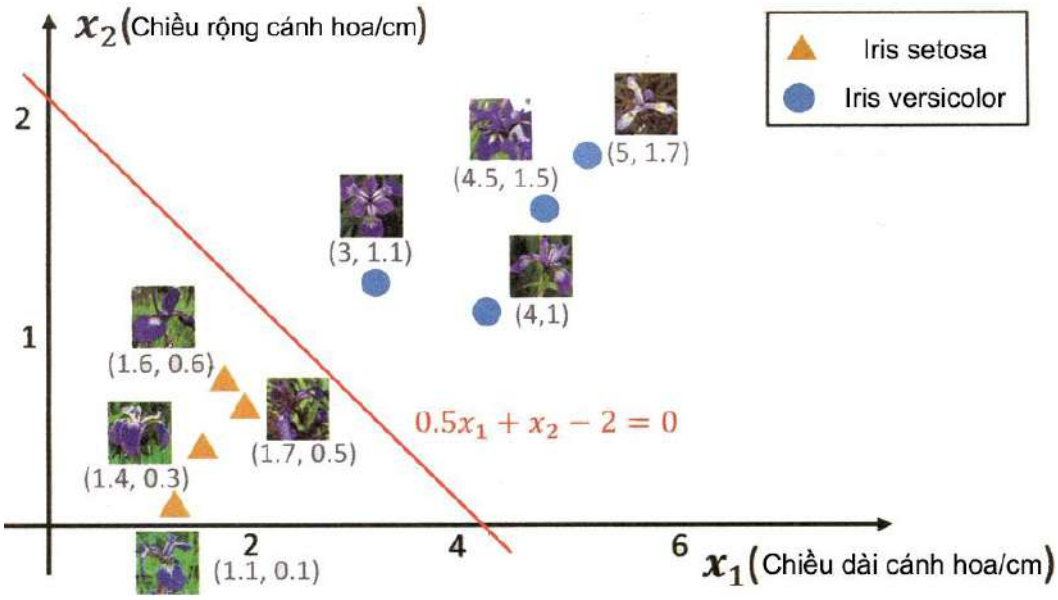
Hình 2-5: Biểu diễn các vector đặc trưng trên hệ tọa độ Đề-các

Trong không gian đặc trưng thể hiện trong Hình 2-5, khoảng cách giữa các điểm đặc trưng có thể được sử dụng để đo sự giống nhau giữa các bông hoa. Tổng quát, một không gian với kích thước (số chiều) bất kỳ đều có thể dùng khoảng cách giữa các điểm đặc trưng để đo lường độ giống nhau giữa các đối tượng. Ví dụ, trong không gian ba chiều, hai điểm được biểu diễn dưới dạng (x_1, x_2, x_3) và (z_1, z_2, z_3) có khoảng cách d được tính bởi công thức:

$$d = \sqrt{(x_1 - z_1)^2 + (x_2 - z_2)^2 + (x_3 - z_3)^2}$$

2.3. Bộ phân loại

Bộ phân loại là một hàm từ vector đặc tính đến lớp dự đoán. Trong bài toán phân loại hoa, chúng ta sử dụng +1 và -1 để biểu diễn tương ứng 2 loại hoa. Chúng ta cũng sử dụng ký tự y để biểu diễn loại hoa, y có thể nhận giá trị 1 hoặc -1. Những đặc trưng của hoa được trích xuất và biểu diễn thành vector đặc trưng, sau đó được vẽ trong không gian đặc trưng. Như được mô tả ở Hình 2-6, vấn đề phân loại hoa được chuyển thành vấn đề tách các điểm đặc trưng trong không gian đặc trưng. Nếu dùng 1 đường thẳng để tách đôi các điểm thì vấn đề trở thành: Có 2 loại điểm trên mặt phẳng tọa độ, vẽ 1 đường thẳng để tách 2 loại điểm đó.



Hình 2-6: Vẽ một đường thẳng để phân biệt hai loại hoa

Trong Hình 2-6, ta có thể dễ dàng vẽ một đường $0.5x_1 + x_2 - 2 = 0$, chia toàn bộ mặt phẳng tọa độ thành phần. Nếu điểm đặc trưng rơi vào phần phía trên bên phải của đường phân tách, output có giá trị +1 biểu diễn một loại hoa, còn nếu điểm đặc trưng rơi xuống phần dưới bên trái, output có giá trị -1 biểu diễn loại hoa còn lại. Bộ phân loại sử dụng quy tắc này có thể được biểu diễn bằng hàm sau:

$$g(x_1, x_2) = \begin{cases} +1, & 0.5x_1 + x_2 - 2 \geq 0 \\ -1, & 0.5x_1 + x_2 - 2 < 0 \end{cases}$$

Trong đó, $0.5x_1 + x_2 - 2$ được gọi là $f(x_1, x_2)$ tương ứng với phương trình đường thẳng trong *Hình 2-6*. Nếu $f(x_1, x_2) \geq 0$, điểm (x_1, x_2) thuộc phần phía trên bên phải, còn không nó thuộc phần phía dưới bên trái.

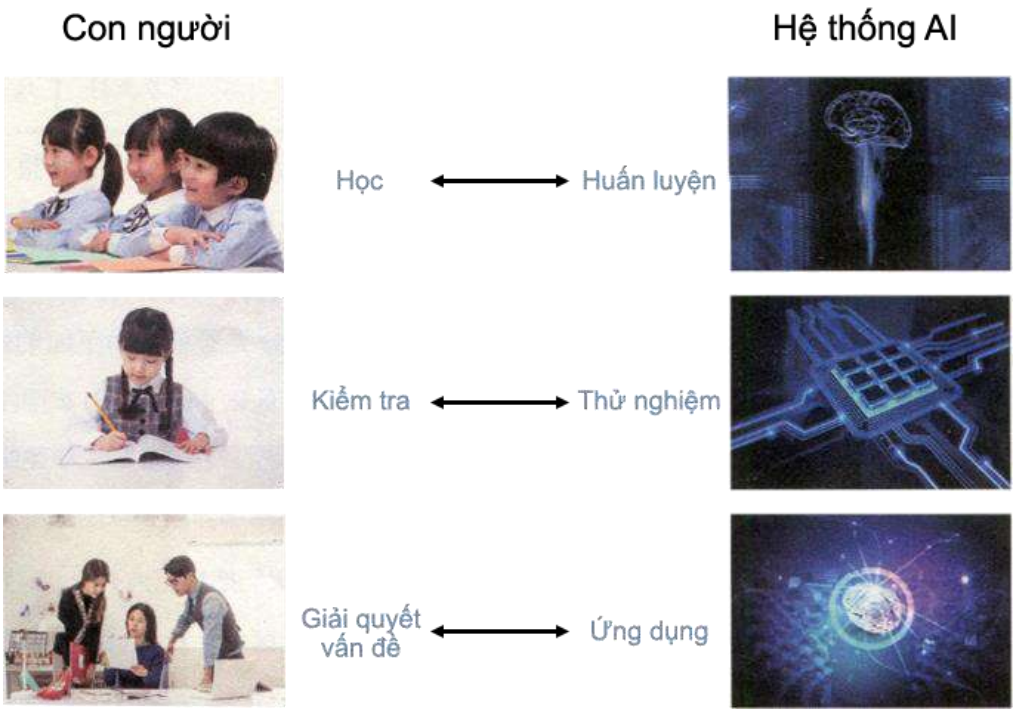
Hàm $f(x)$ là cốt lõi của $g(x)$. Mỗi hàm $f(x)$ khác nhau tương đương với việc vẽ một đường phân tách khác nhau trong *Hình 2-6*. Hàm $f(x)$ rất đa dạng, một bộ phân loại có dạng $f(x_1, x_2, \dots, x_n) = a_1 x_1 + a_2 x_2 + \cdots + a_n x_n + b$ được gọi là bộ phân loại tuyến tính (linear classifier), với n là kích thước của vector đặc trưng. $a_1, a_2, \dots, a_n, b$ là các hệ số của hàm và là tham số của bộ phân loại. Trong ví dụ trên, 0.5, 1, -2 là các tham số của bộ phân loại.

Hình 2-7: Mỗi bộ phân loại là một hàm từ vector đặc trưng tới loại dự đoán

Trong ví dụ đơn giản này, ta có thể trực tiếp vẽ 1 đường phân tách 2 loại điểm, nhưng trong thực tế sự phân bố vị trí của các loại phức tạp hơn nhiều, thường không thể và không hiệu quả khi muốn quan sát và thử vẽ đường phân tách một cách thủ công. Có một vài cách để bộ phân loại tự học cách vẽ đường phân tách đó.

Huấn luyện bộ phân loại

Ta có thể coi hệ thống AI tương tự con người. Mọi người cần học để tiếp thu kiến thức, làm bài kiểm tra để kiểm tra hiệu quả học tập và nắm vững các kỹ năng để giải quyết các vấn đề thực tế trong công việc. Hệ thống trí tuệ nhân tạo cũng tương tự như vậy: Quá trình học tập được gọi là huấn luyện (training), quá trình kiểm tra được gọi là thử nghiệm (testing), quá trình giải quyết vấn đề thực tế được gọi là ứng dụng (application).



Hình 2-8: Sự giống nhau giữa con người và hệ thống trí tuệ nhân tạo

Quá trình để bộ phân loại tìm được tham số thích hợp được gọi là quá trình huấn luyện bộ phân loại. Trong chương này, huấn luyện bộ phân loại là tìm ra đường phân tách tốt.

Chúng ta tiếp thu kiến thức thông qua giáo viên và sách vở, vậy hệ thống trí tuệ nhân tạo học thông qua gì? Câu trả lời là dữ liệu. Dữ liệu là một phần không thể thiếu của trí thông minh nhân tạo, và việc huấn luyện các hệ thống trí tuệ nhân tạo thường đòi hỏi một lượng lớn dữ liệu.

Dữ liệu được sử dụng trong giai đoạn huấn luyện được gọi là dữ liệu huấn luyện (training data), dữ liệu được sử dụng trong giai đoạn thử nghiệm được gọi là dữ liệu thử nghiệm (dữ liệu kiểm tra - testing data). Trong quá trình phân loại, dữ liệu dùng để huấn luyện và kiểm tra thường cần được phân loại từ trước (ghi nhãn trước). Việc đánh dấu thủ công dữ liệu được gọi là chú thích dữ liệu. Quá trình ghi nhãn dữ liệu tốn nhiều thời gian và tốn nhiều công sức, một số nhãn dữ liệu cũng có thể đòi hỏi chuyên môn trong các lĩnh vực liên quan (như đọc phim X Quang trong y tế). Việc thu thập và ghi nhãn dữ liệu là một quá trình rất quan trọng, chất lượng ghi nhãn dữ liệu sẽ ảnh hưởng trực tiếp đến hiệu suất của hệ thống trí tuệ nhân tạo sau khi được huấn luyện.

Dữ liệu hoa thu được bởi nhà thực vật học người Mỹ Edgar Anderson, ông đã đo chiều dài và chiều rộng của cánh hoa trong các bông hoa ở trang

trại ở Gaspésie, Canada. Ngoài ra, theo kiến thức của ông về thực vật học, ông đã dán nhãn trước loại của mỗi bông hoa.

Số thứ tự bông hoa	Chiều dài cánh hoa (cm)	Chiều rộng cánh hoa (cm)	Nhãn
1	1.1	0.1	Iris setosa
2	1.7	0.5	Iris setosa
3	1.4	0.3	Iris setosa
4	1.6	0.6	Iris setosa
5	5.0	1.7	Iris versicolor
6	4.0	1.0	Iris versicolor
7	4.5	1.5	Iris versicolor
8	3.0	1.1	Iris versicolor
…	…	…	…

Hình 2-9: Dữ liệu cánh hoa được thu thập và chú thích bởi Edgar Anderson

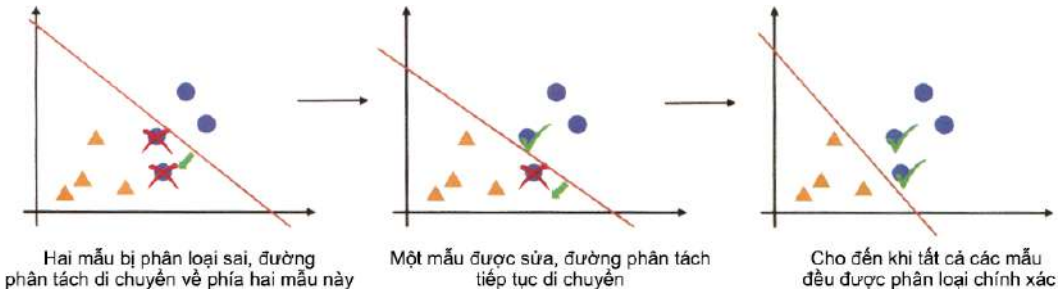
Sau khi được Edgar Anderson thu thập và chú thích dữ liệu, chúng ta đã có được tập dữ liệu được hiển thị trong Hình 2-9. Trong tập dữ liệu này, mỗi hàng đại diện cho một mẫu có chứa các đặc trưng và loại của một bông hoa. Với một tập dữ liệu như vậy, chúng ta có thể huấn luyện một bộ phân loại. Khi tập dữ liệu được sử dụng cho huấn luyện phân loại, chúng ta gọi đó là tập dữ liệu huấn luyện. Tiếp theo là quá trình huấn luyện bộ phân loại dựa trên tập dữ liệu. Quá trình này bao gồm một loạt các bước đánh giá và tính toán, thường được gọi là một thuật toán. Trên tập dữ liệu, có thể tạo ra các bộ phân loại khác nhau bằng các thuật toán khác nhau. Làm thế nào để thiết kế một thuật toán có hiệu suất cao (có độ chính xác phân loại cao) là một chủ đề nghiên cứu kinh điển trong máy học.

Hãy tiếp tục với ví dụ trên - tìm kiếm một bộ phân loại tuyến tính để phân loại hoa. Ở đây, hàm $f(x)$ trong bộ phân loại tuyến tính có thể được biểu diễn là $f(x_1, x_2) = a_1x_1 + a_2x_2 + b$. Mục tiêu là tìm các tham số thích hợp a_1, a_2, b sao cho bộ phân loại tương ứng có thể phân biệt hai loại hoa.

Dưới đây chúng tôi giới thiệu hai thuật toán phổ biến để huấn luyện các bộ phân loại tuyến tính: **Perceptron** và **Support Vector Machines** (SVM), cung cấp hai phương pháp để tự động tìm các tham số bằng cách sử dụng dữ liệu huấn luyện.

Perceptron

Perceptron là một thuật toán để huấn luyện các bộ phân loại tuyến tính, ý tưởng chính của nó là sử dụng dữ liệu huấn luyện bị phân loại sai để điều chỉnh các tham số của bộ phân loại hiện có, sau đó bộ phân loại đã được điều chỉnh sẽ có thể đánh giá chính xác hơn. Điều này được minh họa trong Hình 2-10 bằng một sơ đồ đơn giản: Đường phân tách đầu tiên phân loại sai 2 mẫu, sau đó đường phân tách dịch chuyển sang cạnh của mẫu bị phân loại sai. Sau lần điều chỉnh đầu tiên, một mẫu được sửa thành đúng, tuy nhiên vẫn còn một mẫu phân loại sai - khoảng cách của mẫu này đến đường phân tách giảm đi so với trước khi điều chỉnh. Tiếp theo, đường phân tách tiếp tục di chuyển về phía bên của mẫu phân loại sai cho đến khi đường phân loại đi qua mẫu được phân loại sai. Theo cách này, tất cả dữ liệu huấn luyện được phân loại chính xác.



Hình 2-10: Quá trình huấn luyện Perceptron

Thuật toán học perceptron điều chỉnh các tham số của đường phân tách theo mẫu bị phân loại sai, để đường phân tách dịch sang bên của dữ liệu bị phân loại sai để giảm khoảng cách giữa dữ liệu bị phân loại sai và đường phân tách cho đến khi được phân loại chính xác.

Thuật toán học perceptron cụ thể như sau.

Thuật toán Perceptron

Bước 1: Khởi tạo các tham số a_1, a_2, b.

Bước 2: Lựa chọn dữ liệu huấn luyện trong tập dữ liệu và duyệt từng mẫu dữ liệu. Nếu mẫu bị phân loại sai, tức là $y \times (a_1 x_1 + a_2 x_2 + b) \leq 0$ thì điều chỉnh tham số theo quy tắc:

$$a_1 \leftarrow a_1 + \eta y x_1$$

$$a_2 \leftarrow a_2 + \eta y x_2$$

$$b \leftarrow b + \eta y$$

Bước 3: quay lại bước 2, lặp lại cho đến khi không còn mẫu nào bị phân loại sai trong tập dữ liệu huấn luyện.

Trong đó, η là learning rate, biểu thị mức độ cập nhật các tham số.

Suy nghĩ và thảo luận

Tại sao thuật toán học perceptron tạo ra bộ phân loại tốt ?

Thuật toán học perceptron liên tục làm giảm trường hợp phân loại dữ liệu sai. Ở đây, học sinh có thể có hai câu hỏi: Thứ nhất, làm cách nào để đo được sự phân loại sai ? Thứ 2, sử dụng các phân loại sai như thế nào để điều chỉnh các tham số ? Để trả lời cho hai câu hỏi này, chúng ta sẽ cùng tìm hiểu về hàm mất mát và phương pháp tối ưu hóa.

Hàm mất mát (loss function) là biểu diễn toán học mức độ lỗi của việc phân loại trong quá trình huấn luyện. Mức độ sai lệch càng lớn, hàm loss có giá trị càng lớn. Xác định hàm loss thích hợp là một công đoạn quan trọng của việc huấn luyện. Perceptron và SVM được xây dựng trên các hàm loss khác nhau.

Trong ví dụ phân loại hoa, giả sử có N mẫu dữ liệu huấn luyện, gọi $({x_1}^{(i)}, {x_2}^{(i)})$ là vector đặc trưng của mẫu thứ i, ${y}^{(i)}$ là kết quả phân loại của mẫu thứ i thì hàm loss L được định nghĩa:

$$L(a_1, a_2, b) = \sum_{i=1}^{N} \max(0, -y^{(i)} \times (a_1 {x_1}^{(i)} + a_2 {x_2}^{(i)} + b))$$

Kiến thức bổ sung: Ký hiệu tổng xích ma Σ và maximum

Trong toán học, chúng ta sử dụng ký hiệu Σ để biểu diễn phép tính tổng một cách thuận tiện. Ví dụ, hãy viết một công thức để tính tổng các số từ 1 đến 100. Rõ ràng điều này rất rắc rối nếu viết lần lượt các số hạng, thay vào đó sử dụng ký hiệu tổng xích ma Σ sẽ thuận tiện hơn:

$$\sum_{i=1}^{100} i = 1 + 2 + \cdots + 99 + 100 = 5050$$

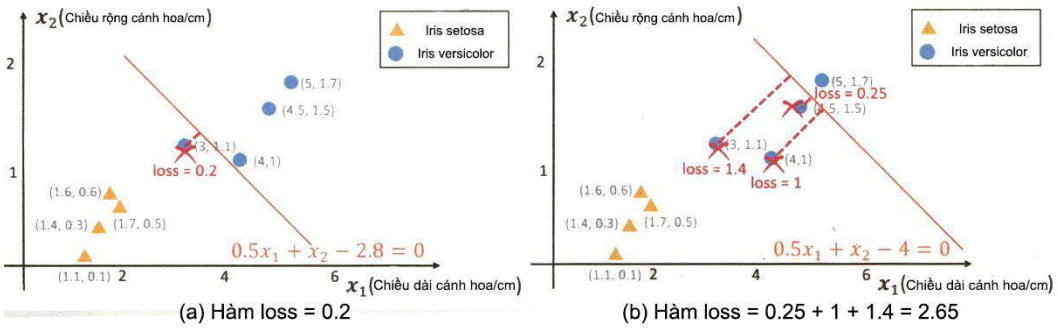
i = 1 bên dưới ký hiệu Σ chỉ ra rằng biến dùng để tính tổng là i và bắt đầu bằng 1, số 100 bên trên cho biết đếm đến khi i = 100.

Chúng ta sử dụng ký hiệu **max** để biểu diễn việc tìm số lớn nhất. Ví dụ max (0, -1) = 0, max (0, 1, 2) = 2.

Hàm loss của thuật toán perceptron bên trên có nghĩa với mỗi mẫu dữ liệu trong tập dữ liệu huấn luyện đều thực hiện việc tính $-y \times (a_1 x_1 + a_2 x_2 + b)$, sau đó kết quả được so sánh với 0 - nếu lớn hơn 0 thì hàm loss tăng, còn nếu không thì hàm loss giữ nguyên. Ta đã biết trong thuật toán perceptron, các mẫu có $-y \times (a_1 x_1 + a_2 x_2 + b) \geq 0$ là các mẫu bị phân loại sai, việc tính toán hàm loss tương đương với việc xác định tất cả những mẫu bị phân loại sai.

Rõ ràng nếu không có mẫu nào bị phân loại sai thì hàm loss có giá trị bằng 0, càng nhiều mẫu bị phân loại sai thì hàm có giá trị càng lớn.

Chúng ta sử dụng công thức tính hàm loss phía trên để tính toán. Trong *Hình 2-11 (a)* có một mẫu bị phân loại sai với hàm loss có giá trị = 0.2. Trong *Hình 2-11 (b)*, có nhiều dữ liệu bị phân loại sai hơn và giá trị hàm loss tăng lên đến 2.65. Ngoài ra *Hình 2-11 (b)* cũng cho thấy dữ liệu bị phân loại sai càng cách xa đường phân tách (càng sai) thì hàm loss có giá trị càng lớn.



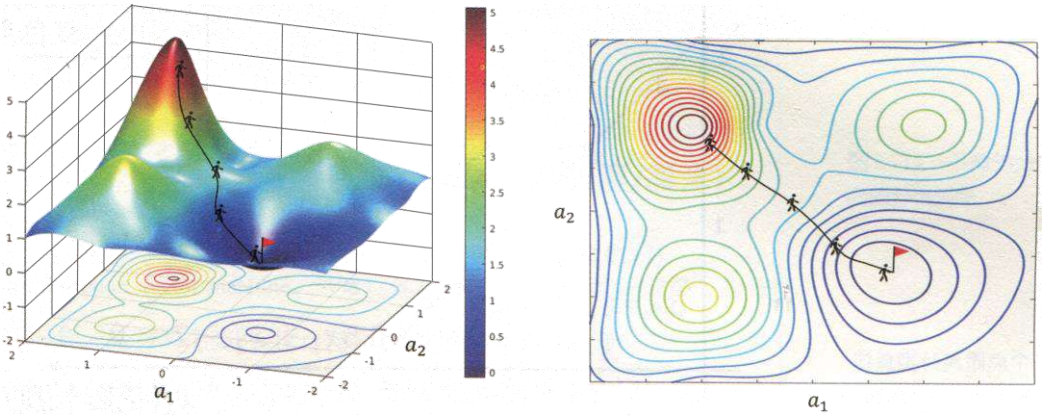
(a) Hàm loss = 0.2 (b) Hàm loss = 0.25 + 1 + 1.4 = 2.65

Hình 2-11: Giá trị hàm loss của perceptron trên dữ liệu phân loại sai

Sau khi sử dụng hàm loss để đo mức độ phân loại sai của bộ phân loại, chúng ta có thể sử dụng phương pháp tối ưu hóa để điều chỉnh các tham số để giảm sự phân loại sai. Quy tắc điều chỉnh các tham số trong thuật toán perceptron ở trên là ứng dụng cụ thể của phương pháp tối ưu hóa trên hàm loss perceptron. Hàm loss thu được từ toàn bộ dữ liệu huấn luyện, việc sử dụng nó để cập nhật tham số tương đương với việc sử dụng dữ liệu phân loại sai trong toàn bộ tập dữ liệu. Bước thứ 2 trong thuật toán học perceptron là chọn một mẫu ngẫu nhiên, nếu mẫu đó bị phân loại sai thì sử dụng để cập nhật tham số, sau đó lặp lại cho đến khi không còn mẫu nào bị phân loại sai nữa. Đó là một thay đổi nhỏ của hàm học perceptron sử dụng phương pháp tối ưu hóa để thu được thuật toán học perceptron.

Tối ưu hóa (optimization) là quá trình thay đổi tham số để làm loss function đạt giá trị nhỏ nhất. Chúng ta sử dụng một ví dụ trực quan để hiểu về tối ưu hóa. Để tiện theo dõi, chúng ta chỉ sử dụng 2 tham số a_1, a_2. *Hình 2-12* biểu diễn các giá trị của cặp tham số a_1, a_2 và giá trị của hàm loss tương ứng.

Ở hình bên trái, ta biểu diễn các giá trị hàm loss lên hệ tọa độ ba chiều, hình bên phải là bản đồ đường đồng mức, các hàm loss có giá trị giống nhau ở trên cùng một đường.



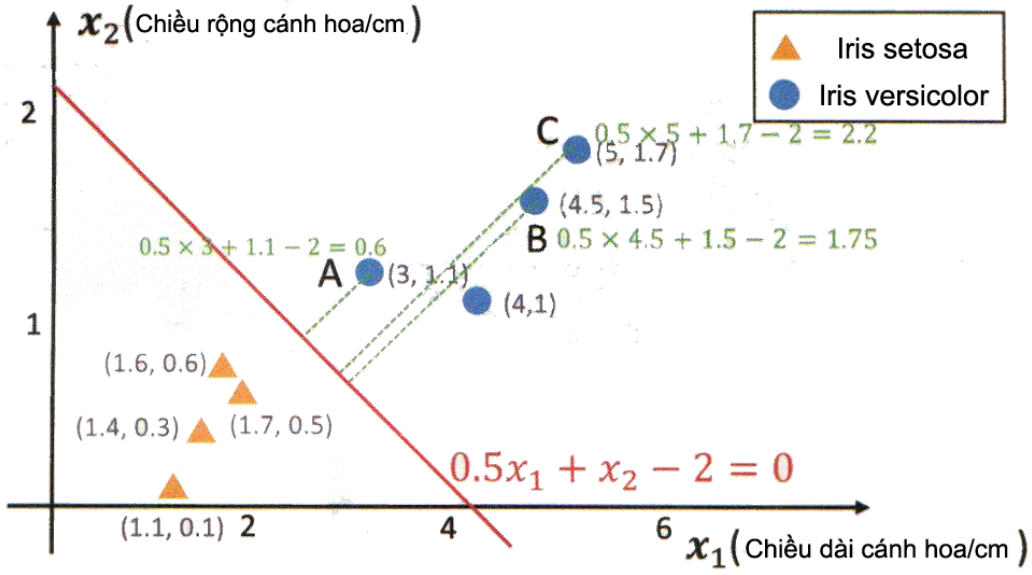
Hình 2-12: Sơ đồ quá trình tối ưu hóa

Hình bên trái *Hình 2-12* biểu diễn giá trị của hàm loss giống đồi núi. Có các ngọn đồi ngọn núi và cả thung lũng. Điểm mà hàm loss có giá trị nhỏ nhất là điểm có độ cao nhỏ nhất. Mục tiêu của việc tối ưu hóa là khiến hàm loss đạt giá trị nhỏ nhất, tức là đi tới điểm thấp nhất ở thung lũng. Quá trình tối ưu hóa là quá trình điều chỉnh a_1, a_2 để đi đến điểm thấp nhất đó.

Nếu ta bước theo hướng xuống dưới, tức là giảm độ cao (giảm giá trị hàm loss) sau mỗi lần điều chỉnh, thì đến một thời điểm ta sẽ tới được vị trí thấp nhất (giá trị hàm loss nhỏ nhất).

Support Vector Machines

Trong phần trước, chúng ta đã được giới thiệu về thuật toán học perceptron. Thông qua các thí nghiệm, chúng ta biết rằng dưới cùng một dữ liệu huấn luyện, thuật toán perceptron sẽ nhận được các đường phân loại khác nhau do lựa chọn thông số ban đầu khác nhau hoặc learning rate khác nhau. Những đường phân loại khác nhau này có thể phân tách các loại dữ liệu khác nhau, vậy có bất kỳ ưu điểm và nhược điểm nào giữa chúng không?



Hình 2-13: Khoảng cách giữa điểm và đường phân loại có thể biểu thị mức độ tin cậy trong dự đoán phân loại.

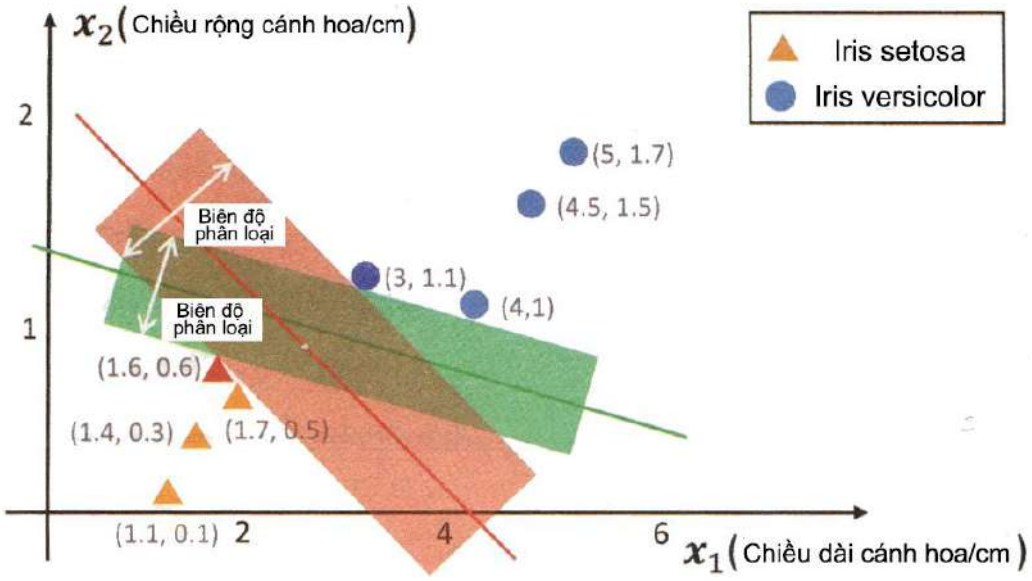
Trước tiên chúng ta hãy xem một ví dụ. Trong *Hình 2-13*, có ba mẫu A, B và C đều ở một bên của đường phân loại. Điểm C nằm ngoài cùng, chúng ta thấy yên tâm, chắc chắn rằng kết quả dự đoán sẽ chính xác. Điểm A rất gần với đường phân loại, chúng ta không chắc chắn về dự đoán này. điểm B nằm giữa điểm A và điểm C, mức độ tin cậy cũng ở mức trung bình giữa A và C.

Nói chung, khoảng cách của một điểm tới đường phân loại có thể cho thấy mức độ tin cậy của trong dự đoán phân loại (liệu sẽ chính xác hay bị phân loại sai). Làm thế nào để biểu diễn khoảng cách từ điểm đến đường phân loại? Trong trường hợp đường phân loại được xác định, $|a_1x_1 + a_2x_2 + b|$ có thể đại diện cho khoảng cách này. Do đó, $y \times (a_1x_1 + a_2x_2 + b)$ không chỉ cho biết độ chính xác của phân loại, mà còn cho biết mức độ tin cậy của dự đoán. Việc tính toán trong Hình 2-13 chỉ đơn giản là hỗ trợ kết luận này. Học sinh sẽ thấy rằng công thức này cũng được sử dụng trong hàm loss của perceptron.

Trong ví dụ trên, chúng ta đã cố định đường phân loại, phân tích các điểm dữ liệu khác nhau và nhận được kết luận một cách trực quan rằng "khoảng cách tới đường phân loại càng xa thì độ tin cậy của dự đoán phân loại càng cao". Tương tự, sau khi đưa vào một loạt dữ liệu huấn luyện, chúng ta hy vọng rằng đường phân loại thu được sau khi huấn luyện sẽ cách xa điểm dữ liệu hơn, do đó dự đoán phân loại trở nên đáng tin cậy hơn. Trong thực tế, chúng ta chỉ cần chú ý khoảng cách từ điểm gần nhất với đường

phân loại, khiến chúng cách xa đường phân loại hơn. Chúng ta gọi tổng khoảng cách từ 2 điểm gần nhất của 2 loại tới đường phân loại là **biên độ phân loại** (classification margin).

Hình 2-14 cho thấy 2 đường phân loại chính xác hai loại hoa, phần bóng mờ biểu thị biên độ phân loại của mỗi đường. Phần bóng mờ của đường màu cam lớn hơn, biểu thị biên độ phân loại lớn hơn, do đó kết quả phân loại đáng tin cậy hơn.



Hình 2-14: Những đường phân loại khác nhau có biên độ phân loại khác nhau

SVM là bộ phân loại có biên độ phân loại lớn nhất trong không gian đặc trưng. Giống như perceptron, nó phân loại hai loại khác nhau. Bộ phân loại tuyến tính là một trong các bộ phân loại, tương tự, SVM tuyến tính cũng là một trong các SVM. Nếu không được chỉ định khác, SVM được đề cập đến trong cuốn sách được coi là SVM tuyến tính.

Về mặt trực quan, rất dễ dàng để tìm ra đường phân loại với biên độ phân loại lớn nhất. Học sinh có thể tưởng tượng lấy một mẩu phấn rất dày vẽ một đường phân tách hai loại dữ liệu sao cho các đường được vẽ là dày nhất. Đường dày nhất được vẽ có biên độ phân loại lớn nhất là đường mà SVM tìm kiếm.

Các bạn sẽ tìm được đường dày nhất được vẽ trên hình 2-15, nó chỉ liên quan đến một số điểm dữ liệu nhất định. Nếu ta chỉ giữ những điểm trên đồ thị tiếp xúc với phần bóng mờ và bỏ các điểm khác, ta có thể sử dụng đường có cùng độ dày để phân biệt hai loại dữ liệu, tức là kết quả phân loại thu được không giống nhau. Điểm tiếp xúc với phần bóng mờ được gọi là support vector, đó cũng là nguồn gốc của tên gọi Support Vector Machine.

Support vector là vector dùng để xác định đường phân loại và là vector khó xác định nhất. Có thể coi đó là dữ liệu chứa nhiều thông tin nhất trong việc giải quyết nhiệm vụ phân loại.

Thử nghiệm 2-2

1. Sử dụng thuật toán SVM được cung cấp để huấn luyện bộ phân loại.

2. Hiển thị riêng biệt biên độ phân loại của bộ phân loại huấn luyện bởi perceptron và SVM, sau đó so sánh.

Kiến thức bổ sung: Hàm loss và SVM

Bằng cách sử dụng ví dụ vẽ các đường thẳng với phấn, chúng ta có thể thấy biên độ phân loại tối đa. Liệu chúng ta có thể viết hàm loss của SVM như perceptron, và sau đó giải quyết nó bằng phương pháp tối ưu hóa trên dữ liệu huấn luyện?

Câu trả lời là có. Tiếp theo, chúng ta sẽ tìm hiểu sơ lược về quá trình thiết lập hàm loss của SVM. Chúng ta thảo luận trong một trường hợp đơn giản: Phân biệt 2 loại dữ liệu và có thể phân loại bằng một đường thẳng.

Mục tiêu của chúng ta là xác định các tham số a_1, a_2, b cho đường phân loại $a_1x_1 + a_2x_2 + b = 0$. Phương pháp SVM là tối đa hóa biên độ phân loại. Nếu điểm dữ liệu (x_1, x_2) được phân loại chính xác, khoảng cách từ điểm này đến đường phân loại có thể được tính theo công thức sau:

$$\frac{|a_1x_1 + a_2x_2 + b|}{\sqrt{{a_1}^2 + {a_2}^2}} = y \times \frac{a_1x_1 + a_2x_2 + b}{\sqrt{{a_1}^2 + {a_2}^2}}$$

Từ đây ta có thể xác định khoảng cách hình học $\gamma^{(i)}$ của một điểm $({x_1}^{(i)}, {x_2}^{(i)})$ bất kỳ tới đường phân loại:

$$\gamma^{(i)} = y^{(i)} \times \frac{a_1{x_1}^{(i)} + a_2{x_2}^{(i)} + b}{\sqrt{{a_1}^2 + {a_2}^2}}$$

Mối quan hệ giữa khoảng cách hình học và khoảng cách từ điểm chúng ta nói đến đường phân loại là gì? Nếu các điểm dữ liệu được phân loại chính xác, khoảng cách hình học là khoảng cách từ điểm đó đến đường phân loại, nếu bị phân loại sai, chúng khác nhau theo dấu dương hoặc dấu âm. Do đó, khoảng cách hình học là trị tuyệt đối của khoảng cách từ điểm đến đường thẳng.

Chúng ta định nghĩa khoảng cách hình học của tất cả dữ liệu huấn luyện tới đường phân loại là khoảng cách từ điểm dữ liệu gần đường phân loại nhất.

$$\gamma = \min_{i=1,\dots,N} \gamma^{(i)}$$

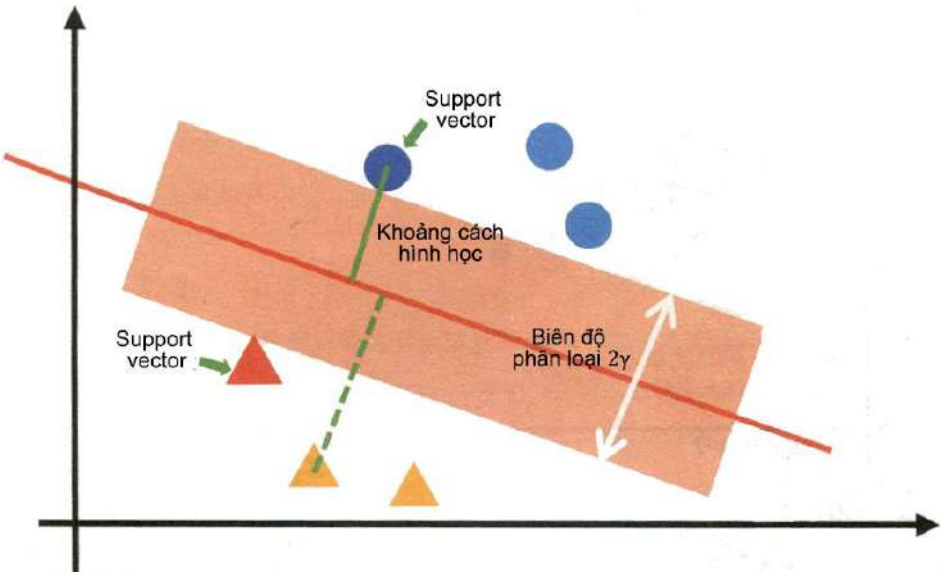
Biểu tượng min chỉ ra rằng biểu thức thứ hai được lấy giá trị nhỏ nhất. Từ Hình 2-15, chúng ta có thể thấy biên độ phân loại bằng 2 lần khoảng cách hình học 2γ.

Do đó, việc tối đa hóa biên độ phân loại là tối đa 2γ, biểu thị bằng ký hiệu toán học là $\max_{a_1,a_2,b} 2\gamma$, ký hiệu max biểu thị giá trị của biểu thức được tối đa hóa và a_1, a_2, b bên dưới max thể hiện các tham số có thể thay đổi. Để 2γ lớn nhất tương đương với $\frac{2}{\gamma}$ nhỏ nhất, được biểu diễn bằng ký hiệu toán học là $\min_{a_1,a_2,b} \frac{2}{\gamma}$. Đây có thể được coi là một hàm loss của SVM, và chúng ta muốn làm hàm loss này đạt giá trị nhỏ nhất.

Giải pháp cho vấn đề này cũng cần đảm bảo rằng khoảng cách hình học của mỗi điểm dữ liệu huấn luyện đến đường phân loại nhỏ nhất bằng γ. Bằng cách này, toàn bộ vấn đề tối ưu hóa có thể được diễn tả như sau:

$$\min_{a_1,a_2,b} \frac{2}{\gamma}.$$

Thỏa mãn với mọi i, $y^{(i)} \times \frac{a_1 {x_1}^{(i)} + a_2 {x_2}^{(i)} + b}{\sqrt{{a_1}^2 + {a_2}^2}} \geq y$



Hình 2-15: Khoảng cách hình học và biên độ phân loại

Vậy chúng ta có được hàm loss hoàn chỉnh của SVM.

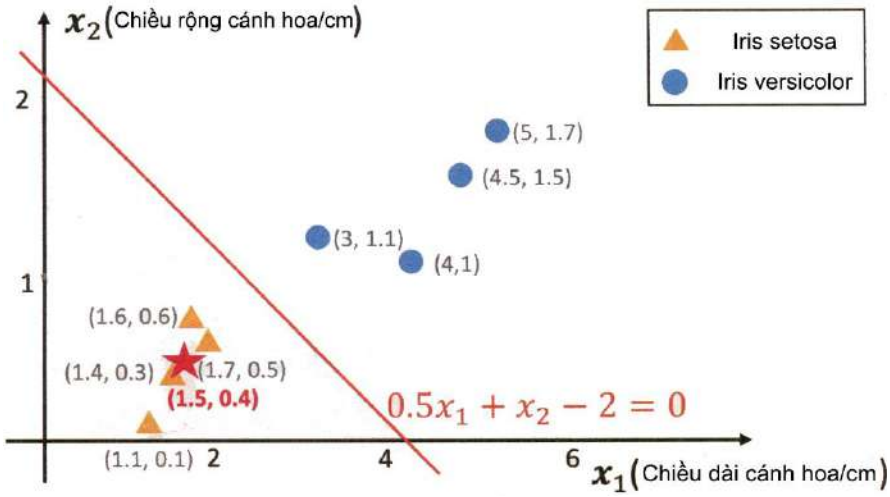
Vấn đề này có thể được giải quyết phương pháp tối ưu hóa, nhưng quá trình giải quyết vấn đề này vượt quá khả năng kiến thức hiện tại của học sinh. Sau khi học các công cụ toán học có liên quan, học sinh mới có thể giải quyết nó.

2.4. Thực tế: kiểm tra và ứng dụng

Chúng ta đã học về thuật toán của hai bộ phân loại: perceptron và SVM. Sau khi hoàn thành, chúng ta muốn biết hiệu quả phân loại của bộ phân loại và thuật toán nào đạt được hiệu suất tốt nhất, vì vậy chúng ta cần thử nghiệm.

Thử nghiệm giống như bài kiểm tra. Trong kỳ thi, học sinh phải làm một bài kiểm tra trả lời các câu hỏi, sau đó giáo viên sẽ chấm và cho điểm.

Tương tự, trong giai đoạn thử nghiệm, bộ phân loại sẽ đối mặt với một lô dữ liệu thử nghiệm và đưa ra các dự đoán cho mỗi mẫu thử. Nếu kết quả phân loại giống với nhãn của mẫu thử thì việc phân loại là chính xác, nếu không thì tức là phân loại không chính xác. Ví dụ, trong trường hợp phân biệt các loài hoa, dữ liệu thử nghiệm có một bông hoa có chiều dài cánh hoa là 1,5 cm và chiều rộng là 0,4 cm. Vector đặc trưng $(1{,}5, 0{,}4)$ của mẫu thử được vẽ trong không gian đặc trưng nằm ở vị trí ngôi sao năm cánh màu đỏ trong *Hình 2-16*. Trong trường hợp này sự phân loại là chính xác.



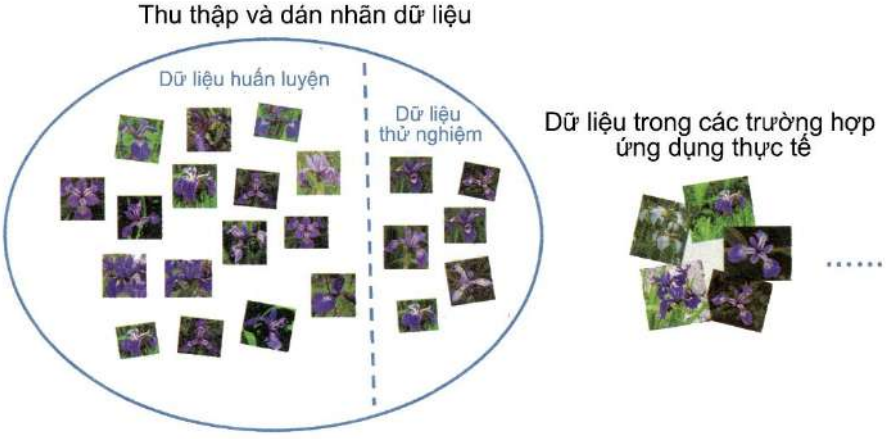
Hình 2-16: Thử nghiệm

Sau khi thử nghiệm toàn bộ bộ mẫu thử, chúng ta đếm số mẫu được dự đoán chính xác, tỉ lệ mẫu đúng với tổng số mẫu có thể phản ánh độ chính xác của dự đoán (classification accuracy) - tương đương với số điểm do giáo viên đưa ra sau khi chấm.

$$\text{Độ } chính\ xác\ = \frac{Số\ mẫu\ phân\ loại\ đúng}{Tổng\ số\ mẫu\ phân\ loại} \times 100\%$$

Biết được độ chính xác phân loại, chúng ta có thể so sánh mức hiệu quả của các thuật toán và bộ phân loại, qua đó chọn bộ phân loại tốt nhất.

Sau khi kiểm tra và chọn một bộ phân loại tối ưu, bước tiếp theo là ứng dụng vào thực tế. Nếu chúng ta thấy một bông hoa thuộc một trong hai loại hoa đã được huấn luyện để phân loại và muốn biết loại của nó, ta chỉ cần đo chiều dài, chiều rộng cánh hoa rồi nhập vào bộ phân loại được huấn luyện, bộ phân loại sẽ đưa ra dự đoán. Quá trình này là giai đoạn ứng dụng của bộ phân loại.



Hình 2-17: Dữ liệu huấn luyện, dữ liệu thử nghiệm và dữ liệu ứng dụng

Sự khác biệt giữa thử nghiệm và ứng dụng là gì? Xét theo chức năng, phép thử được sử dụng để đánh giá hiệu suất của bộ phân loại, trong khi đó ứng dụng là việc sử dụng bộ phân loại trong tình huống thực tế. Chúng ta thường huấn luyện nhiều bộ phân loại cho một vấn đề, kiểm tra trên dữ liệu thử nghiệm, chọn bộ phân loại hoạt động tốt nhất và để ứng dụng vào thực tế. Nếu xét trên dữ liệu, giai đoạn thử nghiệm sử dụng các mẫu đã được biết trước loại (mẫu có nhãn), còn quá trình ứng dụng sử dụng dữ liệu từ thực tế không được dán nhãn và phức tạp hơn như trong mô tả ở *Hình 2-17*.

.

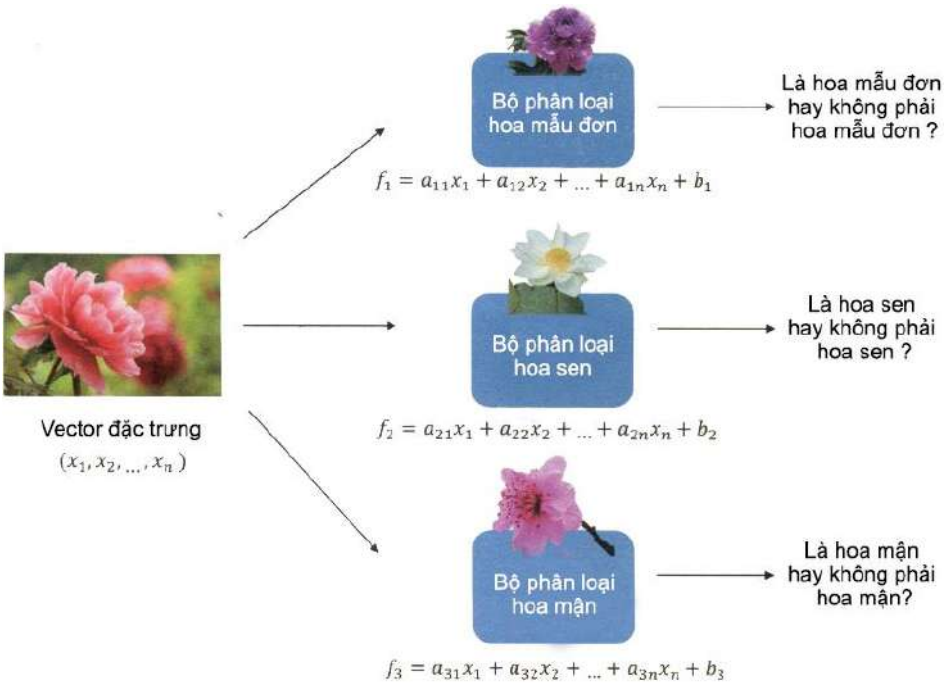
2.5. Phân loại đa danh mục

Ở trên chúng ta đã được giới thiệu về phân loại hai loại đối tượng (phân loại nhị phân - binary classification). Trong thực tế, chúng ta thường cần phân loại nhiều loại, chẳng hạn như hoa mẫu đơn, hoa sen và mận. Vậy làm cách nào để chúng tôi giải quyết các vấn đề phân loại đa danh mục (multiclass classification) như vậy?

Nhìn lại ví dụ trước, chúng ta đã sử dụng một hàm phân loại để giải quyết vấn đề hai loại. Trong bài toán đa phân loại, chúng ta có thể dịch bài toán nhiều loại khác nhau thành nhiều nhiều bài toán hai loại không? Chúng ta sẽ sử dụng nhiều hàm nhị phân, mỗi hàm có tham số riêng và chỉ chịu trách nhiệm phân biệt một danh mục. Như thể hiện trong *Hình 2-18*, chúng ta có ba bộ phân loại của hoa mẫu đơn, hoa sen và mận riêng biệt. Mỗi bộ chỉ chịu trách nhiệm phân biệt một loại: Là hoa mẫu đơn hay không phải hoa mẫu đơn, là hoa sen hay không phải hoa sen, là mận hay không phải là mận. Khi vectơ đặc trưng của hình ảnh được nhập vào, ba bộ phân loại có thể xuất dự đoán của riêng chúng. Bằng cách kết hợp ba kết quả dự đoán, chúng ta có thể thu được kết quả dự đoán của vấn đề đa phân loại.

Cụ thể, nếu kết quả f1 dương và f2, f3 âm, thì chúng ta có thể chắc chắn rằng loại của hoa là hoa mẫu đơn. Tuy nhiên, có thể có trường hợp hai kết quả cùng dương. Làm thế nào để đưa ra một đánh giá toàn diện vào lúc này? Đây là vấn đề mang tính không chắc chắn, như chúng ta nghe trong dự báo thời tiết cho biết tỉ lệ có thể mưa là 80% - tức là có thể mưa hoặc không mưa, nhưng xác suất để mưa khá cao. Vậy chúng ta có thể chấp nhận chiến lược tương tự không?

Trong thực tế, chúng ta có thể truyền số lượng các bộ phân loại và các giá trị đầu ra output f1, f2, f3 đến một hàm mũ chuẩn hóa. Nó có thể biến đầu ra thành một xác suất - cho biết khả năng đối tượng đầu vào input thuộc loại nào. Hàm mũ chuẩn hóa được sử dụng rộng rãi trong mạng nơ ron vấn đề đa phân loại trong chương tiếp theo.



Hình 2-18: Sơ đồ đa phân loại

Hàm mũ chuẩn hóa (softmax) "nén" một vectơ (chẳng hạn như các giá trị đầu ra của nhiều hàm nhị phân thành một vectơ) thành một vectơ khác, sao cho mỗi phần tử có giá trị trong khoảng $(0,1)$ và tổng các phần tử bằng 1. Quá trình này được gọi là chuẩn hóa. Công thức cụ thể của hàm softmax:

$$\sigma(z_j) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

Trong đó $j = 1, \ldots, K$. Vectơ này có kích thước bằng K. Đầu tiên các phần tử được chuyển đổi theo hàm số mũ, sau đó tính tỷ số giữa kết quả này và tổng các kết quả. Vì phép biến đổi theo hàm số mũ được thực hiện trước và sau đó được chuẩn hóa nên hàm softmax được gọi là hàm mũ chuẩn hóa.

Bảng 2-1 liệt kê các phép tính cho hàm mũ chuẩn hóa. Nếu ba phân loại có kết quả -1, 2, 3 sẽ tạo thành một vectơ (-1, 2, 3). Sau khi chuẩn hóa hàm mũ, vectơ thu được là (0.013, 0.265, 0.722), các phần tử có tổng là 1. Giá trị tối đa là 0.722 trong vector đầu ra tương ứng với giá trị lớn nhất của 3 trong vector đầu vào. Hàm này thường dùng để chuẩn hóa vectơ, làm nổi bật giá trị lớn nhất và bỏ qua các giá trị khác dưới mức nhỏ nhất quy định.

Input z	-1	2	3	Tổng
Chuyển đổi	$e^{-1} \approx 0.368$	$e^{2} \approx 7.389$	$e^{3} \approx 20.086$	27.843
Kết quả chuẩn hóa	$\frac{0.368}{27.843} \approx 0.013$	$\frac{7.389}{27.843} \approx 0.265$	$\frac{20.086}{27.843} \approx 0.722$	1

Bảng 2.1: Bảng tính hàm mũ chuẩn hóa

Sau khi chuẩn hóa hàm mũ, giá trị của chúng đều lớn hơn 0 và có tổng là 1. Có thể coi giá trị đó là xác suất khả năng rơi vào một loại. Ví dụ, đầu ra (0.013, 0.265, 0.722) tương đương xác suất 1.3% thuộc về loại đầu tiên, 26.5% thuộc về loại thứ hai, và 72.2% khả năng là loại thứ ba. Vì vậy, đầu vào có nhiều khả năng là loại thứ ba nhất.

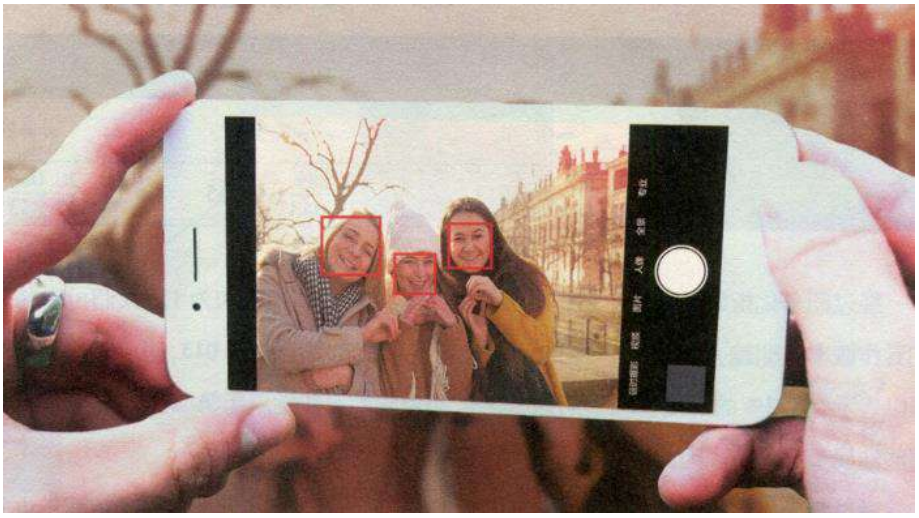
Bằng cách chuẩn hóa hàm mũ, giá trị đầu ra của bộ phân loại có nhiều ý nghĩa hơn, không chỉ cho chúng ta biết loại đầu ra, mà còn cho thấy xác suất dự đoán chính xác của bộ phân loại. Ví dụ, nếu xác suất đầu ra là 99% là hoa mẫu đơn thì dự đoán này là rất chính xác, nếu xác suất đầu ra là 65% là mận thì dự đoán này không chắc chắn, chúng ta cần cân nhắc kết quả này.

2.6. Hai vấn đề phân loại ứng dụng trong cuộc sống

Chương này giới thiệu bài toán của hai phân loại - phân loại các thứ thành hai loại. Vấn đề này có một loạt các ứng dụng trong cuộc sống thực. Vấn đề "Đây là gì ?" gặp phải trong cuộc sống thuộc về vấn đề phân loại 2 loại. Đây có phải là một khuôn mặt của con người? Đây có phải là một triệu chứng của bệnh ung thư? Đây có phải là nơi có khoáng sản? ... Dưới đây cuốn sách sẽ giới thiệu các vấn đề nhận diện khuôn mặt trong máy ảnh và phát hiện ung thư trong điều trị y tế để xem làm thế nào hai công nghệ phân loại được áp dụng trong cuộc sống thực.

Nhận diện khuôn mặt trong máy ảnh

Khi chúng ta chụp ảnh, cho dù sử dụng điện thoại di động hay máy ảnh SLR, khi ống kính nhắm vào khuôn mặt, các khung hình chữ nhật trên màn hình sẽ xuất hiện thể hiện vị trí các khuôn mặt như trong *Hình 2-19*. Vậy công nghệ này hoạt động như thế nào?



Hình 2-19: Nhận diện khuôn mặt trong máy ảnh

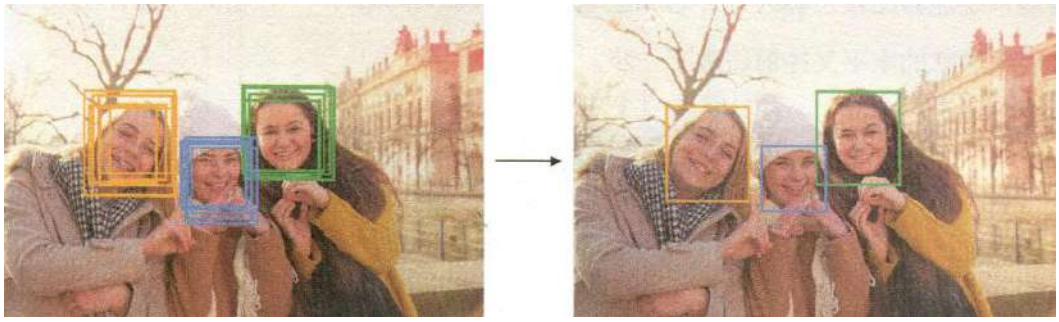
Công nghệ nhận diện khuôn mặt trong máy ảnh sử dụng kỹ thuật hai lớp. Toàn bộ quá trình được thể hiện trong *Hình 2-20*. Một bức ảnh đầu tiên được cắt thành các khối hình ảnh một cách dày đặc và có thể chồng lên nhau để liên tục cắt ảnh thành các khối ảnh nhỏ. Một bức ảnh thường có hàng nghìn khối ảnh bị cắt.

Sau đó, mỗi khối hình ảnh sẽ đi qua bộ phân loại khuôn mặt để xác định xem đó có phải là khuôn mặt con người hay không. Bộ phân loại khuôn mặt là bộ phân loại 2 loại, giống như phân loại hoa đã được nhắc đến ở các phần trước. Đối với một khối hình ảnh được dự đoán là khuôn mặt người, máy ảnh sẽ hiển thị vị trí của khung trên màn hình — đây chính là bí mật của công nghệ phát hiện khuôn mặt trong máy ảnh.

Các bạn có thể thắc mắc về việc các khuôn mặt có thể sẽ có kích thước khác nhau. Thực tế khi cắt ảnh sẽ có rất nhiều kích thước khác nhau từ lớn đến nhỏ được cắt. Điều đó đảm bảo không bỏ sót các khuôn mặt với kích thước khác nhau. Do đó, số lượng các khối ảnh được cắt là rất lớn. Vì việc sử dụng bộ phân loại khá đơn giản, lượng tính toán không nhiều và sử dụng các kỹ thuật khác để tối ưu tốc độ xử lý, thời gian thực hiện toàn bộ quá trình được rút ngắn khiến chúng ta khó có thể cảm nhận, tốc độ đạt được gần như thời gian thực.

Hình 2-20: Quá trình nhận diện khuôn mặt trong máy ảnh

Các bạn cũng có thể thấy các khối hình ảnh có kích thước và vị trí khác nhau có thể được xác định là khuôn mặt cùng một lúc, do đó sẽ có nhiều khung chồng chéo nhau. Trong thực tế, đây là trường hợp được minh họa trong hình bên trái của *Hình 2-21*, các vị trí và khối khác nhau của các kích cỡ khác nhau cùng được xác định là khuôn mặt người. Những khối này đều ở gần mặt, và chúng ta có thể kết hợp các khối này thành một bằng kỹ thuật dung hợp hậu xử lý để có được kết quả ở bên phải.



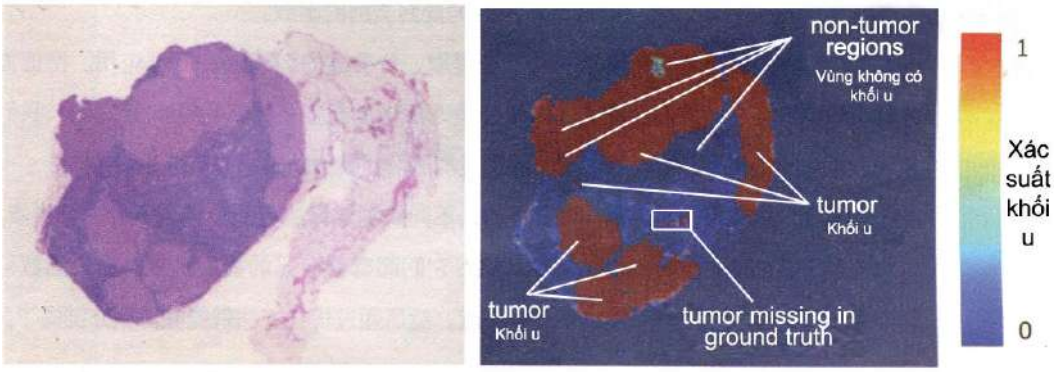
Hình 2-21: Dung hợp nhiều khung hình trong nhận diện khuôn mặt

Phát hiện ung thư

Đánh giá liệu có hay không có ung thư là một ứng dụng của phân loại trong lĩnh vực y tế dựa trên hình ảnh mẫu mô sinh học của bệnh nhân. Nhà nghiên cứu có thể chẩn đoán ung thư sau khi kiểm tra mẫu mô sinh học của bệnh nhân. Chẩn đoán của họ là rất quan trọng đối với việc điều trị của bệnh nhân. Tuy nhiên, việc xem xét các phần bệnh lý rất phức tạp, đòi hỏi chuyên môn sâu rộng và nhiều năm kinh nghiệm.

Xét trên góc nhìn về phân loại, phát hiện ung thư là một vấn đề phân biệt hai loại - xác định xem mỗi vùng của một mẫu mô sinh học là một khối u. Hiện nay, với sự tiến bộ của các hệ thống trí thông minh nhân tạo trong chẩn đoán một số bệnh ung thư, tỷ lệ chính xác của hệ thống trí tuệ nhân tạo để chẩn đoán ung thư đang dần dần tiến gần hơn đến các bác sĩ có kinh nghiệm. Trong *Hình 2-22*, hình ảnh bên trái là một hình ảnh của sinh thiết hạch bạch

huyết, và hình ảnh bên phải là một bài kiểm tra gần đây của Google, kết quả này đạt đến trình độ của con người trong dữ liệu thử nghiệm. Kết quả này rất ấn tượng, nhưng các mẫu mô thực tế của bệnh ung thư phức tạp hơn, các hình ảnh bệnh lý sẽ bị ảnh hưởng bởi các yếu tố khác, và thậm chí các hình ảnh lát bệnh lý hiếm gặp cũng có thể xuất hiện. Các bác sĩ có thể dựa vào kinh nghiệm và kiến thức để đối phó với những tình huống này. Việc chuẩn đoán của trí thông minh nhân tạo vẫn còn một khoảng cách lớn trong vấn đề này.



Hình 2-22: Lát hình ảnh phát hiện ung thư lĩnh vực bệnh lý

Trong phát hiện ung thư, lành tính và ác tính thường khó phân biệt. Ví dụ trong *Hình 2-23*, một khối u ác tính có di căn ung thư vú và một mô bình thường, đại thực bào, có vẻ giống nhau. Hệ thống trí tuệ nhân tạo hiện tại có thể phân biệt chính xác khối u với đại thực bào - một bước tiến về phía cấp độ của các bác sĩ có kinh nghiệm.

Hình 2-23: Phân biệt chính xác khối u và đại thực bào trong hạch bạch huyết

Những trí thông minh nhân tạo thực hiện việc chuẩn đoán đang dần chuyển từ phòng thì nghiệm sang hỗ trợ các bác sĩ trong chẩn đoán của họ và

giúp đưa ra chẩn đoán ung thư chính xác và kịp thời hơn. Điều này không chỉ cải thiện hiệu quả mà còn làm giảm gánh nặng cho các bác sĩ, đồng thời giúp kết quả xét nghiệm chính xác hơn và giúp bệnh nhân đạt được các cơ hội điều trị kịp thời. Công nghệ không ngừng phát triển và dự kiến trong tương lai gần, công nghệ trí tuệ nhân tạo sẽ được sử dụng rộng rãi trong lĩnh vực y tế.

2.7. Tóm tắt chương

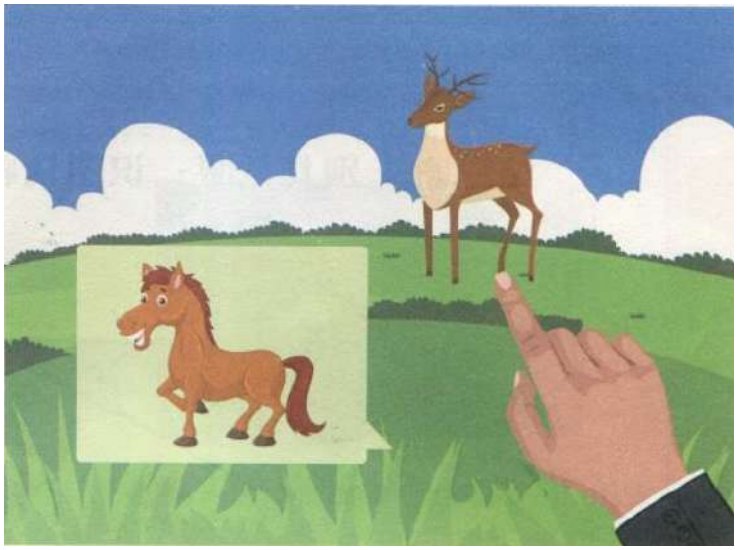
Phân loại là quá trình gán cho một đối tượng nào đó nhãn đúng với loại của đối tượng đó và được sử dụng rộng rãi trong cuộc sống. Các đặc trưng và quá trình phân loại là khái niệm quan trọng. Đặc trưng là thuộc tính của đối tượng được chọn dựa trên đặc điểm của đối tượng đó và được biểu diễn bởi vector đặc tính.

Quá trình phân loại có thể được chia thành ba giai đoạn - trích xuất đặc trưng, huấn luyện bộ phân loại và ứng dụng thử nghiệm. Trích xuất đặc trưng là một quá trình từ dữ liệu đến các vectơ đặc trưng và là trọng tâm của các phương pháp phân loại truyền thống. Sau khi có được các vector đặc trưng, chúng ta sử dụng dữ liệu và các thuật toán để huấn luyện bộ phân loại. Khi bộ phân loại được kiểm tra thử nghiệm, nó có thể được áp dụng trong cuộc sống thực.

Việc huấn luyện bộ phân loại được thực hiện bằng thuật toán huấn luyện và sử dụng các thuật toán huấn luyện khác nhau có thể nhận được các bộ phân loại khác nhau. Chương này giới thiệu thuật toán huấn luyện SVM và perceptron, mỗi thuật toán có hàm loss riêng. Hàm loss đo lường mức độ lỗi của bộ phân loại trong quá trình huấn luyện, sau đó thu được bộ phân loại bằng phương pháp tối ưu hóa.

Chương III: Nhận dạng ảnh

Minh thích nhiếp ảnh và có vô số bức ảnh trong album, một số là những sinh vật đáng yêu trong vườn thú, một số là động vật hoang dã. Có các bức ảnh về những hoa đầy màu sắc, mèo con, chó con, xe hơi và máy bay. Khi nhìn vào những bức ảnh này, nhiều chi tiết không thể nhớ được nữa. Tên của loài động vật dễ thương này là gì? Tên của loài hoa này là gì? Chú chó nhỏ kia là loại chó gì thế? "Nếu máy tính có thể tự động nhận ra những vật thể này thì thật tốt biết bao." - Minh nghĩ.



Trong quá khứ đã ghi lại rằng nhiều người đã nhầm lẫn con nai là con ngựa; một số người muốn sử dụng chim ưng, đại bàng để săn thỏ mà không biết chúng, mua nhầm thành vịt và trở thành tiếng cười.

Tại thời điểm này, việc có thể phân biệt chính xác các loài động vật mà không bị ảnh hưởng bởi các yếu tố con người có giá trị như một kho báu. Vấn đề này có thể giải quyết được không ?

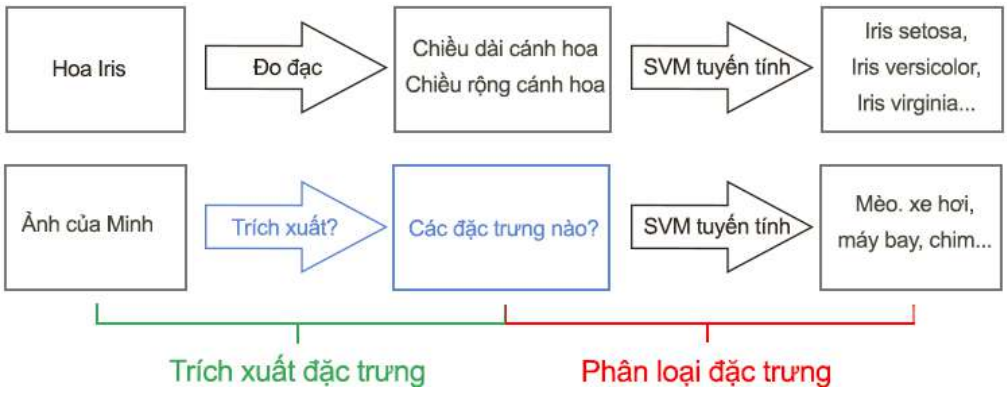
Ngày nay, hàng ngàn năm sau, sự xuất hiện của công nghệ học tập sâu đã cung cấp cho chúng ta một cơ hội để tạo ra những điều như thế. Chúng ta hãy nhìn vào những bí ẩn của công nghệ học tập sâu và mở ra cuộc hành trình huyền diệu của việc tạo ra kho báu giá trị này.

3.1. Phân loại hình ảnh dựa trên các đặc trưng

Có mèo con, chó con, xe hơi và máy bay trong album ảnh của Minh. Tuy nhiên, cũng có những bức ảnh về những thứ không chắc chắn lắm. Ví dụ, hình ảnh đầu tiên trong *Hình 3-1* là một chú chim cánh cụt hay con vật khác? Hình ảnh thứ hai là loại mèo nào? Xác định loại đối tượng trong ảnh là một nhiệm vụ phân loại. Thông qua các bài học trong Chương 2, chúng ta biết rằng nhiệm vụ phân loại bao gồm hai bước cốt lõi: trích xuất đặc trưng và phân loại đặc trưng. Như trong *Hình 3-2*, trong ví dụ về phân loại hoa, chúng ta trích xuất một vector đặc trưng hai chiều từ mẫu hoa bằng cách đo chiều dài và chiều rộng của cánh hoa. Vector đặc trưng này được nhập vào bộ phân loại, sau một loạt các phép tính bộ phân loại có thể xác định loại hoa này.

Chúng ta có thể thực hiện theo quy trình tương tự và thiết kế hệ thống để phân loại hình ảnh. Vậy chúng ta nên sử dụng đặc trưng nào cho nhiệm vụ phân loại hình ảnh? Làm thế nào để trích xuất chúng một cách hiệu quả từ những bức ảnh? Trước khi trả lời những câu hỏi này, trước tiên chúng ta hãy hiểu những hình ảnh trong mắt của máy tính trông như thế nào.

Hình 3-1: Album ảnh của Minh

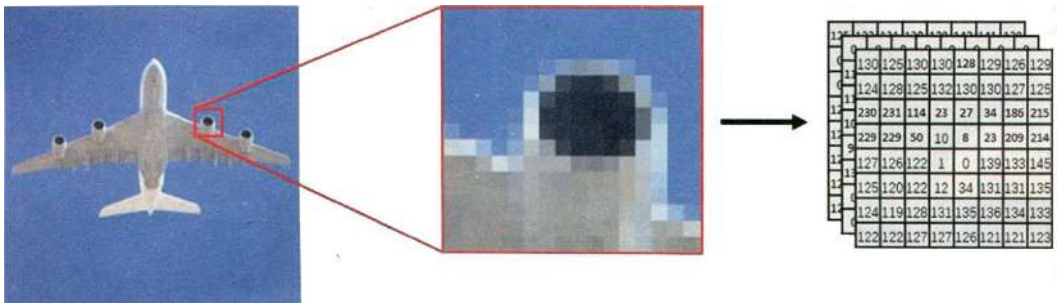


Hình 3-2: Hai bước cơ bản của nhiệm vụ phân loại

Hình ảnh trong con mắt của máy tính

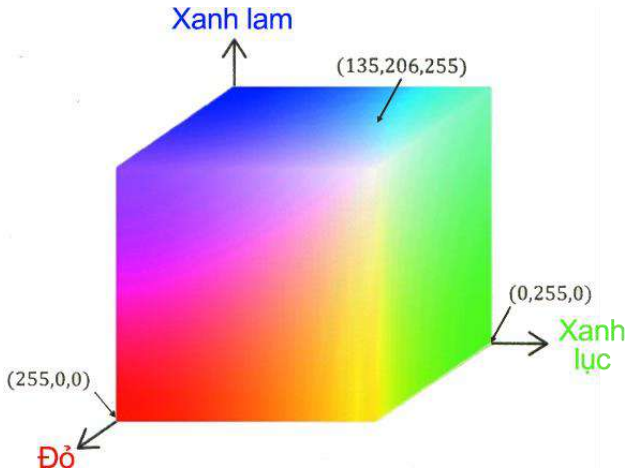
Trước khi tìm hiểu trích xuất đặc trưng hình ảnh, hãy xem cách một hình ảnh được biểu diễn trong máy tính. Như mô tả trong *Hình 3-3*, nếu phóng to một hình ảnh, chúng ta có thể thấy nó bao gồm các ô mắt lưới nhỏ, mỗi ô trong số đó là một điểm màu. Nếu chúng ta sử dụng các số khác nhau để biểu diễn các màu khác nhau, hình ảnh có thể được biểu diễn dưới dạng một mảng hình chữ nhật được gọi là ma trận để có thể được lưu trữ trong máy tính. Một mắt lưới được gọi là một điểm ảnh (pixel), số hàng và cột của lưới được gọi chung là độ phân giải. Chúng ta thường nói độ phân giải của ảnh là 1280×720, có nghĩa là hình ảnh có 1280 hàng, 720 cột các điểm ảnh. Ngược lại, nếu có ma trận các số, ta có thể chuyển đổi từng giá trị trong ma

trận thành màu tương ứng và hiển thị nó trên màn hình máy tính để tạo lại hình ảnh.



Hình 3-3: Biểu diễn hình ảnh trên máy tính

Có 2 loại ảnh là ảnh đen trắng và ảnh màu. Ảnh đen trắng sử dụng màu trong thang độ xám, còn ảnh màu sử dụng các màu. Màu trong thang độ xám chỉ có sự khác biệt mức độ sáng tối, chỉ cần một số để biểu diễn cho các màu khác nhau trong thang độ xám. Thông thường chúng ta sử dụng giá trị 0 cho màu tối nhất (đen), 255 cho màu sáng nhất (trắng), và một số nguyên từ 1 đến 254 cho màu xám với các sắc thái khác nhau. Đối với hình ảnh màu, chúng ta sử dụng ba số (R, G, B) để đại diện cho màu sắc thu được bằng cách chồng ba màu cơ bản lên nhau gồm màu đỏ (R), màu xanh lục (G), và màu xanh lam (B). Đối với mỗi màu cơ bản, chúng ta sử dụng một số nguyên nằm trong khoảng từ 0 đến 255 để chỉ ra mức độ đậm nhạt của thành phần màu này. Như thể hiện trong hình 3-4, số của thành phần màu nào càng lớn thì màu đó chiếm tỉ lệ càng lớn, ví dụ (255,0,0) có nghĩa là màu đỏ tinh khiết, (0, 255, 0) có nghĩa là màu xanh lục tinh khiết, (135, 206, 255) là màu xanh da trời.



Hình 3-4: Biểu diễn màu sắc

Một hình ảnh màu có thể được biểu diễn bằng một mảng các khối bao gồm các số nguyên. Mảng số được sắp xếp theo các khối lập phương như vậy là một tensor bậc 3. Chiều dài và chiều rộng của tensor bậc ba này chính là độ phân giải của hình ảnh và có chiều cao là 3. Đối với hình ảnh kỹ thuật số, chiều cao của tensor bậc ba cũng được gọi là số kênh, vì vậy hình ảnh màu có ba kênh. Ma trận là tensor bậc 2 (xem thêm Kiến thức bổ sung) nhưng cũng có thể được coi là một tensor bậc 3 với chiều cao là 1, vì vậy hình ảnh đen trắng có màu trong thang độ xám chỉ có một kênh.

Kiến thức bổ sung: Tensor

Tensor là một khái niệm cơ bản trong toán học, vật lý và kỹ thuật. Nhiều khái niệm chúng ta nhắc đến trước đây là các dạng tensors đặc biệt, ví dụ giá trị vô hướng (scalar) là tensor bậc 0, vector là tensor bậc 1, ma trận là tensor bậc 2.

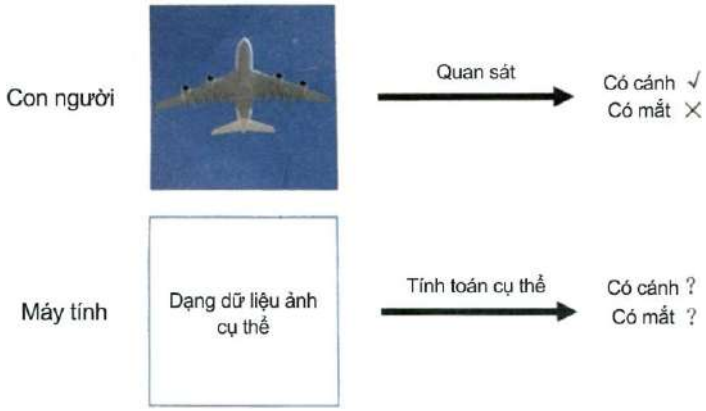
Tổng quan về đặc trưng của hình ảnh

Trước khi chúng ta tìm hiểu các đặc trưng của hình ảnh, chúng ta hãy nghĩ đơn giản: Những đặc trưng nào của ảnh có thể dùng để phân biệt các ảnh với nhau? Ví dụ, trong Bảng 3-1, chúng ta có thể phân biệt giữa chim và mèo bằng cách sử dụng đặc điểm "có cánh hoặc không có cánh" như một đặc trưng. Mèo cũng có thể phân biệt giữa xe hơi và máy bay. Lấy "có mắt hoặc không có mắt" là một tính năng khác, chúng ta hoàn toàn có thể phân biệt bốn loại ảnh này.

	Mèo con	Chim	Máy bay	Xe hơi
Đặc trưng 1: Có cánh hay không	Không	Có	Có	Không
Đặc trưng 2: Có mắt hay không	Có	Có	Không	Không

Bảng 3-1: Phân biệt đặc điểm của 4 loại ảnh

Làm thế nào để trích xuất 2 đặc trưng này từ các bức ảnh ? Đối với con người, quá trình này rất đơn giản, chúng ta có thể thấy các đặc trưng này trong nháy mắt. Nhưng đối với máy tính, hình ảnh là chuỗi dữ liệu được lưu trữ theo một cách cụ thể. Việc máy tính trích xuất các đặc trưng như "có cánh hoặc không có cánh" từ những dữ liệu này thông qua một loạt các tính toán là vô cùng khó khăn (xem *Hình 3-5*).



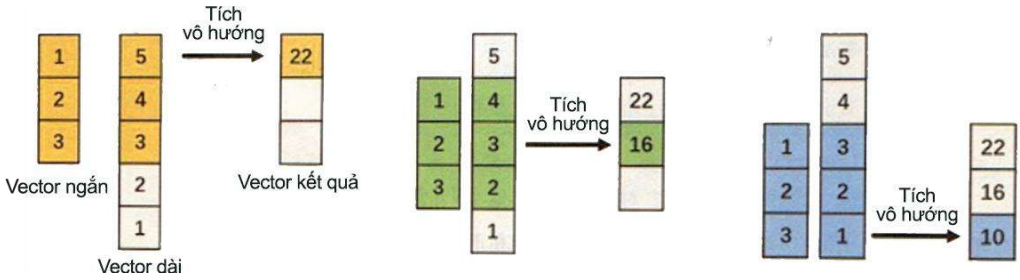
Hình 3-5: Sự khác biệt giữa con người và máy tính trong việc thấy được đặc trưng của ảnh

Trước sự ra đời của học sâu (deep learning), việc thiết kế các đặc trưng cho hình ảnh là một chủ đề nghiên cứu quan trọng trong lĩnh vực thị giác máy tính (computer vision). Trong những ngày đầu, con người phải thiết kế thủ công các đặc trưng hình ảnh khác nhau để có thể mô tả các thuộc tính cơ bản của màu sắc, cạnh (edge), kết cấu (texture)..., kết hợp với các kỹ thuật máy học để có thể giải quyết vấn đề nhận dạng đối tượng (object recognition) và phát hiện đối tượng (object detection).

Vì hình ảnh có thể được biểu diễn như một tensor bậc ba trong máy tính, việc trích xuất đặc trưng từ hình ảnh là quá trình tính toán tensor bậc ba. Một trong những toán tử quan trọng nhất là **tích chập** (convolution).

Phép tích chập

Phép tích chập được sử dụng rộng rãi trong xử lý ảnh và nhiều lĩnh vực khác. Phép tích chập, như cộng, trừ, nhân, chia, là một phép toán. Phép tích chập có thể thực hiện với một vectơ, một ma trận hoặc một tensor bậc ba. Hãy bắt đầu với phép tích chập vectơ, giải thích các bước cơ bản, sau đó tổng quát trên các ma trận và tensor bậc ba.



Hình 3-6: Phép tích chập vector

Kết quả của phép tích chập hai vector vẫn là một vector. Quá trình tính toán được thể hiện trong hình 3-6. Đầu tiên chúng ta căn chỉnh phần tử đầu tiên của hai vectơ và cắt bớt các phần tử phụ trong vector dài, sau đó tính tích vô hướng của hai vectơ có cùng kích thước và sử dụng kết quả tính toán làm phần tử đầu tiên của vectơ kết quả. Tiếp theo trượt vectơ xuống một phần tử, cắt bỏ các phần tử không tương ứng của vectơ dài, tính toán tích vô hướng. Lặp lại quá trình "trượt - cắt bỏ phần thừa - tính tích vô hướng" cho đến khi phần tử cuối cùng của vectơ ngắn được căn chỉnh với phần tử cuối cùng của vectơ dài. Cuối cùng ta có được vector kết quả. Trường hợp đặc biệt khi độ dài của hai vectơ bằng nhau, kết quả phép tích chập chính là kết quả phép tích vô hướng hai vector.

Kiến thức bổ sung: Mô tả toán học về tích chập giữa 2 vector

Quá trình tích chập của 2 vector có thể được mô tả bằng toán học. Giả sử ta có vector a có độ dài m, $a = (a_1, a_2, \dots, a_m)$ và vector b độ dài n $(n \geq m)$, $b = (b_1, b_2, \dots, b_n)$. Vector kết quả của phép tích chập a và b là vector c có độ dài $n - m + 1$, $c = (c_1, c_2, \dots, c_{n-m+1})$. Với mọi $i \epsilon \{1, 2, \dots, n - m + 1\}$, ta có $c_i = \sum_{k=1}^{m} a_k b_{k+i-1} = a_1 b_i + a_2 b_{i+1} + \cdots + a_m b_{i+m-1}$.Ta thường sử dụng ký tự dấu hoa thị " $*$ " để biểu diễn toán tử tích chập, ví dụ: $(1,2,3) * (5,4,3,2,1) = (22,16,10)$.

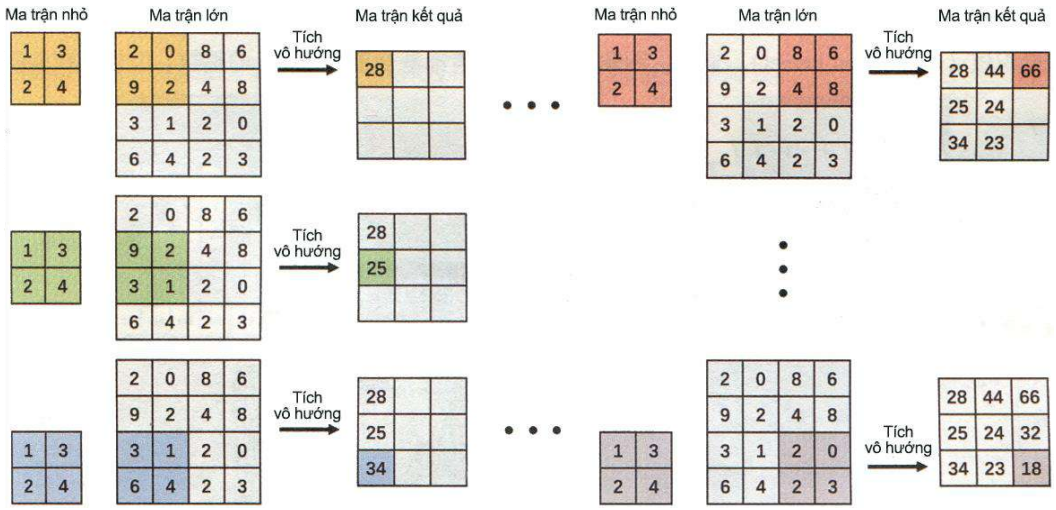
Như có thể thấy từ định nghĩa trên, vector kết quả thường có kích thước nhỏ hơn so với vector dài. Để giữ nguyên độ dài sau khi tích chập, ta có thể thêm một vài số 0 vào 2 đầu vector. Như *Hình 3-6*, ta thêm 2 số 0 vào 2 đầu, vector dài hơn trở thành $(0, 5, 4, 3, 2, 1, 0)$, và vector kết quả giữ nguyên độ dài 5.

Tương tự như vậy, ta định nghĩa tích chập trên ma trận. *Hình 3-7* mô tả tích vô hướng của 2 ma trận. Với 2 ma trận cùng kích thước, tích vô hướng của 2 ma trận là tổng các tích của 2 số có trí tương ứng của hai ma trận.

$$\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \bullet \begin{bmatrix} 0 & 3 \\ 5 & 1 \end{bmatrix} = 1\times0+3\times3+2\times5+4\times1 = 23$$

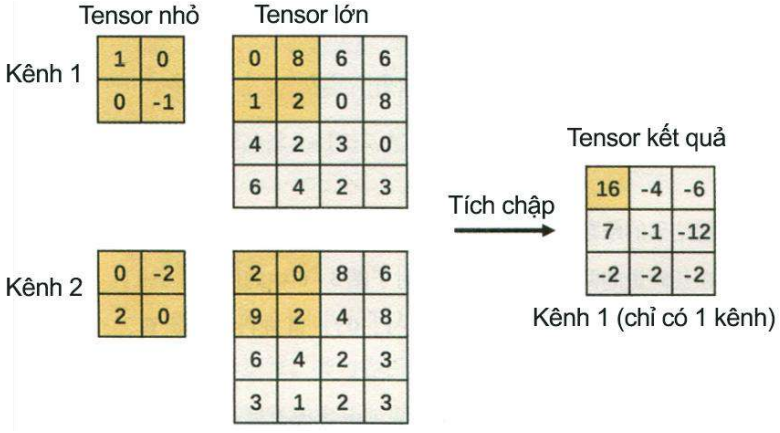
Hình 3-7: Tích vô hướng hai ma trận

Khi tính tích chập 2 vector, ta chỉ cần dịch theo 1 hương (hình 3-6), con khi tính tích chập 2 ma trận, ta cần dịch theo chiều ngang và theo chiều dọc như mô tả ở hình 3-8.



Hình 3-8: Tích chập hai ma trận

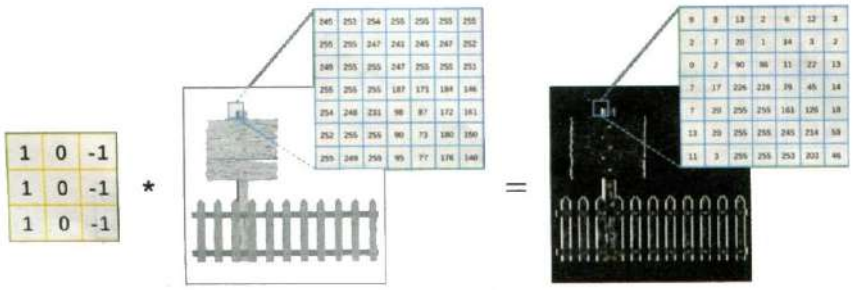
Tương tự, tích chập giữa các tensor bậc 3 được định nghĩa như mô tả *Hình 3-9*. Ở đây trong sách chỉ đề cập đến trường hợp đơn giản, học sinh nào có hứng thú sẽ tự tìm hiểu trường hợp tổng quan. Khi số kênh của 2 tensor bằng nhau, việc dịch chuyển giống với tích chập ma trận theo chiều dài và chiều rộng. Sau đó các kênh được cộng lại với nhau. Kết quả phép tích chập là một tensor bậc 3 có 1 kênh.



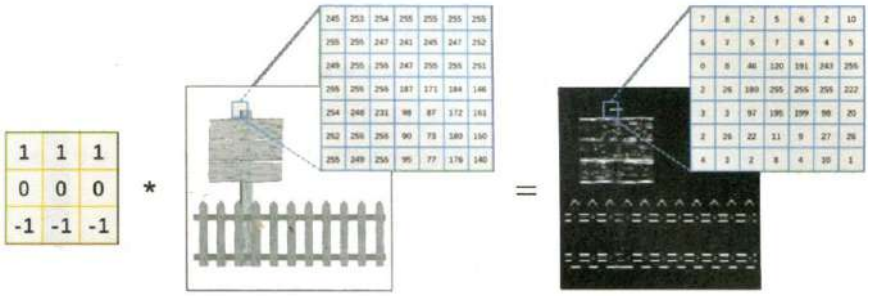
Hình 3-9: Tích chập tensor bậc 3.

Việc dùng phép tích chập để xử lý ảnh được sử dụng rộng rãi, có nhiều phương pháp trích xuất đặc trưng ảnh sử dụng tích chập. Lấy thang độ xám làm ví dụ, chúng ta biết rằng một ảnh đen trắng sử dụng màu trong thang độ xám được biểu diễn dưới dạng ma trận số nguyên. Nếu ta sử dụng một ma trận khác nhỏ hơn và ma trận đó để thực hiện phép tích chập, chúng ta sẽ có một ma trận mới được xem như một ảnh mới. Nói cách khác, bằng phép tích chập, chúng ta có thể biến đổi ảnh gốc thành một ảnh mới. Hình ảnh mới này đôi khi cho thấy một số thuộc tính rõ ràng hơn hình ảnh gốc và ta có thể

sử dụng như đặc trưng của ảnh gốc. Ma trận nhỏ để thực hiện phép tích chập đó gọi là convolution kernel. Thường thì những phần tử trong ma trận ảnh là số từ 0 đến 225, nhưng phần tử trong kernel có thể là một số thực bất kỳ.



Hình 3-10: Trích xuất các cạnh thẳng đứng sử dụng tích chập

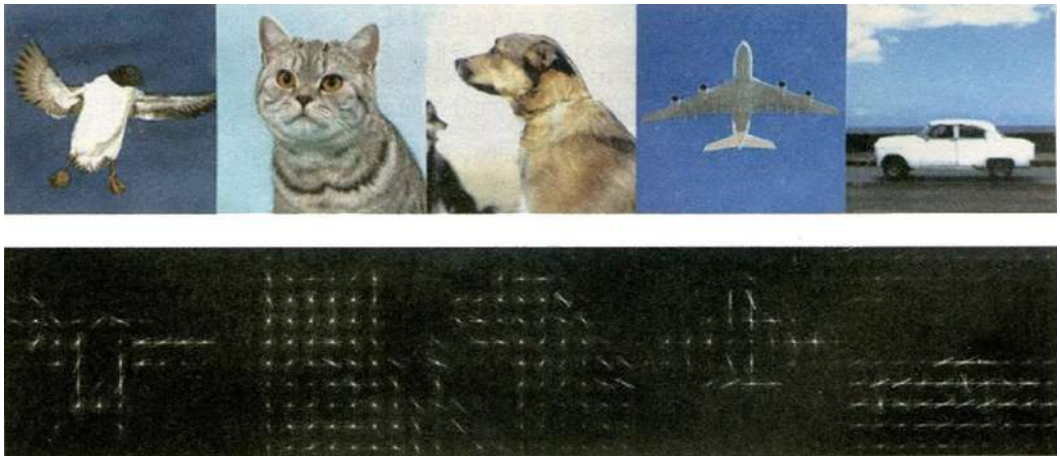


Hình 3-11: Trích xuất các biên ngang sử dụng tích chập

Bằng phép tích chập, ta có thể trích xuất các **đặc trưng biên** (edge features) từ bức ảnh. Biên (edge) trong ảnh là đường phân cách các phần có màu khác nhau, thường biểu thị đường bao quanh của một đối tượng trong ảnh và thường dùng để phát hiện đối tượng. Với biên ngang (horizontal edge), điểm ảnh ở phía trên và phía dưới biên ngang khác nhau đáng kể, biểu thị cho màu sắc khác nhau giữa phần trên và phần dưới. Tương tự như vậy, điểm ảnh ở bên trái và bên phải của biên dọc (vertical edge) cũng có sự khác biệt lớn. Trong ví dụ ở *Hình 3-10*, chúng ta sử dụng convolution kernel kích thước 3×3 gồm ba cột 1, 0, và -1 thực hiện phép tích chập với ảnh gốc để trích xuất các đường biên dọc của ảnh. Trong ví dụ *Hình 3-11*, convolution kernel được sử dụng có kích thước 3×3 gồm 3 dòng 1, 0, và -1 để trích xuất biên ngang. Thực tế, 2 convolution kernel tính toán sự khác biệt giữa các điểm ảnh phía trên và phía dưới, bên trái và bên phải của mỗi vùng có kích thước 3×3 ở ảnh gốc. (Để hiển thị kết quả như một bức ảnh, ta lấy giá trị tuyệt đối của kết quả.). Nhờ đó, ta có thể trích xuất các đặc trưng biên khác nhau từ ảnh.

Những nghiên cứu sâu hơn đã thiết kế một vài đặc trưng phức tạp và hiệu quả hơn. Histogram of oriented gradients - HOG là một dạng biểu đồ sử dụng như một đặc trưng cổ điển, được ứng dụng hiệu quả trong phát hiện và nhận dạng đối tượng. HOG sử dụng kỹ thuật nhận dạng đường biên và một vài phương pháp thống kê để biểu diễn đường biên của đối tượng trong một bức ảnh. Những đối tượng khác nhau có đường biên khác nhau, chúng ta có thể sử dụng HOG để phân biệt các đối tượng khác nhau trong bức ảnh (ví dụ *Hình 3-12*).

Quá trình trích xuất của HOG được chia làm hai bước chính. Đầu tiên, ta thực hiến phép tích chập để trích xuất đặc trưng biên của ảnh. Sau đó, ta chia bức ảnh thành nhiều phần và đo đặc trưng biên dựa trên hướng và biên độ để tạo thành một biểu đồ. Cuối cùng là kết hợp tất cả các ô vuông trong mỗi phần để tạo thành vector đặc trưng. Quá trình cụ thể khá phức tạp nên cuốn sách sẽ không đề cập đến ở đây.

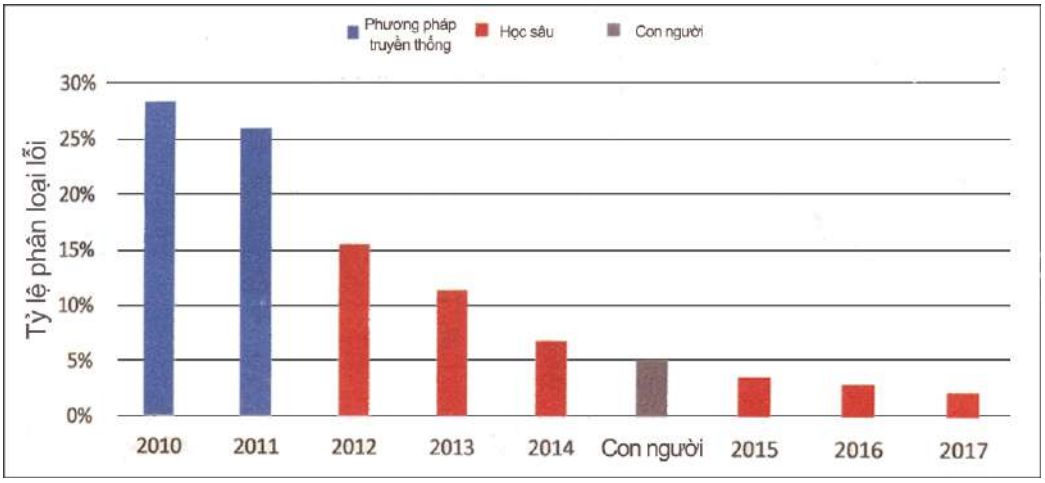


Hình 3-12: HOG của các đối tượng có hình dạng khác nhau

3.2. Phân loại ảnh dựa trên Deep Neural Network

Qua phần trc, ta đã thấy đc cách sử dụng HOG và bộ phân loại SVM để hoàn thành nv phân loại ảnh. Tuy nhiên, tỉ lệ chính xác của việc phân loại chưa đạt yêu cầu. Thực tế có một vấn đề trong lĩnh vực thị giác máy tính tại thời điểm này: Với những đặc tính hình ảnh được thiết kế nhân tạo, độ chính xác của phân loại ảnh chỉ đạt đến một ngưỡng nhất định.

Image Net Challenge là một cuộc thi quốc tế trong lĩnh vực máy tính, một trong những nhiệm vụ của cuộc thi là để máy tính tự động phân loại 1000 bức ảnh. Trong Image Net Challenge đầu tiên năm 2010, nhóm vô địch đã sử dụng hai đặc trưng được thiết kế thủ công với SVM đạt tỷ lệ phân loại lỗi là 28,2%. Trong cuộc thi năm 2011, tỷ lệ lỗi phân loại đã giảm xuống 25,7% nhờ thiết kế đặc trưng tốt hơn. Tuy nhiên đối với con người, "hệ thống trí tuệ nhân tạo" như vậy chưa được gọi là "thông minh". Nếu sử dụng cùng một bộ dữ liệu, tỷ lệ phân loại lỗi của con người chỉ là 5,1%, thấp hơn 20% so với hệ thống phân loại tiên tiến nhất (như trong Hình 3-13).



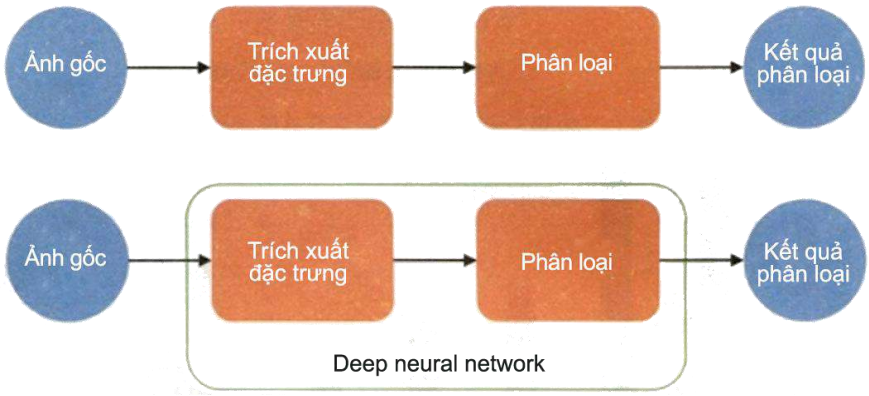
Hình 3-13: Kết quả Image Net Challenge qua các năm

Liệu chúng ta có thể cố gắng tiếp tục với các đặc trưng ảnh tốt hơn ? Có thể. Nhưng công việc đó đòi hỏi các nhà khoa học và kỹ sư cùng với chuyên môn và sự sáng tạo trong lĩnh vực để khám phá và thử nghiệm trong nhiều năm, thậm chí cần may mắn để tạo ra sự đột phá. Sự khó khăn trong việc thiết kế đặc trưng làm chậm đi đáng kể sự phát triển của thị giác máy tính. Tuy nhiên, năm 2012 cuộc thi Image Net đem lại sự ngạc nhiên. Đội thuộc đại học Toronto sử dụng **Deep Neural Network** (DNN – Mạng nơron sâu) lần đầu tiên, làm giảm tỉ lệ sai của việc phân loại đi 10%, khiến tỉ lệ chính xác đạt 84.7%. Kể từ đó, Image Net là sân chơi của cuộc thi về DNN. Chỉ 3 năm sau, đội của Microsoft Research đưa ra kiến trúc mạng mới khiến tỉ lệ sai còn 4.9%, vượt qua sự chính xác của con người lần đầu tiên. Năm 2017, tỉ lệ sai đạt 2.3%. Đây là năm cuối cùng của cuộc thi Image Net, vì DNN đã giải quyết vấn đề phân loại ảnh tốt hơn.

DNN có khả năng mạnh mẽ như vậy vì nó có thể học tự động một cách hiệu quả các đặc tính từ các bức ảnh. Trong nhiệm vụ phân loại ảnh, thường rất khó để trực tiếp diễn tả khái niệm trừu tượng cấp cao như “có cánh hoặc không có cánh” hoặc “có mắt hoặc không có mắt”. Tuy nhiên, sau

sự xuất hiện của DNN, điều đó hoàn toàn khả thi. Trong các lĩnh vực khác nhau của thị giác máy tính, các đặc điểm mà DNN học dần dần thay thế các đặc điểm thiết kế thủ công, và AI trở nên thông minh hơn.

Mặt khác, sự xuất hiện của DNN cũng làm giảm sự phức tạp của hệ thống AI. Hình 3-14 mô tả hệ thống phân loại kiểu mẫu truyền thống, trích xuất đặc trưng và phân loại là 2 bước độc lập, và DNN tích hợp 2 quá trình đó lại với nhau. Chúng ta chỉ cần input một bức ảnh cho mạng neural và ta có thể trực tiếp dự đoán danh mục của ảnh, không cần hoàn thiện từng bước trích xuất đặc trưng và phân loại. Từ quan điểm này, DNN không lật đổ hệ thống phân loại kiểu mẫu truyền thống mà cải thiện hệ thống truyền thống.



Hình 3-14: Sự khác biệt giữa DNN và mô hình hệ thống phân loại truyền thống

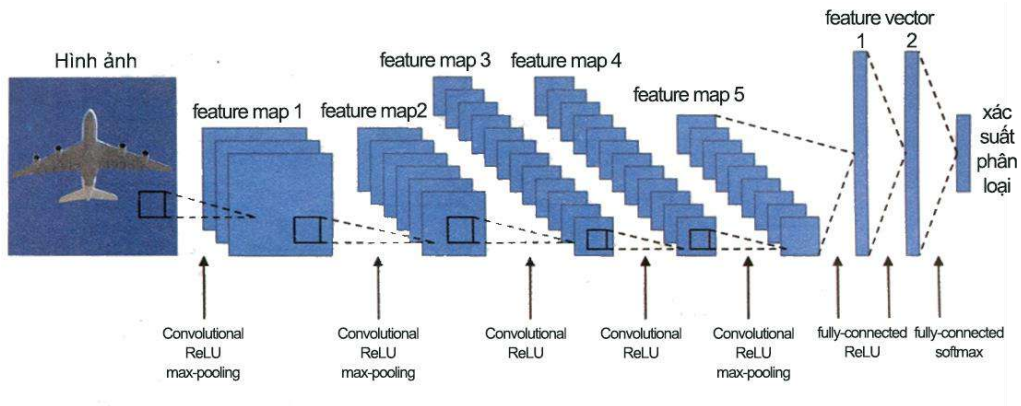
Cấu trúc DNN

Một DNN thường gồm nhiều lớp (layer) kết nối với nhau thành chuỗi. Lớp đầu tiên thường lấy ảnh làm input, trích xuất đặc trưng từ ảnh thông qua các phép toán nhất định. Các đặc trưng trích xuất từ lớp trước của mỗi lớp được biến đổi sang dạng cụ thể để có được đặc trưng phức tạp hơn. Quá trình trích xuất đặc trưng thứ bậc như vậy có thể được tích lũy, khiến mạng nơ ron có khả năng trích xuất đặc trưng mạnh mẽ. Sau nhiều lớp biến đổi, mạng nơ ron có thể biến đổi ảnh gốc thành các đặc trưng trừu tượng cấp cao.

Quá trình trình trừu tượng từ đơn giản đến phức tạp, từ thấp đến cao, có thể được trải nghiệm thông qua các ví dụ trong cuộc sống. Ví dụ trong quá trình học tiếng anh, tổng hợp các chữ cái ta được các từ, tổng hợp các từ có thể tạo ra câu, thông qua phân tích các câu, ta có thể hiểu nghĩa, thông qua phân tích nghĩa, ta có thể biết được tư tưởng hoặc mục đích diễn đạt. Các kiểu ngữ nghĩa, tư tưởng… như vậy là sự trừu tượng cấp cao hơn.

Tiếp theo hãy quan sát một mạng nơ ron cụ thể để có cảm nhận trực quan về cấu trúc DNN. Các khái niệm lớp convolutional, lớp ReLU nonlinear activation, lớp pooling, lớp fully connected, lớp softmax xuất hiện trong mạng này sẽ được giới thiệu lần lượt.

Trong hình 3-15 là mạng nơ ron Alex Net đã chiến thắng Image Net Challenge 2012. Phần chính của mạng này gồm 5 lớp convolutional và 3 lớp fully connected. 5 lớp convolutional nằm ở phía trước của mạng, chuyển đổi hình ảnh để trích xuất các đặc trưng. Sau mỗi lớp convolutional là lớp ReLU nonlinear activation thực hiện việc biến đổi phi tuyến. Lớp convolutional đầu tiên, thứ 2 và thứ 5 được kết nối với lớp max-pooling để giảm độ phân giải của feature map. Sau các lớp convolutional, connected nonlinear activation và pooling, feature map được biến đổi thành vector đặc trưng kích thước 4096. Vector này tiếp tục được biến đổi thành eigenvector cuối cùng sau hai sự biến đổi của hai lớp fully connected và ReLU. Sau khi vượt qua thêm lớp fully connected và lớp chuẩn hóa softmax, ta thu được dự đoán về loại của bức ảnh.



Hình 3-15: Sơ đồ cấu trúc mạng nơ ron Alex Net

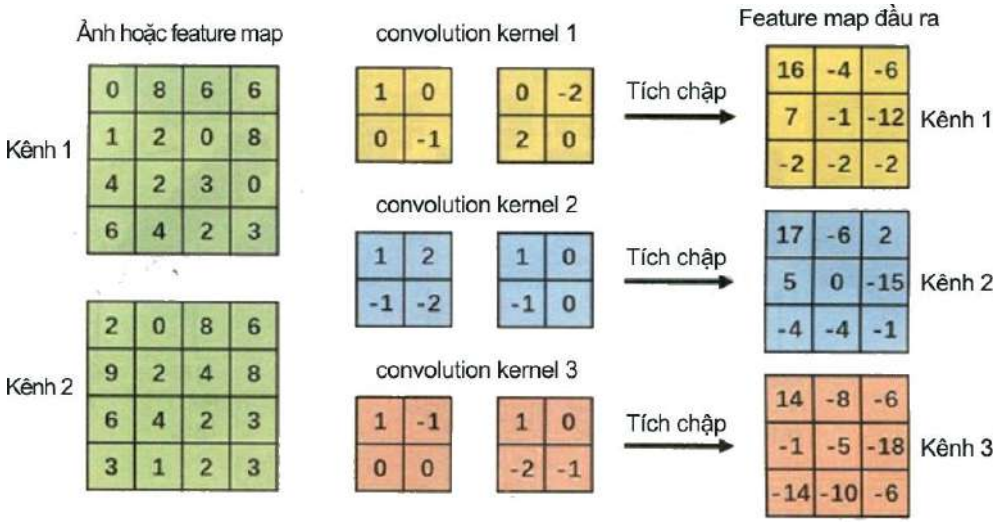
Lớp Convolutional

Lớp convolutional được sử dụng phổ biến bởi DNN trong xử lý ảnh. Khi một DNN chủ yếu được tạo thành bởi lớp convolutional, ta gọi đó là **Convolutional Neural Network** (CNN)

Lớp convolutional là lớp biến đổi ảnh gốc hoặc feature map từ lớp phía trước. Trong phần trước chúng ta đã tìm hiểu về trích xuất đặc trưng biên, một convolution kernel cụ thể có thể thực hiện một biến đổi hình ảnh nhất định để trích xuất một đặc trưng nhất định như biên dọc hoặc biên ngang. Trong lớp convolutional, để trích xuất nhiều dạng đặc trưng từ một

bức ảnh, ta thường sử dụng nhiều convolution kernel thực hiện các phép tích chập khác nhau trên ảnh đầu vào (*Hình 3-16*). Với một convolution kernel ta có thể thu được một tensor bậc 3 có 1 kênh; với nhiều convolution kernel ta có thể thu được một tensor bậc 3 có nhiều kênh. Bằng cách kết hợp các kết quả như những kênh khác nhau, ta có được một tensor bậc 3 mới. Số kênh của tensor bậc 3 này bằng với số lượng convolution kernels được sử dụng. Vì mỗi kênh là một đặc trưng trích xuất từ ảnh gốc, chúng ta gọi tensor bậc 3 này là feature map. Feature map là đầu ra cuối cùng của lớp convolutional.

Cả feature map và ảnh màu là tensor bậc 3 và có một vài kênh. Do đó, lớp convolutional có thể hoạt động cả trên ảnh gốc và feature map của lớp khác. Thông thường, lớp convolutional đầu tiên lấy ảnh gốc làm đầu vào, còn những lớp convolutional tiếp theo lấy feature map của lớp trước đó làm đầu vào.



Hình 3-16: Trích xuất nhiều đặc trưng bằng nhiều convolution kernel và kết hợp chúng thành feature map nhiều kênh.

Lớp fully connected

Trong nhiệm vụ phân loại ảnh, ảnh đầu vào được chuyển đổi thành vector đặc trưng sau khi vượt qua một vài lớp convolutional. Nếu cần biến đổi vector đặc trưng này, chúng ta thường sử dụng lớp fully connected.

Trong lớp fully connected, chúng ta sử dụng một vài vector cùng kích thước để thực hiện phép tích vô hướng với vector đầu vào, sau đó nối tất cả kết quả lại thành vector đầu ra. Cụ thể, nếu lớp fully connected có đầu vào là vector X, ta sử dụng tổng cộng K vector tham số W_k và thực hiện tích vô

hướng với X, sau đó cộng thêm giá trị b_k vào kết quả: $y_k = X \cdot W_k + b_k$. Cuối cùng, ta có K giá trị kết quả y_k và thêm vào vector Y là vector đầu ra của lớp.

Lớp softmax

Lớp softmax thực hiện công việc tính hàm mũ chuẩn hóa trong bộ phân loại tuyến tính nhiều lớp. Cụ thể với vector đầu vào $X = (x_1, x_2, \dots, x_n)$, tính n giá trị $y_k = \frac{e^{x_k}}{e^{x_1} + \cdots + e^{x_n}}$ và nối vào vector đầu ra $Y = (y_1, y_2, \dots, y_n)$. Lớp softmax thường là lớp cuối cùng của mạng phân loại. Nó lấy vector đặc trưng và số lượng danh mục phân loại làm đầu vào (vector đặc trưng này thường là đầu ra của một lớp fully-connected) và có đầu ra là xác suất của mỗi danh mục thể loại mà bức ảnh thuộc về.

Lớp nonlinear activation

Thông thường chúng ta cần kết nối một lớp non-linear activation tới mỗi lớp convolutional và fully connected. Tại sao vậy ? Trong thực tế, các tính toán được thực hiện ở lớp convolutional hay fully connected đều là hàm của biến độc lập, được gọi là hàm tuyến tính. Hàm tuyến tính có một tính chất: Kết hợp của các phép tính tuyến tính vẫn là tuyến. Nói cách khác, nếu chúng ta đơn giản chỉ chồng hai lớp convolutional và lớp fully connected trực tiếp, hiệu ứng trên ảnh đầu vào sẽ được thay thế bởi lớp fully connected. Như vậy, mặc dù chúng ta sử dụng rất nhiều lớp, việc biến đổi ảnh của các lớp sẽ được nhập vào với nhau. Nếu chúng ta thực hiện phép tính phi tuyến (non-linear) sau mỗi phép tuyến tính, hiệu ứng biến đổi sẽ được bảo toàn. Có rất nhiều kiểu lớp nonlinear activation. Kiểu cơ bản là chọn một hàm phi tuyến, sau đó sử dụng hàm này với mỗi phần tử của feature map hoặc feature vector đầu vào để có được kết quả. Các hàm phi tuyến thường được sử dụng là:

- Hàm logic (logistic function) (*Hình 3-17* trái):

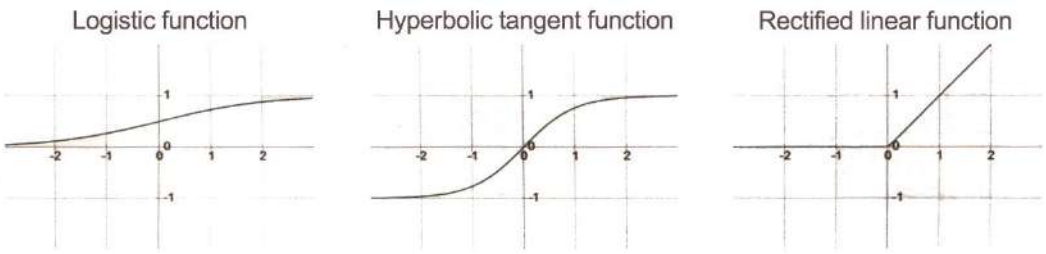
$$s(x) = \frac{1}{1 + e^{-x}}$$

- Hàm tiếp tuyến hyperbol (hyperbolic tangent function) (*Hình 3-17* giữa)

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

- Rectified linear function (*Hình 3-17* phải)

$$\mathrm{ReLU}(x) = \begin{cases} 0, x < 0 \\ x, x \geq 0 \end{cases}$$



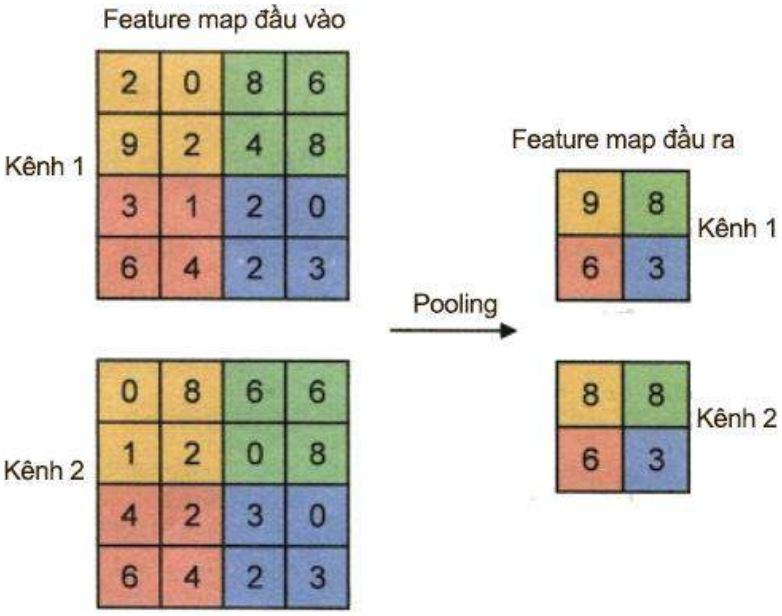
Hình 3-17: Các hàm nonlinear activation khác nhau

Lấy một lớp nonlinear activation (lớp ReLU) gồm một hàm linear rectification làm ví dụ, với đầu vào là vector đặc trưng hoặc feature map, những phần tử nhỏ hơn 0 sẽ được bằng 0 và giữ nguyên giá trị các phần tử còn lại. Vì sự tính toán của ReLU rất đơn giản, thường tốc độ tính toán của lớp này sẽ nhanh hơn nhiều các lớp khác, đồng thời trong thực tế đem lại hiểu quả cao, vì vậy lớp này được sử dụng rộng rãi trong các DNN.

Lớp Pooling

Khi tính tích chập, chúng ta sử dụng convolution kernel để trượt qua mỗi điểm ảnh của ảnh gốc hoặc feature map. Nếu bức ảnh hay feature map có độ phân giải lớn, việc tính toán ở lớp convolutional trở nên nặng nề. Để giải quyết vấn đề này, chúng ta thường chèn thêm lớp pooling vào sau một vài lớp convolutional để giảm độ phân giải của feature map.

Các bước thực hiện trong lớp pooling như sau: Đầu tiên chia các kênh của feature map thành các ma trận. Với mỗi ma trận, chúng ta cắt thành các hình vuông kích thước bằng nhau. Ví dụ Hình 3.18, ta chia ma trận 4x4 thành 4 hình vuông, mỗi hình có kích thước 2x2. Sau đó ở mỗi hình vuông nhỏ, ta lấy ía trị lớn nhất hoặc giá trị trung bình và tổng hợp kết quả thành ma trận mới. Cuối cùng ta ghép các ma trận kết quả thành các kênh để hình thành một tensor bậc 3, đây là đầu ra của lớp pooling layer. Lớp pooling lấy giá trị lớn nhất của mỗi hình vuông nhỏ được gọi là lớp max pooling, lớp lấy giá trị trung bình được gọi là lớp average pooling.



Hình 3-18: Sơ đồ lớp max pooling

Trong *Hình 3-18*, sau khi được tổng hợp, chiều dài và chiều rộng của feature map được giảm 1/2, số lượng phần tử giảm 1/4. Thông thường chúng ta sẽ thêm lớp pooling vào sau lớp convolutional. Sau một vài kết hợp của hai lớp convolution và pooling, độ phân giải của feature map nhỏ hơn nhiều so với độ phân giải của ảnh gốc bất kể số kênh, lượng tính toán và các tham số được giảm đi đáng kể.

Mạng nơron nhân tạo và mạng thần kinh sinh học

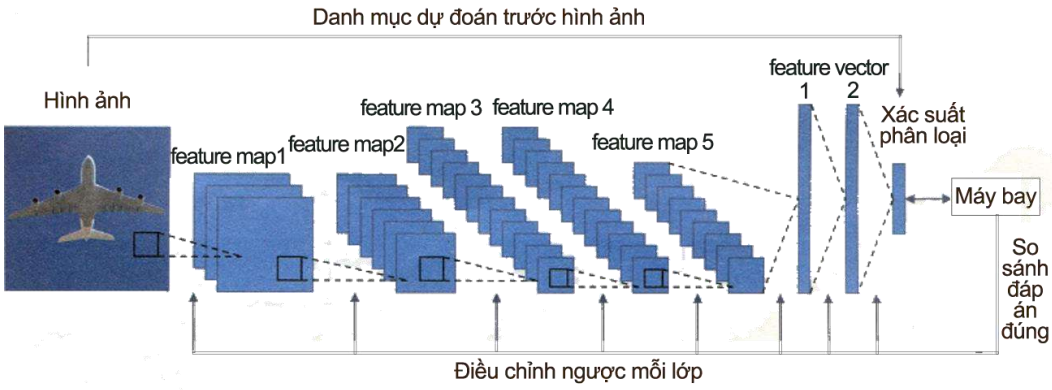
Mạng nơron nhân tạo ban đầu được lấy cảm hứng từ mạng thần kinh sinh học. Mạng nơron sinh học được tạo thành từ hàng trăm triệu tế bào thần kinh (neural) kết nối với nhau. Khi chúng ta suy nghĩ hoặc phản ứng với các kích thích bên ngoài, các nơron giao tiếp với nhau. Tế bào thần kinh nhân tạo là một mô hình toán học của các tế bào thần kinh sinh học. Sử dụng nơron nhân tạo làm đơn vị cơ bản, chúng ta có thể xây dựng các lớp convolutional, fully connected, ReLU,..., và sau đó xây dựng các mạng nơron nhân tạo. Các kết nối giữa các phần tử của feature map hoặc vector đặc trưng cũng được gọi là nơron và giá trị của phần tử được gọi là phản hồi của nơron.

Tuy nhiên, các tế bào thần kinh nhân tạo chỉ là một mô hình toán học, không mô tả chính xác hành vi phức tạp của các tế bào thần kinh sinh học. Trong lĩnh vực máy học, trọng tâm của mạng nơron nhân tạo chủ yếu giới hạn trong các nhiệm vụ trí tuệ nhân tạo cụ thể. Trong các ứng dụng thực tế, không

có mối liên hệ trực tiếp nào giữa các mạng thần kinh nhân tạo chính thống và mạng thần kinh sinh học.

Huấn luyện mạng thần kinh nhân tạo

Các bộ phân loại cần được huấn luyện để phân biệt các vectơ đặc trưng thuộc các loại khác nhau, DNN cũng cần được huấn luyện để học các tính năng hình ảnh hiệu quả. Huấn luyện về bản chất là quá trình tìm kiếm các tham số tốt nhất. Trong một bộ phân loại tuyến tính, các tham số chứa tất cả các hệ số của các hàm tuyến tính. Trong mạng nơron, các giá trị phần tử của các convolution kernel trong lớp convolutional, các hệ số của phép tích vô hướng trong lớp fully connected đều là các tham số. Để chia vector hai chiều của hoa thành hai loại, chúng ta chỉ cần huấn luyện ba tham số. Trong Alex Net, có tới gần 60 triệu tham số để học, điều này khó hơn nhiều so với việc huấn luyện các bộ phân loại tuyến tính. Với vấn đề huấn luyện mạng nơron, các nhà khoa học trí tuệ nhân tạo đã đề xuất một thuật toán backpropagation (Hình 3-19), một trong những phương tiện hiệu quả nhất.



Hình 3-19: Sơ đồ thuật toán Back propagation

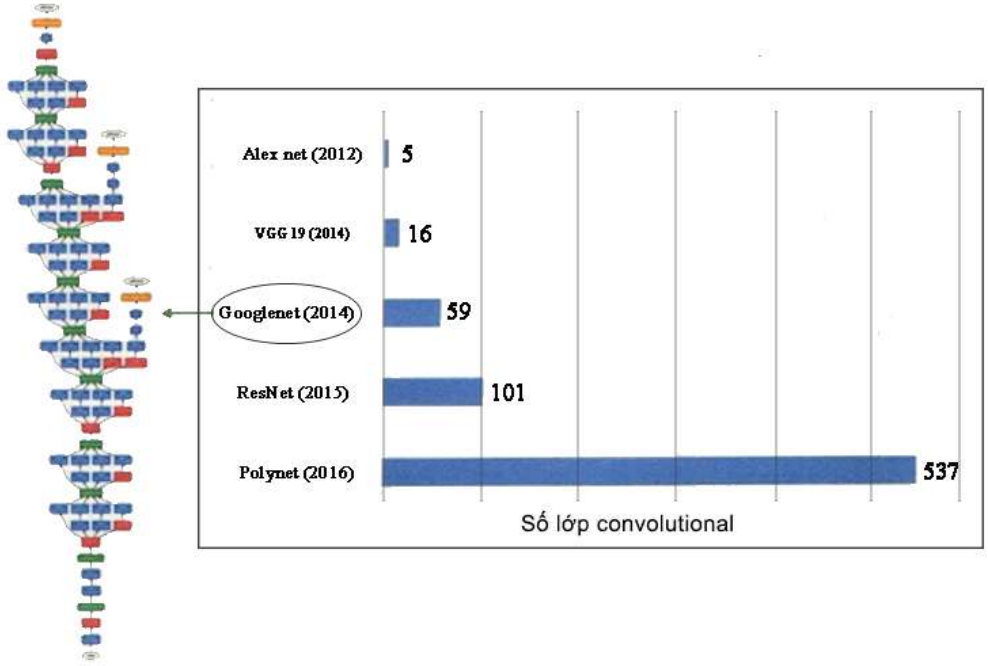
Khi huấn luyện, chúng ta đưa ảnh đầu vào vào mạng, sau tính toán của từng lớp, ta có được xác suất dự đoán của mỗi loại. Chúng ta so sánh câu trả lời đúng với kết quả, nếu kết quả dự đoán không đủ tốt, ta sẽ tiến hành điều chỉnh tham số theo từng lớp, bắt đầu từ lớp cuối cùng đến lớp đầu tiên, như vậy mạng có thể đưa ra dự đoán tốt hơn. Chúng ta gọi phương pháp thay đổi tham số từ phía sau ra phía trước như vậy là thuật toán back propagation. Thuật toán điều chỉnh cụ thể liên quan đến kiến thức phức tạp hơn như chain rule và stochastic gradient descent sẽ không được mô tả chi tiết ở đây.

3.3. Sự phát triển và những thách thức của DNN

“Deep”

Từ “deep” (nghĩa là “sâu”) trong deep learning - học sâu đại diện cho số lớp và số tham số của mô hình . Mô hình càng nhiều mẫu, không gian học và điều chỉnh càng lớn, khả năng biểu đạt mạnh mẽ hơn. Một vài nhận định được đưa ra: “Miễn là tỉ lệ lỗi giảm dần, bạn có thể liên tục tăng số lượng lớp trong DNN để cải thiện kết quả.”

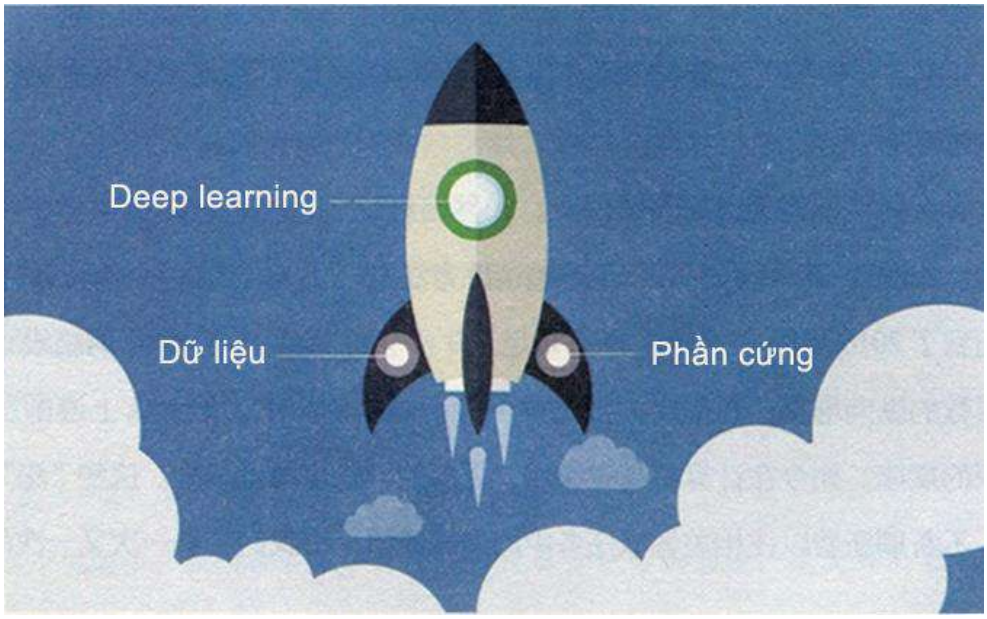
Sự cải thiện nhanh chóng về hiệu suất của DNN, cùng với sự phức tạp của cấu trúc mạng tăng lên, dẫn đến việc tăng số lượng các lớp. Như hình 3-20, năm 2012 mạng Alex Net vượt qua phương pháp truyền thống 10% chỉ có 5 lớp convolution, nhưng đến 2016 mạn PolyNet có hơn 500 lớp convolution. Mặc dù mô hình mạng hiện đại không đơn thuần chỉ là ghép các lớp với nhau, số lượng các lớp không có nghĩa là độ sâu của mạng, nhưng xét về tổng quan mạng có càng nhiều lớp thì càng sâu. Ngày nay, trong lĩnh vực thị giác máy tính, sự xuất hiện của các mạng sâu hơn nữa không phải là hiếm. Những mạn sâu và phức tạp này sẽ tiếp tục đạt được những kết quả tốt hơn nữa và gây kinh ngạc cho chúng ta.



Hình 3-20: Mạng cổ điển và lớp convolutional của nó qua các năm

“Sự giúp đỡ” sâu

Không có sự thành công nào của học sâu mà không có lý do, thực tế nó liên quan đến cả sự tích lũy, đổi mới, phát triển của bản thân và liên quan chặt chẽ đến các động lực bên ngoài. Ở đây chúng ta làm nổi bật hai nhân tố: Dữ liệu và sức mạnh tính toán.



Hình 3-21: Tăng cường cho học sâu: Dữ liệu và sức mạnh tính toán

Dữ liệu

Với sự ra đời của cuộc cách mạng thông tin, dữ liệu trên mạng bùng nổ theo cấp số nhân. Sự xuất hiện của lượng lớn dữ liệu số gây khó khăn cho các phương pháp phân tích và xử lý dữ liệu truyền thống. Tuy nhiên thách thức này đem lại cơ hội tuyệt vời cho sự phát triển của học sâu.

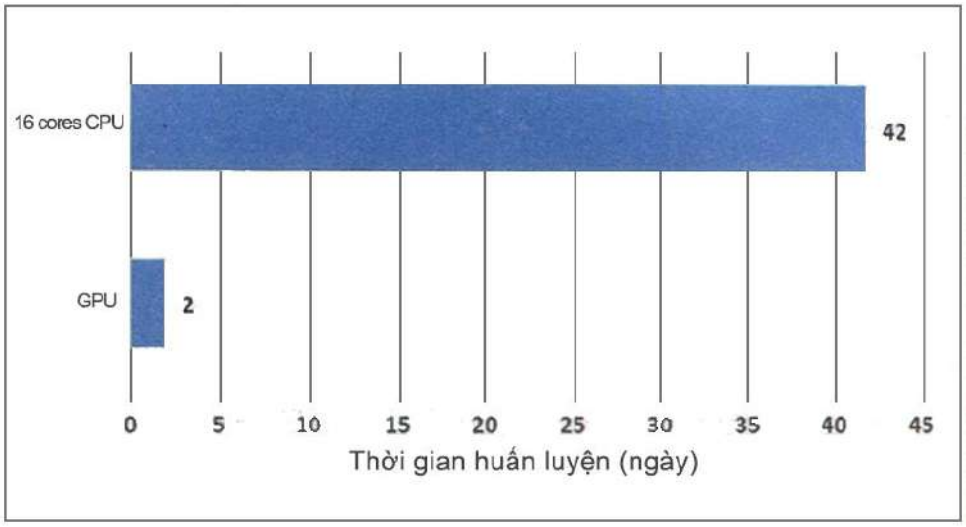
Cùng với sự phát triển của khoa học dữ liệu, hiệu suất mô hình sâu gắn bó chặt chẽ với tổng số lượng dữ liệu huấn luyện và sự đa dạng. Một mô hình có hiểu biết sâu thường xuất sắc hơn trong việc xử lý vấn đề thực tế. Có thể ví dữ liệu như nhiên liệu để lái tên lửa deep learning tiến về phía trước.

Khả năng tính toán

Mặc dù các nhà khoa học AI có ý tưởng rất tốt về DNN, tuy nhiên họ vẫn cần tài nguyên phần cứng để hỗ trợ. Đặc biệt, quá trình huấn luyện mạng

nơ ron yêu cầu một lượng lớn tài nguyên để tính toán, mạng càng sâu và càng phức tạp thì càng cần nhiều tài nguyên hơn.

Nhiệm vụ tính toán nặng nề này gây khó khăn cho các CPU bình thường. Cùng lúc đó, sự xuất hiện và liên tục cập nhật của những bộ xử lý đồ họa (GPU) mạnh mẽ hơn đã góp phần tạo nên học sâu ngày hôm nay. Alex Net, đội tiên phong trong DNN, để huấn luyện được toàn bộ mô hình phân loại Image Net đã tốn hơn một tháng để sử dụng hoàn toàn 16 core CPU, chỉ 2/3 sử dụng GPU mới. Các ngày cải thiện đáng kể hiệu quả huấn luyện (*Hình 3 - 22*).



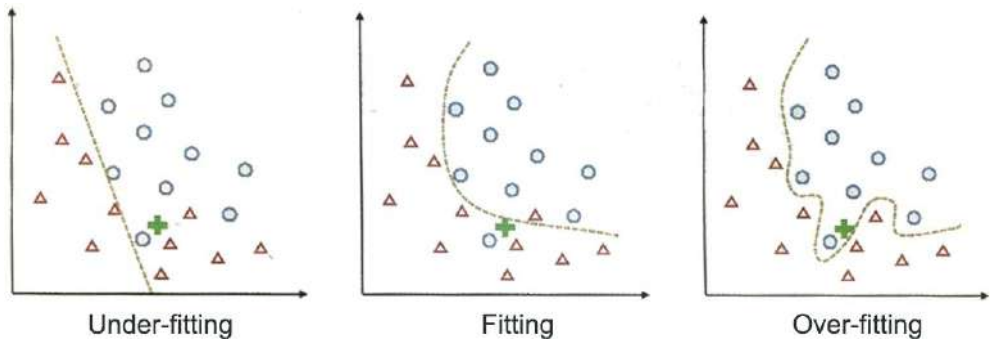
Hình 3-22: So sánh thời gian huấn luyện Alex Net trên các thiết bị phần cứng khác nhau

“Sự khó khăn” sâu

Nếu chúng ta có được mô hình với hiệu suất tốt hơn chỉ đơn giản bằng cách làm mạng sâu hơn thì tất cả các nghiên cứu trong lĩnh vực học sâu trở nên rất dễ dàng. Chỉ giải quyết vấn đề bằng cách liên tục tăng độ sâu của mạng? Tuy nhiên, tình huống thực sự không đơn giản như ta nghĩ. Sự thật thì mạng nơ ron sâu hơn không chỉ mang đến sự tiêu thụ tài nguyên khổng lồ mà hiệu suất của nó với các nhiệm vụ tương ứng còn có thể giảm. Tại sao điều đó lại xảy ra? Sau đây là 2 lý do quan trọng.

Không phù hợp và phù hợp quá mức (under-fitting và over-fitting)

Quá nhiều lớp đem đến quá nhiều tham số, điều mà có thể dẫn đến vấn đề phổ biến trong máy học: over-fitting. Quá trình huấn luyện mô hình được thực hiện trên tập dữ liệu huấn luyện và sự đánh giá về hiệu suất được thực hiện trên tập dữ liệu kiểm tra. Một vài mô hình tỏ ra rất xuất sắc trên tập huấn luyện, nhưng hiệu suất trên tập kiểm tra không đạt yêu cầu, thậm chí có kết quả kém. Hiện tượng một mô hình phức tạp "phục vụ" quá mức tập huấn luyện và khiến nó thực hiện kém trên lượng dữ liệu mới được gọi là overitting. Nó giống như khi ta học giải các bài toán, thỉnh thoảng ta gặp các vấn đề, các câu hỏi lạ, thậm chí tham khảo cả những câu trả lời sai, nhưng chúng ta muốn học vẹt, ghi nhớ lời giải của các bài toán này và máy móc áp dụng vào các bài toán thông thường gây ra kết quả sai. Mô hình under-fitting giống như một học sinh dốt: Hiệu suất trên dữ liệu huấn luyện và dữ liệu mới không đạt yêu cầu, nói ngắn gọn là khả năng có giới hạn. Hiệu tương mô hình quá đơn giản dẫn đến tỉ lệ chính xác thấp trong quá trình huấn luyện và rất khó để cải thiện và có hiệu suất thấp trên dữ liệu mới được gọi là under-fiting.



Hình 3-23: Under-fitting, fitting và over-fitting

Trong nhiệm vụ phân loại ảnh, over-fitting có thể hiểu là khả năng của mô hình quá mạnh. Trong quá trình huấn luyện, không chỉ các đặc điểm thật sự của mẫu được phân loại trong tập dữ liệu huấn luyện được nhớ, mà còn rất nhiều thông tin nhiễu được ghi lại theo. Điều đó dẫn đến mô hình với độ chính xác cao, thậm chí đạt 100% với tập huấn luyện. Tuy nhiên, khi thêm thông tin mới vào, kết quả của mô hình thường sai. Như được chỉ ra ở hình 3-23, dữ liệu mới (chữ thập màu xanh) thuộc phân loại của tam giác đỏ, nhưng bị phân loại nhầm thành vòng tròn xanh. Còn mô hình under-fitting là do quá ít tham số, không thể trích xuất toàn bộ các đặc trưng của đối tượng, vì thế nó chỉ có thể tách rời các dữ liệu, độ chính xác khi phân loại tương đối thấp.

Vậy làm cách nào để tránh over-fitting khi đang làm sâu kiến trúc mạng và tăng khả năng biểu đạt của mạng ? Phương pháp regularization có

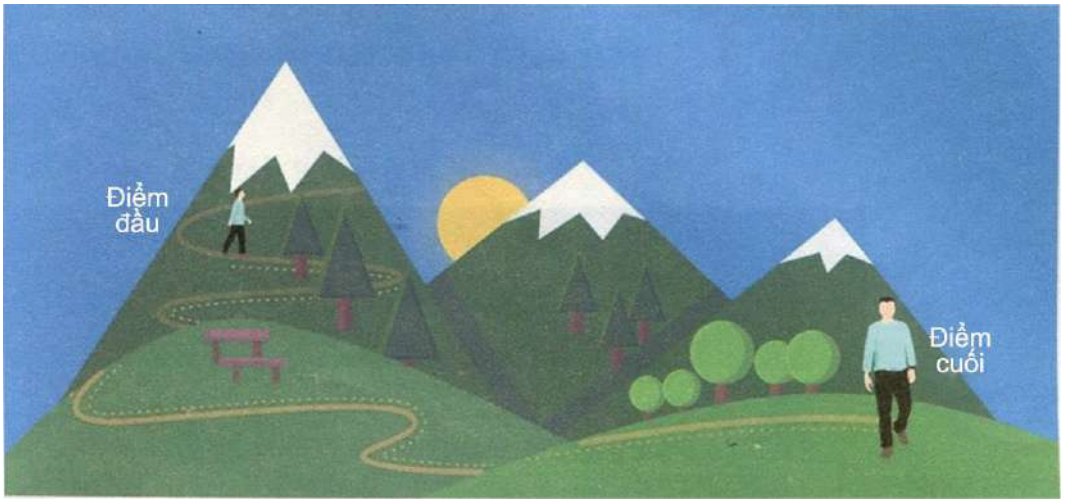
tên gọi weight decay hoặc các phương pháp regularization khác trong mạng nơ ron sẽ giải quyết vấn đề này. Những chi tiết kỹ thuật tương ứng không đề cập đến ở đây, những ai có hứng thú có thể tự tìm hiểu thêm.

Vanishing gradient

Chúng ta đã giải quyết được vấn đề over-fitting gây ra bởi việc làm sâu mạng tới mức độ nhất định, nhưng phát hiện ra các lớp đơn giản vẫn làm khả năng của mô hình giải quyết vấn đề không tăng cũng k giảm. Vì khả năng bị giảm này không liên quan đến overfitting, vậy lý do ở đây là gì ?

Trong thực tế, một ngăn xếp các lớp đơn giản có thể dẫn đến hiện tượng ảnh hưởng đến hiệu suất: mất mát đạo hàm (gradient vanish). Vậy gradient là gì và tại sao nó lại bị mất mát ?

Chương 2 đã giới thiệu ngắn gọn về khái niệm tối ưu hóa. Mục đích tối ưu hóa để làm cho giá trị tiên đoán chính xác hơn. Huấn luyện mạng sâu cũng là một quá trình tối ưu hóa. Hãy quan sát một mạng thực hiện một công việc cụ thể. Nếu quá trình huấn luyện mạng giống như xuống đồi, quá trình tối ưu hóa tương đương với việc tìm điểm thấp nhất. Gradient tương đương với hướng tối ưu của mỗi bước đi (*Hình 3-24*).



Hình 3-24: Sơ đồ xuống dốc

Nguồn trực tiếp nhất của sự hướng dẫn này là sự khác biệt giữa đầu ra của mô hình và đầu ra mục tiêu, chính là lỗi. Thuật toán backpropagation đã đề cập đến sự thay đổi các tham số của mỗi lớp được thực hiện thông qua quá trình truyền ngược của lỗi. Nếu mạng quá sâu, lỗi này xuất phát từ lớp có khoảng cách xa, khi truyền đi sẽ bị giảm (hoặc khuếch đại) theo hàm số mũ

qua từng lớp. Giả sử mỗi lần truyền qua một lớp, giá trị sẽ giảm 1 nửa, vậy sau 10 lớp, giá trị còn $0.5^{10} = 0.00097$, có nghĩa giá trị gradient của lớp thứ 11 từ cuối lên trên chỉ bằng 1/1000 gradient của lớp cuối cùng. Lúc này gradient dần dần tiến tới 0, có nghĩa là dần dần "biến mất" trong quá trình truyền ngược. Sau khi gradient biến mất, quá trình tối ưu hóa của mạng mất đi sự hướng dẫn và sẽ không tìm được giải pháp tốt hơn (như *Hình 3-25*).

Hình 3-25: Mạng bị bối rối

Các kỹ thuật thường được sử dụng để giải quyết vấn đề này là batch normalization và cross-layer connection (shortcut). Chúng ta sẽ không thảo luận chi tiết, nếu quan tâm các bạn có thể tìm kiếm các thông tin liên quan.

3.4. Ứng dụng phân loại trong cuộc sống hàng ngày.

Sau khi học được rất nhiều kiến thức về học sâu và phân loại hình ảnh, ngoài việc giúp Minh sắp xếp lại album ra chúng ta có thể làm điều gì khác? Trong thực tế, công nghệ phân loại hình ảnh xuất hiện khắp nơi trong cuộc sống hàng ngày và có một loạt ứng dụng như nhận dạng khuôn mặt, tìm kiếm hình ảnh... Hãy lấy nhận diện khuôn mặt làm ví dụ để giới thiệu những thay đổi mà phân loại hình ảnh và học sâu mang lại cho cuộc sống.

Trong năm 2014, đội tuyển Đại học Hồng Kông Trung Quốc đã đưa hiệu suất của máy móc trong nhiệm vụ nhận dạng khuôn mặt vượt qua con người lần đầu tiên. Từ sự kiện mang tính bước ngoặt này, "nhận dạng khuôn

mặt" đã trở thành một trong những nhiệm vụ của thuật toán học sâu, trở thành ứng dụng học sâu đầu tiên giúp thay đổi cuộc sống của chúng ta.

Thuật toán máy học truyền thống cũng đã cố gắng giải quyết vấn đề này trước khi DNN được áp dụng cho nhiệm vụ "nhận dạng khuôn mặt". Tuy nhiên, thuật toán truyền thống không bảo đảm đồng thời tính chính xác và hiệu quả trong quá trình nhận dạng, tình trạng này thuật toán nhận diện khuôn mặt truyền thống khó đạt được quy mô ứng dụng. Các mô hình học sâu hiện tại đã được huấn luyện hàng trăm triệu hình ảnh dữ liệu khuôn mặt có thể đáp ứng các yêu cầu yêu cầu quy mô lớn và độ chính xác cao đồng thời, và nó cũng có thể áp dụng đối với tất cả các khía cạnh của cuộc sống.

Nhận diện khuôn mặt là một quá trình "tìm khuôn mặt" để "nhận diện" từ một hình ảnh kỹ thuật số hoặc một khung hình video, trong đó "nhận diện khuôn mặt" là một nhiệm vụ phân loại hình ảnh. Cụ thể, toàn bộ quy trình nhận dạng bao gồm các bước sau: Phát hiện khuôn mặt, trích xuất đặc trưng, so sánh khuôn mặt, bảo quản và phân tích dữ liệu. Phát hiện khuôn mặt bao gồm phát hiện ảnh chứa khuôn mặt, vị trí của khuôn mặt, góc khuôn mặt..., tức là hoàn thành việc "nhìn". Trích xuất đặc trưng là làm cho máy "hiểu": Bằng cách phân tích khuôn mặt được phát hiện trong bước trước, ta thu được các đặc trưng tương ứng của khuôn mặt, chẳng hạn như khuôn mặt đang mỉm cười hay đeo kính. Thông tin thu được trong hai bước này sẽ được sử dụng để so sánh các bức chân dung đã được ghi lại trong số khuôn mặt (chẳng hạn như ảnh thẻ ID) theo một cách nhất định để giải quyết vấn đề "follow-to-do". Cuối cùng, kết quả của các phân tích này sẽ được sử dụng tùy theo tình hình cụ thể.

Ý nghĩa của việc cho phép máy "nhìn thấy", "hiểu" và "nhận ra" khuôn mặt là gì? Hãy xem xét một vài ví dụ ứng dụng cụ thể điển hình dưới đây. Công nghệ nhận dạng khuôn mặt mang lại những thay đổi đẹp cho cuộc sống của chúng ta.

Quét mặt: Nhận dạng khuôn mặt giúp cuộc sống tiện lợi hơn

Kể từ khi chiếc máy tính đầu tiên ra đời, cuộc sống chúng ta đã trải qua những thay đổi to lớn do sự phát triển của ngành công nghiệp Internet, chúng ta có thể bắt đầu bước vào "thời đại thông tin" mà không cần biết thế giới (?). Năm 2006, Bai Jiefei Sinton và những người khác đã đề xuất khái niệm học tập sâu, thứ có thể khiến mạng nơron . Cuộc sống của chúng ta dẫn chúng ta vào "thời đại trí tuệ nhân tạo" (như trong *Hình 3-26*).

Năm 2006, Bai Jiefei Sinton và cộng sự đã đề xuất khái niệm **Deep Learning**, khiến mạng nơron có các technical hotspot hiệu quả và mạnh mẽ hơn. Sau một khoảng lặng ngắn, nó sẽ thay đổi nhanh chóng mà không thể

ngăn cản. Cuộc sống dẫn dắt chúng ta đến "kỷ nguyên trí tuệ nhân tạo" (như mô tả *ở Hình 3-26*).

Hình 3-26: Trường hợp ứng dụng nhận diện khuôn mặt phong phú

Với công nghệ nhận dạng khuôn mặt, cuộc sống của chúng ta ngày càng thuận tiện hơn. Khi đi mua sắm, chúng ta chỉ cần quét mặt để thanh toán, thật mới lạ và thuận tiện. Khi đi tàu điện ngầm, chúng tôi không phải lo lắng về việc quên mang theo thẻ tàu điện ngầm, hệ thống sẽ tự động quét khuôn mặt để vào trạm, một cách tốt để tránh tình trạng mua vé và xếp hàng vào nhà ga, tiết kiệm rất nhiều thời gian đi lại. Khi đi làm vào buổi sáng hoặc đi học, ta không cần nhân viên ghi lại, hệ thống trực tiếp tại cửa ra vào sẽ tự động xác định và ghi lại thời gian vào ra của bạn, đảm bảo tính xác thực. Ứng dụng cụ thể của công nghệ nhận dạng khuôn mặt là vô tận, cuộc sống của chúng ta được thúc đẩy mạnh mẽ bởi công nghệ trí tuệ nhân tạo, dần dần bước ký nguyên mới tốt hơn, thuận tiện hơn.

Skynet: Công nghệ nhận dạng khuôn mặt hỗ trợ an ninh

Nhà ga đông người, và những kẻ tình nghi ăn cắp kho báu quốc gia được trộn lẫn trong đám đông. Tuy nhiên, cảnh sát mặc thường phục tìm kiếm những kẻ tình nghi không khác gì việc tìm một cây kim trong đống cỏ khô. Trong khi đó, tại trụ sở của chỉ huy ở một vị trí khác trong thành phố, hình ảnh của giám sát độ nét cao được chiếu trên màn hình, áp lực trong video giám sát tăng lên, mắt người khó có thể nắm bắt được thông tin hiệu quả từ video giám sát. Nhưng cảnh sát không bận tâm với điều này vì công nghệ trí tuệ nhân tạo đã thực hiện điều đó thay cho họ. Mỗi khuôn mặt trong đám đông được phát hiện chính xác, việc trích xuất đặc trưng và phân tích được hoàn thành theo thời gian thực, sau đó được so sánh với những tên tội phạm bị truy nã trong thư viện hình sự. Đột nhiên, tiếng chuông báo động vang lên qua phòng chỉ huy, nghi phạm đã được tìm thấy! Công nghệ nhận diện khuôn mặt "đôi mắt lửa" đã tìm thấy nghi phạm chính xác trong đám đông và đóng khung vị trí của kẻ tình nghi. Ngay lập tức, nghi phạm bị bắt.

Hình 3-27: Sơ đồ giao diện hệ thống điều khiển xác định khuôn mặt

Mô tả ở trên là cảnh trí tuệ nhân tạo trong công nghệ nhận dạng khuôn mặt giúp cảnh sát bắt giữ nghi phạm. Khi dân số và di dân gia tăng, tầm quan trọng của công việc an ninh tăng lên và đối mặt với những thách thức lớn hơn. Những người vi phạm pháp luật luôn luôn có một số may mắn, nghĩ rằng họ có thể giấu tên của họ hoặc ẩn trên thị trường và trốn tránh các biện pháp trừng phạt pháp lý. Hệ thống kiểm soát khuôn mặt hoàn hảo sẽ khiến mọi tội phạm không thể che giấu được. Hệ thống điều khiển khuôn mặt được trang bị công nghệ nhận diện khuôn mặt tiên tiến, đồng thời với mạng lưới giám sát được cải tiến hàng ngày, bằng cách xây dựng hệ thống tự động từ nhận thức, cảnh báo sớm, phân tích đến việc ra quyết định, chúng ta có thể trích xuất thông tin hữu ích từ video giám sát theo thời gian thực, dần dần nâng cấp từ "nhìn rõ ràng" lên cấp độ cao hơn là "hiểu". Một ngày nào đó, với hệ thống giám sát nghiêm ngặt, không một tên tội phạm nào có thể thoát khỏi sự trừng trị của pháp luật được nữa.

3.5. Tóm tắt chương

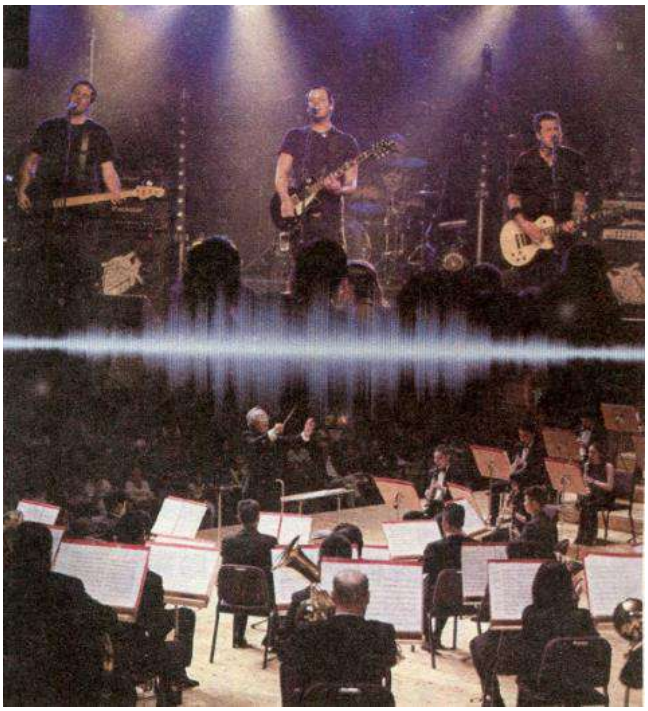
Trong chương này, chúng ta đã học được cách thực hiện phân loại hình ảnh. Đầu tiên, chúng ta hiểu được cách biểu diễn của hình ảnh trên máy tính, biết rằng việc khai thác tính năng hình ảnh là một phép toán cụ thể trên tensor bậc ba. Trong quá trình trích xuất đặc trưng thủ công, chúng ta đã tìm hiểu về phép tích chập và và làm cách nào sử dụng phép tích chập để trích

xuất đặc trưng hình ảnh, chẳng hạn như các đặc trưng biên và HOG. Các đặc trưng thiết kế thủ công có những hạn chế của chúng và việc sử dụng các đặc trưng được tự động học bằng DNN đã được sử dụng rộng rãi trong xử lý hình ảnh, vì vậy chúng ta tập trung vào cách sử dụng DNN cho phân loại hình ảnh. Chúng ta đã tìm hiểu sơ bộ về cấu trúc DNN và hiểu một số lớp cơ bản trong mạng như, convolutional, lớp pooling, fully connected, ReLU và lớp softmax. Đồng thời, chúng ta đã biết về quá trình huấn luyện mạng bằng thuật toán backpropagation. Cuối cùng là sự phát triển và thách thức của DNN, những áp dụng rộng rãi của việc phân loại hình ảnh trong cuộc sống hàng ngày.

Thông qua việc nghiên cứu kiến thức và hoàn thành các trải nghiệm liên quan, chúng ta có thể hiểu học sâu có sức mạnh vượt trội hơn các phương pháp truyền thống, vì vậy nó có thể thực hiện tốt hơn những nhiệm vụ phức tạp. Đồng thời, việc huấn luyện mạng nơron nhiều lớp đòi hỏi nhiều dữ liệu và khả năng tính toán hơn.

Qua thời gian, sự phát triển của deep learning đã có những bước tiến lớn, đồng thời cũng liên tục gặp phải những vấn đề và thách thức mới. Những thách thức này góp phần làm động lực cho nghiên cứu hiện tại, thúc đẩy việc cải tiến liên tục các kết quả nghiên cứu hiện có và sự phát triển của các công nghệ mới.

Chương IV. Phân tích âm nhạc



Cuối tuần, Minh đi nghe 2 buổi hòa nhạc. Buổi đầu là nhạc rock. Trong giai điệu mạnh mẽ mạch lạc, hoặc đam mê hoặc giai điệu buồn, đi cùng với sự chăm chỉ của ca sĩ. Buổi thứ 2 là dàn nhạc, âm thanh của các nhạc cụ khác nhau đan xen lại. Nhiều sự thay đổi phong phú. Thời điểm đó giống như uống và nói chuyện tại bữa tiệc. Trong nháy mắt, nó lại giống như cuộc chiến trên chiến trường. Khi về nhà, Minh nghĩ có rất nhiều thay đổi tuyệt vời trong âm thanh. Liệu máy tính có thể đánh giá những giai điệu đẹp đẽ đó như con người ?

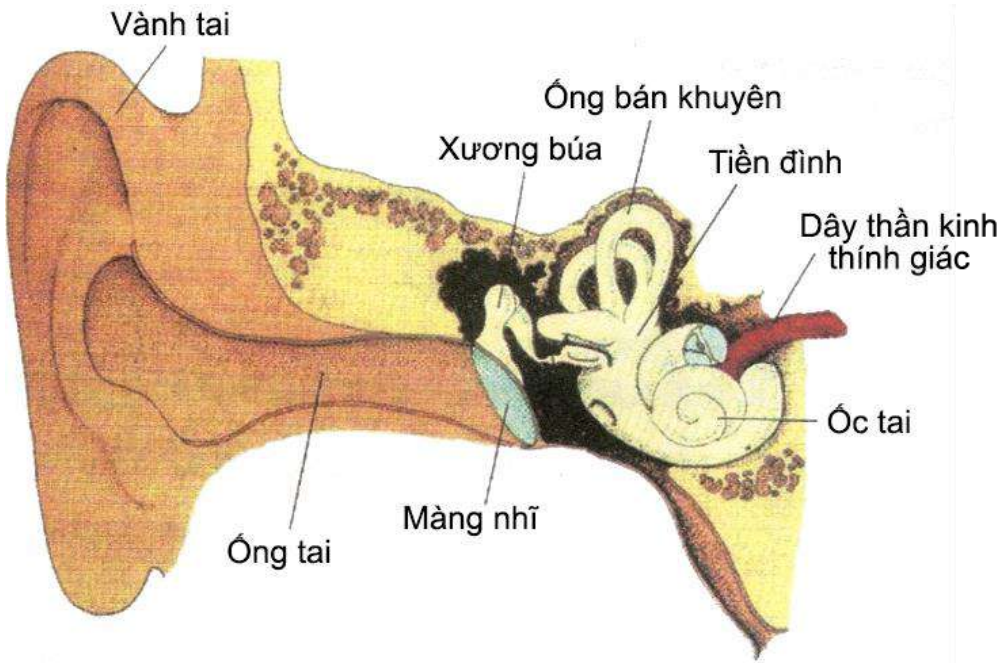
Giữa hàng nghìn loại âm thanh trên thế giới, có 2 loại đặc biệt liên quan chặt chẽ tới con người. 1 loại là âm thanh trao đổi thông tin tạo ra bởi cơ quan thanh nhạc của con người, được gọi là lời nói (speech), loại còn lại là do con người tạo ra. Nghệ thuật âm thanh với giai điệu và nhịp điêu là âm nhạc (music). Âm thanh đóng vai trò quan trọng trong giao tiếp hàng ngày. Chúng ta tiếp xúc rất nhiều người trong ngày, câu hòi từ giáo viên, chia sẻ với bạn bè trong lớp, nói chuyện với bố mẹ,... Tiếng nói chắc chắn là công cụ hiệu quả nhất. Thật thú vị, thính giác của con người ưa thích lời nói. Ở một địa điểm ồn ào, còn người có thể xác định chính xác người đang nói chuyện với họ đang nói về vấn đề gì, còn được gọi là hiệu ứng tiệc cocktail. Âm nhạc là một nghệ thuật độc đáo. Khi giai điệu chạm được đến cảm xúc, những suy nghĩ đều được đắm chìm trong các nốt nhạc, cảm nhận được sự thanh tẩy của tâm hồn. Âm nhạc có một lịch sử lâu đời, có những giai điệu phong cách hoàn toàn khác nhau, như nhạc jazz, rock và các phong cách khác. Nhạc sĩ có phong cách riêng và thính giả có sở thích riêng. Có hàng trăm tư tưởng tranh luận trong lĩnh vực âm nhạc.

Trong chương trước, chúng ta đã cài đặt một cặp “Mắt thấu thị” trên máy tính để cho nó học cách nhận ra hình ảnh. Trong chương này, chúng ta cần phải sử dụng trí thông minh nhân tạo để cung cấp cho máy tính một đôi "Tai gió" để máy tính có thể hiểu được giọng nói, đánh giá âm nhạc và khám phá bí ẩn của âm thanh.

4.1. Nghệ thuật lắng nghe

Tai người

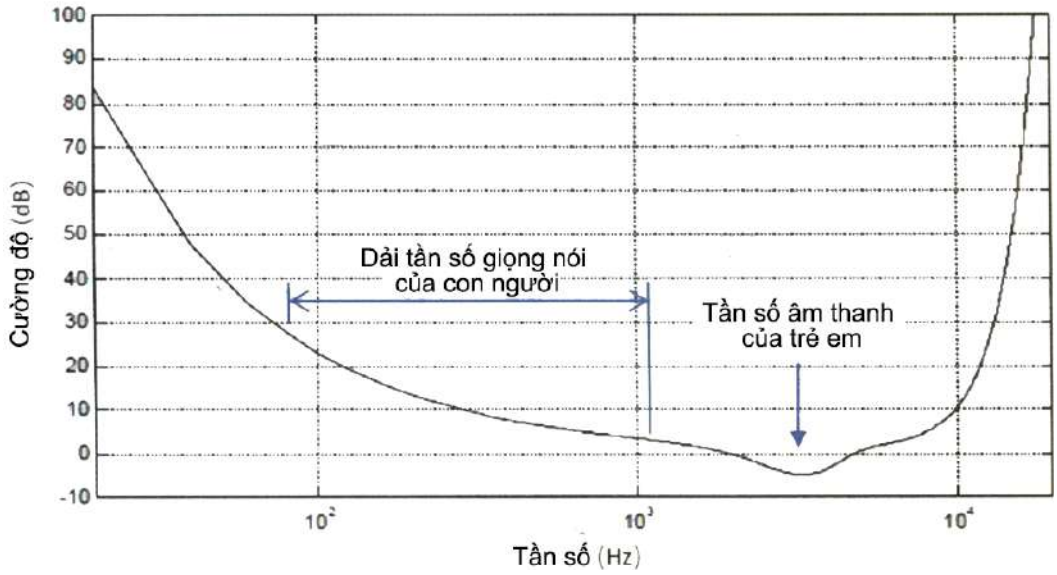
Chúng ta đã học nguyên tắc tạo sóng âm và truyền sóng âm trong môn vật lý. Âm thanh được tạo ra bởi sự rung động của vật thể, truyền trong môi trường, sau đó tới tai người và được nhận thấy bởi con người. Chúng ta cũng đã học cấu trúc tai người trong môn sinh như được mô tả ở *Hình 4-1*. Sóng âm được thu lại bởi vành tai và truyền đến ốc tai qua loạt cấu trúc. Có nhiều thụ thể thính giác trong ốc tai truyền âm thanh đến các dây thần kinh thính giác và cuối cùng gây ra cảm nhận về âm thanh.



Hình 4-1: Cấu tạo tai người

Tần số là một đặc trưng quan trọng của âm thanh. Nó biểu thị số lần vật thể rung trong một giây, đơn vị Hertz (Hz). Cấu trúc tinh tế của tai người cũng xác định rằng chúng ta có độ nhạy khác nhau với các tần số khác nhau của âm thanh, như trong *Hình 4-2*, trục ngang biểu diễn cho tần số, và trục dọc biểu diễn cường độ của âm thanh, đơn vị decibel (dB). Giá trị càng nhỏ, con người càng nhạy cảm với âm thanh của tần số. Thật thú vị, tần số nhạy cảm nhất của tai người giống như tần số giọng nói của đứa trẻ. Tần số của giọng nói của con người nằm trong khoảng 85-1100 Hz. Có thể thấy rằng tai

người cũng tương đối nhạy cảm với tần số trong phạm vi này. Điều này có nghĩa là cấu trúc của tai người phần lớn thuận tiện cho giao tiếp giữa con người với nhau. Hãy tưởng tượng nếu chúng ta có thể nghe thấy tiếng siêu âm, thế giới sẽ trở nên như thế nào?.



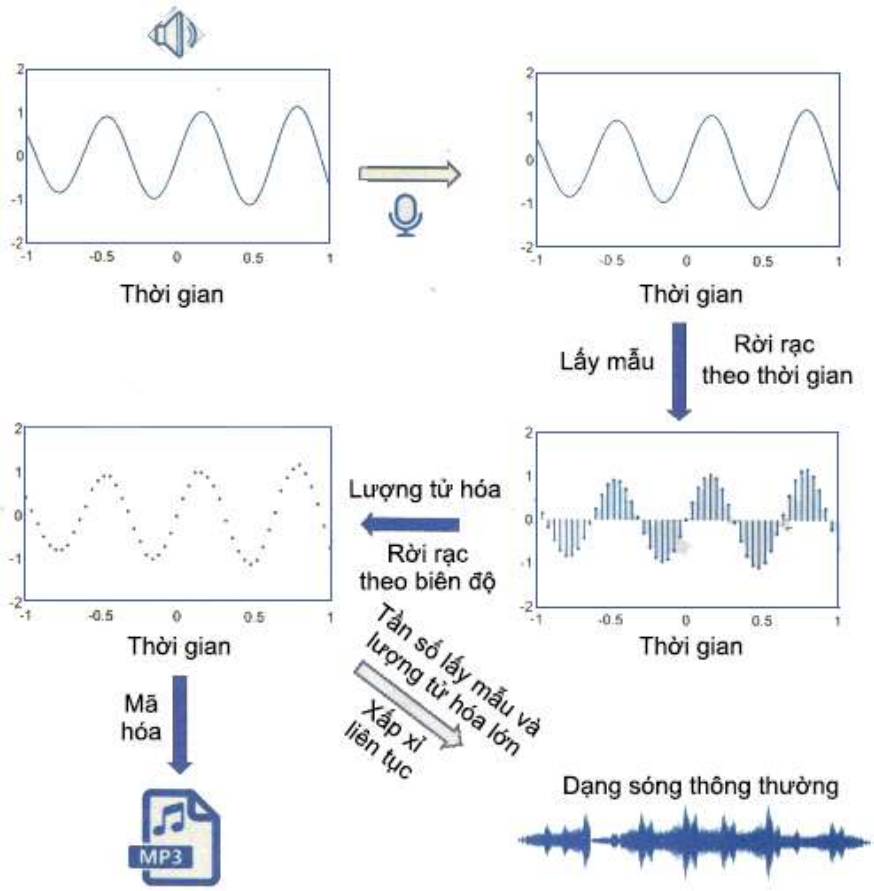
Hình 4-2: Độ nhạy của thính giác tai người với âm thanh tần số khác nhau.

Số hóa âm thanh

Máy tính không có tai, vì vậy nó cảm nhận âm thanh như thế nào? Bạn cần phải chuyển đổi các sóng âm thanh thành các tệp âm thanh dễ lưu trữ và xử lý (chẳng hạn như định dạng MP3). Quá trình này được thể hiện trong hình 4-3. Quá trình từ các sóng âm thanh đến tệp MP3 cuối cùng trải qua các bước chính gồm **lấy mẫu** (sampling), **lượng tử hóa** (quantization) và **mã hóa** (encoding).

Đầu tiên, chúng ta chuyển đổi sóng âm thành tín hiệu điện (như điện áp) thông qua các cảm biến trong micro, giống như các thụ thể thính giác trong ốc tai truyền sóng âm thanh tới dây thần kinh thính giác. Nhưng máy tính không thể lưu trữ tín hiệu liên tục, vì vậy chúng ta cần phải làm cho tín hiệu điện rời rạc theo thời gian bằng cách lấy mẫu, và sau đó lượng tử hóa nó để làm cho nó rời rạc trong biên độ. Âm thanh trở thành một điểm dữ liệu rời rạc và máy tính có thể lưu trữ nó ở các định dạng tệp khác nhau thông qua các phương thức mã hóa khác nhau. Định dạng MP3 thường được sử dụng khi nghe nhạc là một trong số đó. Thực chất tệp âm thanh trong máy tính mô tả một loạt các điểm dữ liệu được sắp xếp theo trình tự thời gian, vì vậy nó cũng được gọi là chuỗi thời gian (time series), hãy hình dung chúng như dạng

sóng phổ biến (waveform), trục ngang đại diện cho thời gian, hình dạng sóng không có ý nghĩa vật lý trực tiếp mà phản ánh chuyển động rung của cảm biến khi có âm thanh. Vì sự dịch chuyển rung liên tục dao động trong khoảng thời gian bằng không, dạng sóng cũng dao động xung quanh 0 theo thời gian. Khi tần số lấy mẫu cao, dạng sóng xuất hiện gần như liên tục.



Hình 4-3: Số hóa âm thanh

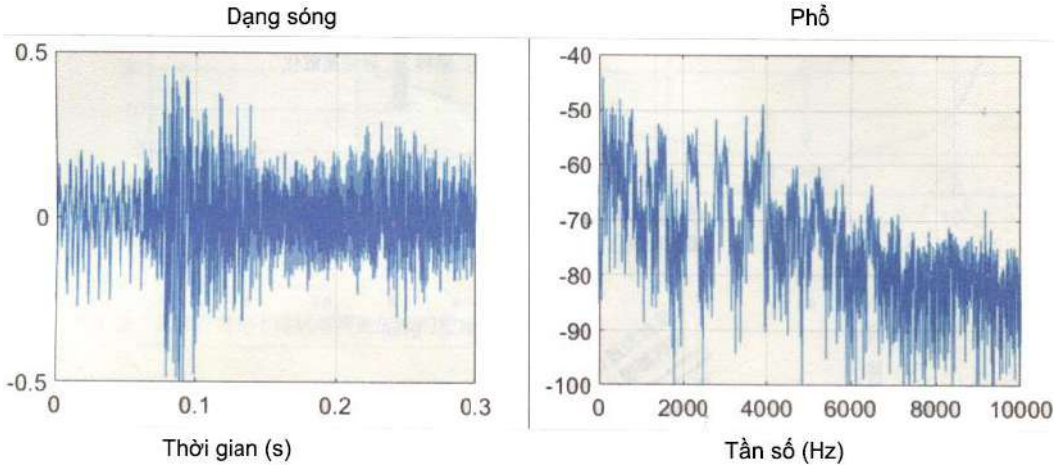
Kiến thức bổ sung: Tần số lấy mẫu

Tần số lấy mẫu (sampling rate) cũng được gọi là tốc độ lấy mẫu. Tốc độ lấy mẫu càng cao thì âm thanh được khôi phục càng tự nhiên hơn. Thông thường tốc độ lấy mẫu của định dạng MP3 là 4100 Hz. Bởi vì tai người không nhạy cảm với âm thanh tần số cao, việc tiếp tục tăng tỷ lệ lấy mẫu có ít ảnh hưởng đến trải nghiệm thính giác, nhưng nó lãng phí không gian lưu trữ.

Tương tự như hình ảnh, phạm vi giá trị sau khi số hóa âm thanh cũng bị giới hạn. Âm thanh chung thường có hai kênh (tương ứng với tai trái, phải) và hình ảnh thường có ba kênh (tương ứng với màu đỏ, xanh lục và xanh dương). Trừ khi có quy định khác trong chương này, chúng ta chỉ xem xét một kênh.

Hiểu ba yếu tố của âm nhạc thông qua phổ âm thanh

Thông qua việc số hóa âm thanh, máy tính "lắng nghe" âm thanh, vậy máy tính "hiểu" âm thanh như thế nào? Ở đây chúng ta lấy ba yếu tố của âm nhạc như là một ví dụ để tìm hiểu một phương pháp phân tích âm thanh của máy tính phổ biến - phổ tần số (frequency spectrum). Hình 4-4 cho thấy dạng sóng (trái) và phổ (phải) của một đoạn nhạc.



Hình 4-4: Dạng sóng và phổ của một đoạn nhạc

Trục ngang của phổ biểu diễn tần số, trục dọc đại diện cho biên độ của phổ, có nghĩa là biên độ tương ứng với các tần số khác nhau. Vì cường độ âm thanh của các tần số khác nhau trong một đoạn âm thanh có sự chênh lệch lớn, biên độ phổ thường sử dụng logarit, nghĩa là biên độ phổ khác nhau 20 thì biên độ khác nhau 10 lần. Ví dụ, nếu biên độ 1000 Hz trên phổ là -50, và biên độ tương ứng là 5000 Hz là -70, thì biên độ âm thanh 1000 Hz lớn hơn 10 lần so với biên độ âm thanh 5000 Hz. Phổ phản ánh lượng năng lượng bị chiếm bởi âm thanh ở các tần số khác nhau, và chúng ta thường chỉ tập trung vào độ lớn của phổ. Ví dụ, trong một điệp khúc, âm treble và bass mạnh thì yếu, dải tần số tương ứng với tần số cao trong một phạm vi nhất định là lớn, và dải tần số tương ứng với vùng tần số thấp là lớn.

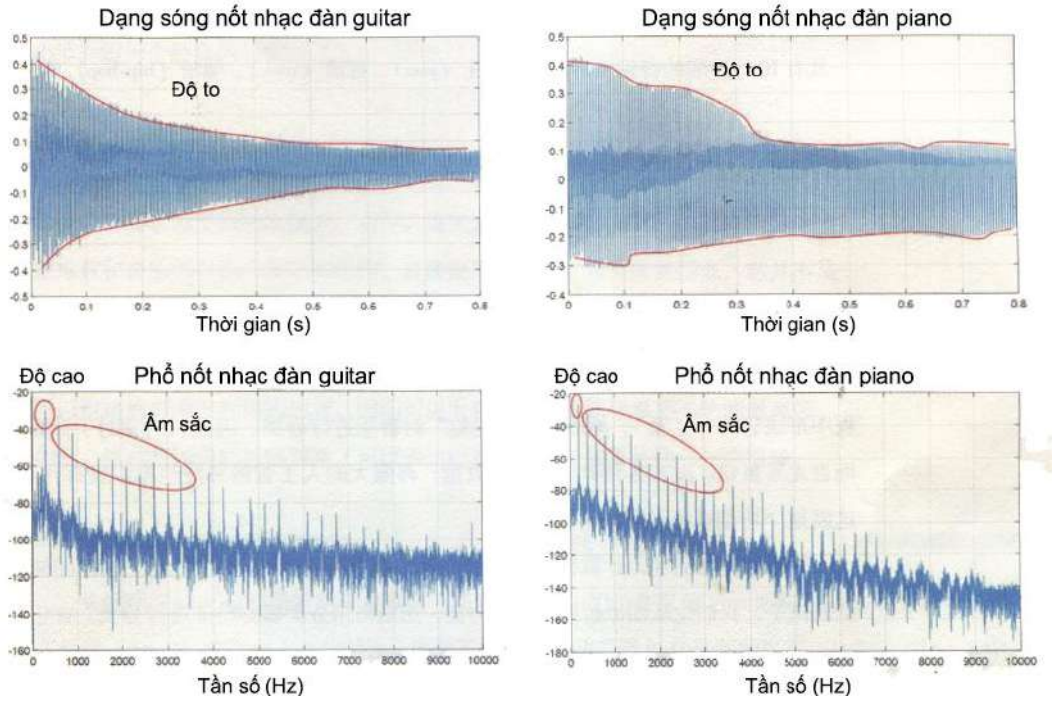
Trong môn vật lý, các bạn đã được học 3 yếu tố của âm thanh là **độ to**, **độ cao** và **âm sắc** có thể mô tả các đặc điểm của âm thanh.

Độ to: Yếu tố trực quan nhất, miêu tả độ mạnh của âm thanh, biểu diễn bởi cường độ.

Độ cao: Mức độ của âm thanh con người có thể nghe. Tần số càng cao, âm thanh càng cao và ngược lại, vì vậy phổ có thể sử dụng để miêu tả độ cao.

Âm sắc: Là đặc trưng phức tạp hơn. Âm thanh được phát ra với cùng độ cao và độ to của những nhạc cụ khác nhau hoặc hát với những người khác nhau thì nghe sẽ khác nhau. Lý do là vì trong quá trình rung của nhạc cụ hay dây thanh quản, ngoài tần số tương ứng với độ cao là f, còn đi kèm với các tần số cao khác (tần số 2f, 3f, 4f...) được gọi là âm bội. Những âm bội này có các cường độ khác nhau, tạo nên sự độc đáo về âm thanh.

Hãy cùng theo dõi ví dụ về guitar và piano. Trong hình 4-5, bên trái là dạng sóng và phổ của âm thanh khi gảy dây guitar và bên phải là dạng sóng và phổ của âm thanh bấm phím piano. Rất dễ dàng để thấy biểu đồ dạng sóng của cả nhạc cụ độ to thay đổi từ lớn đến nhỏ. Từ phổ, ta có thể thấy loạt các đỉnh. Tần số tại đỉnh cao nhất đầu tiên là độ cao của âm, vị trí của bội số của tần số có các kích thước khác nhau, tỉ lệ giữa chúng phản ánh âm thanh khác nhau.



Hình 4-5: Dạng sóng và phổ của nốt nhạc đàn guitar và piano

4.2. Phân loại phong cách âm nhạc

Trong phần này, chúng ta muốn máy tính hiểu về âm thanh sâu hơn và hoàn thành nhiệm vụ phân loại phong cách âm nhạc. Phong cách âm nhạc được tạo ra bởi sự kết hợp của nhiều yếu tố âm nhạc khác nhau. Nhiệm vụ của chúng ta là để máy tính "lắng nghe" một đoạn nhạc dài 30 giây và sau đó xác định loại phong cách (thể loại) của đoạn nhạc đó. Có 10 phong cách âm nhạc có thể có trong nhiệm vụ này, bao gồm nhạc jazz, rock, hip-hop, v.v.

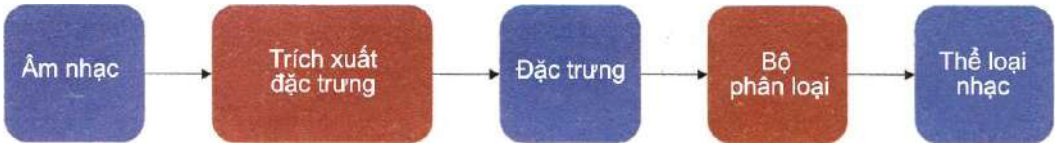
Đầu vào: Chuỗi thời gian (time series) của đoạn nhạc dài 30 giây.

Đầu ra: Thể loại nhạc.

Phong cách âm nhạc “trong tai” máy tính

Con người có thể đánh giá phong cách âm nhạc bằng cảm giác và kinh nghiệm. Máy vi tính cũng cần phải hình thành "kinh nghiệm" trong quá trình "nghe" nhạc, và sau đó phân loại âm nhạc theo "kinh nghiệm". Do đó, dữ liệu đặc biệt quan trọng đối với máy tính. Nếu không có tất cả các loại dữ liệu âm nhạc, các thuật toán trí tuệ nhân tạo mạnh mẽ cũng rất khó để thực hiện vai trò, "có bột mới gột nên hồ".

Dữ liệu âm nhạc trở thành "kinh nghiệm" của máy tính như thế nào? Nhìn lại ví dụ về nhận dạng hình ảnh chúng ta đã hoàn thành trong Chương 3, trước tiên ta trích xuất các đặc trưng từ dữ liệu hình ảnh và sau đó phân loại các đặc trưng bằng cách sử dụng bộ phân loại. Phân loại thể loại nhạc có thể được thực hiện theo cách như vậy, như trong *Hình 4-6*, ta chia nhiệm vụ thành hai phần. Đầu tiên, thiết kế một bộ trích xuất đặc trưng để trích xuất các đặc trưng từ âm nhạc; thứ hai, huấn luyện một bộ phân loại để xác định thể loại dựa trên các đặc điểm của âm nhạc.



Hình 4-6: Lưu đồ phân loại nhạc

Kiến thức bổ sung: Đặc trưng

Một đặc trưng thường là chuỗi ngắn hơn nhiều so với dữ liệu, nhưng chứa hầu hết các thông tin tiêu biểu của dữ liệu. Ví dụ, chúng ta có thể có một mô tả của đoạn nhạc: “Trống, guitar và tiếng bass được hòa quyện, giai điệu thay đổi một cách nhịp nhàng, mọi người không thể không lắc lư cùng với nhau”. Nhạc rock không khó để mô tả theo ví dụ đó. Hãy coi đoạn nhạc đó là nhạc rock, vì vậy trống, guitar và bass là .., có thể dùng làm đặc trưng của thể loại rock. Đặc trưng này dễ để hiểu, nhưng chỉ có thể diễn tả bằng ngôn ngữ. Để thiết kế các đặc trưng dễ hiểu cho máy tính, ta cần nhập vào dữ liệu nhạc.

Ta có thể tưởng tượng chuỗi thời gian của đoạn nhạc như một vector. Với đoạn nhạc 30 giây, tần số lấy mẫu là 44100Hz, kích thước của vector tương ứng với đoạn nhạc đó là bao nhiêu ?

Tấn số lấy mẫu được biết là chuỗi thời gian mỗi giây có thể được thể hiện bằng vector kích thước 44100, vì vậy kích thước của cả đoạn nhạc là 30x44100 = 1323000, khoảng 1.3 triệu. Một bức ảnh đen trắng độ phân giải 1000x1000 chứa 1 triệu pixel, được thể hiện bằng vector kích thước 1 triệu, cũng tương đương đoạn nhạc 30s. Sử dụng bộ phân loại để trực tiếp phân loại dữ liệu kích thước lớn như vậy là kém hiệu quả và khiến bộ phân loại chịu gánh nặng tính toán. Vì vậy trích xuất các đặc trưng tốt từ dữ liệu nhạc là một phần rất quan trọng, phần tiếp theo sẽ giới thiệu một đặc trưng cổ điển.

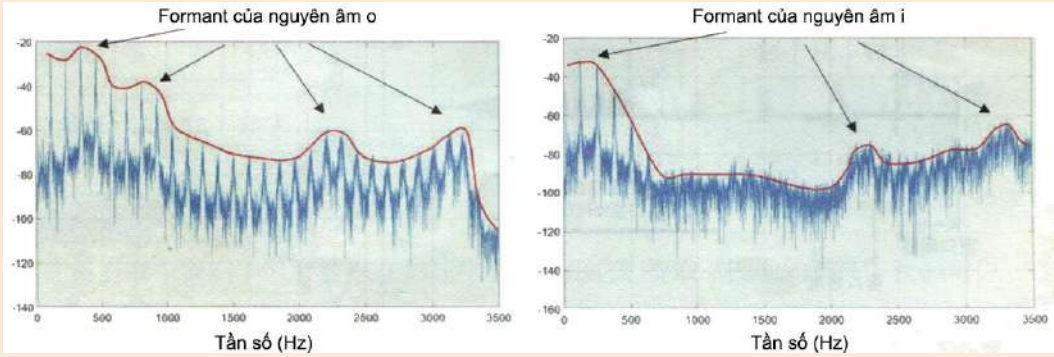
Đặc trưng âm thanh: Mel-Frequency Cepstral Coefficients

Chúng ta đã học được khái niệm về phổ, có thể phản ánh thông tin của ba yếu tố âm nhạc. Tuy nhiên, kích thước dữ liệu của phổ và kích thước dữ liệu của âm nhạc là như nhau, khó có thể trực tiếp sử dụng phổ để phân loại, và nó không phải là một đặc trưng tốt. Ở đây chúng ta muốn tìm hiểu một đặc trưng hiệu quả hơn và được sử dụng rộng rãi hơn so với phổ - **Mel-Frequency Cepstral Coefficients** (MFCC). Kích thước của đặc trưng này rất nhỏ, nó có thể mô tả gần giống hình dạng của phổ, vì vậy có thể mô tả gần đúng mức năng lượng của âm thanh các tần số khác nhau. Không chỉ vậy, MFCC còn thể hiện một đặc điểm quan trọng của âm thanh - **formant**.

Kiến thức bổ sung: Formant

Formant là các vùng trên phổ tương đối tập trung năng lượng. Format được sử dụng phổ biến trong phân tích giọng nói vì nó hiển thị rất rõ ràng trong phổ nguyên âm và các đỉnh cộng hưởng của các nguyên âm khác nhau cũng khác biệt đáng kể. Ví dụ, trong Hình 4-7, vị trí của formant trên phổ khác nhau đáng kể giữa nguyên âm o (trái) và i (bên phải).

Tại sao có formant trong phổ? Nó chỉ ra rằng khi một người nói thì miệng, mũi, và cổ họng tạo thành một khoang được kết nối. Âm thanh của một tần số cụ thể được khuếch đại và formant được biểu diễn trên phổ. Quá trình tạo ra âm thanh của nhạc cụ có thể được so sánh với việc phát âm của con người. Ví dụ thân đàn violin là hộp cộng hưởng, hoạt động giống miệng của con người. Âm thanh có các tần số nhất định có thể cộng hưởng trong hộp cộng hưởng để hình thành một giọng nhất định

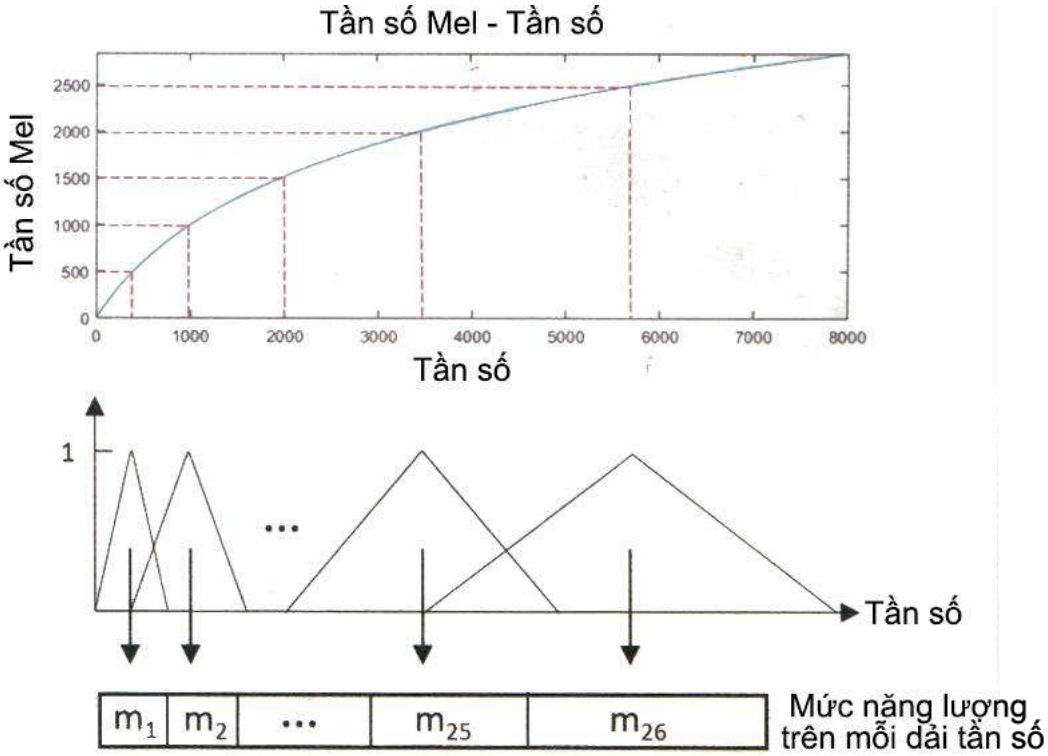


Hình 4-7: Các formant

Vì đặc trưng MFCC có rất nhiều ưu điểm, làm thế nào để trích xuất đặc trưng này? Như tên gọi của nó, trước tiên chúng ta phải xử lý phổ tần số Mel để thu được một tập các đặc trưng 26 chiều, và sau đó tính toán cepstrum của nó để có được 13 đặc trưng MFCC cuối cùng. Chúng ta hãy xem xét quá trình cụ thể của hai bước này.

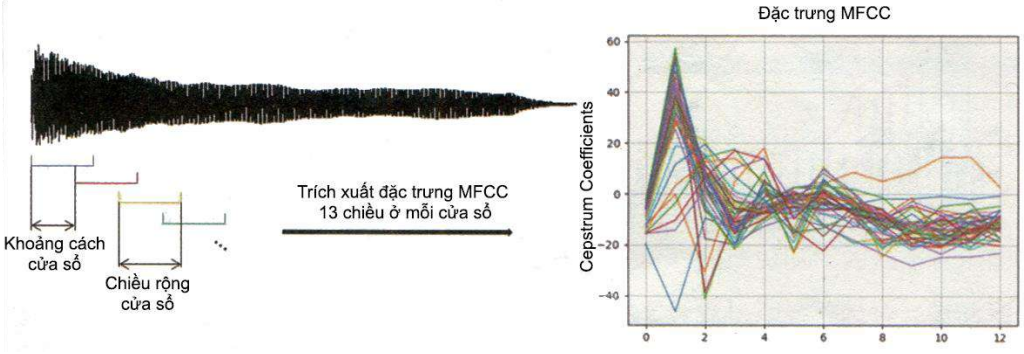
Như thể hiện trong *Hình 4-8*, Mel-Frequency là một thang tần số đặc biệt, là một hàm số của tần số thông thường $mel(f) = 1125 \times \ln\left(1 + \frac{f}{700}\right)$. Khoảng tần số có cùng độ dài dưới thang tần số Mel tương ứng với khoảng không bằng nhau ở tần số thông thường: độ phân giải cao ở phần tần số thấp và độ phân giải thấp ở phần tần số cao. Điều này tương tự như nhận thức thính giác của tai người, tức là nhạy cảm với âm thanh tần số thấp và không nhạy cảm với âm thanh tần số cao trong một dải tần số nhất định. Phổ được tính trung bình trên mỗi khoảng tần số, biểu diễn cho lượng năng lượng âm

thanh trong mỗi dải tần số. Có tổng cộng 26 dải tần số, dẫn đến đặc trưng 26 chiều.



Hình 4-8: Mel-frequency

Cepstral thu được bằng cách biến đổi toán học các đặc trưng 26 chiều nói trên và giảm kích thước xuống 13 để có được đặc trưng MFCC. Quá trình chuyển đổi cụ thể phức tạp hơn. Đặc trưng 13 chiều này phản ánh năng lượng của tín hiệu âm thanh trong các dải tần số khác nhau. Nó giữ lại một số đặc trưng quan trọng của tín hiệu âm thanh, bao gồm cả các formant mà ta đã tìm hiểu.

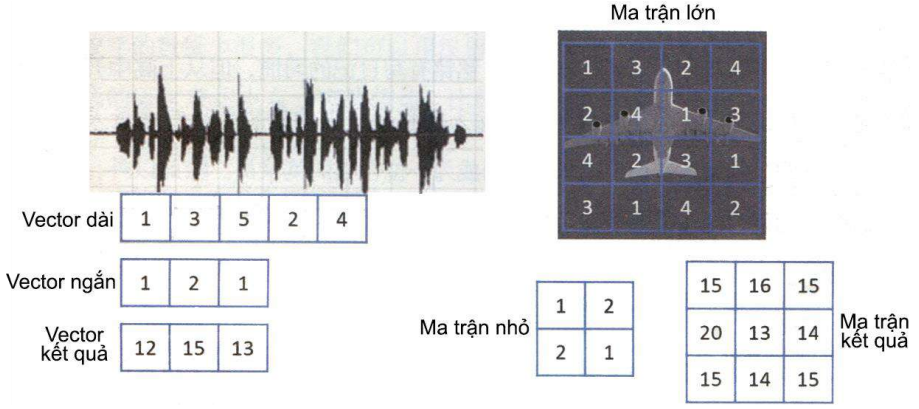


Hình 4-9: Trích xuất đặc trưng MFCC

Hình 4-9 là một ví dụ trích xuất đặc trưng MFCC. Đầu tiên đoạn âm thanh được chia làm các khoảng bằng nhau (có thể chồng lấn lên nhau). Các đoạn được chia gọi là các cửa sổ và có hai tham số là chiều rộng và khoảng cách các cửa sổ. Những tham số này có thể được điều chỉnh tùy thuộc vào đặc điểm đoạn âm thanh. Giá trị điển hình của các tham số là chiệu rộng 25 mili giây và khoảng cách 10 mili giây.

Phương pháp học sâu

Nhiệm vụ của chúng ta có hai phần chính: Đầu tiên là trích xuất các đặc trưng và thứ hai là phân loại các đặc trưng. Chúng ta đã tìm hiểu về đặc trưng MFCC và giờ đã có thể thiết kế một bộ phân loại để phân loại các đặc trưng MFCC của nhạc. Để đạt được độ chính xác cao hơn, chúng ta sẽ sử dụng mạng nơron. Đầu vào là đặc trưng MFCC của nhạc và đầu ra là thể loại của nó. Trên thực tế, mạng nơron này trích xuất các đặc trưng mạnh mẽ hơn dựa trên MFCC và sử dụng đặc trưng này để hoàn thành việc phân loại kiểu.



Hình 4-10: Tích chập một chiều và hai chiều

Trong chương 3 ta đã tìm hiểu về lớp convolutional và lớp pooling và sử dụng trong trích xuất đặc trưng ảnh. Ta có thể sử dụng một kiến trúc tương tự như vậy để trích xuất đặc trưng của âm thanh, tuy nhiên sự khác biệt là âm thanh chỉ có một chiều thời gian, còn hình ảnh là không gian 2 chiều. Sự khác biệt của convolution kernel sử dụng trong trích xuất đặc trưng âm thanh được mô tả ở *Hình 4-10*. Hãy kiểm tra kết quả tích chập đúng hay sai.

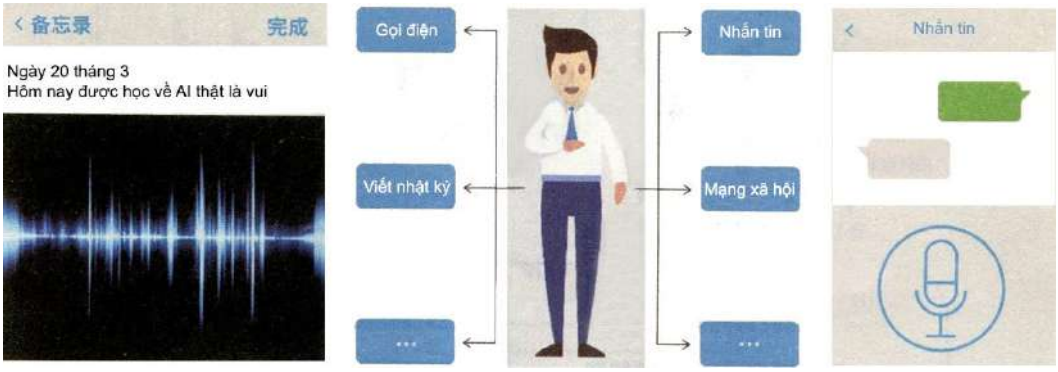
Sau lớp convolutional và pooling, mạng nơron trích xuất các đặc trưng mạnh hơn MFCC và sau đó ta phân loại các đặc trưng này. Tương tự như nhiệm vụ nhận dạng hình ảnh, đầu tiên chúng ta nhận được một chuỗi có cùng độ dài và số lượng thể loại thông qua lớp fully connected và sau đó sử dụng lớp softmax để lấy ra xác suất của mỗi thể loại.

4.3. Công nghệ nhận dạng giọng nói

Ứng dụng nhận dạng giọng nói

Mục đích của nhận dạng giọng nói là dịch các từ được mọi người nói thành các chỉ dẫn mà máy móc có thể hiểu, do đó cho phép giao tiếp bằng giọng nói giữa con người và máy móc. Công nghệ nhận dạng giọng nói đã được sử dụng rộng rãi trong đời thực. Nhân vật chính Minh có thói quen viết nhật ký, nhưng kể từ khi anh bắt đầu học trung học, anh không còn sử dụng cuốn nhật ký mà trực tiếp sử dụng phương thức nhập liệu bằng giọng nói để ghi lại cuộc sống tuyệt vời của mình vào điện thoại di động một cách thuận tiện. Ngoài ra, máy móc có thể hiểu những gì con người nói, và bây giờ nhiều điện thoại thông minh cung cấp trợ lý bằng giọng nói. Minh hiếm khi gõ phím khi nhắn tin mà nói trực tiếp với trợ lý ảo "Gửi tin nhắn cho bố" và nói nội dung muốn gửi, trợ lý ảo sẽ gửi nó.

Gửi tin nhắn văn bản, thực hiện cuộc gọi điện thoại và gọi taxi có thể dễ dàng thực hiện thông qua đối thoại. Có thể tưởng tượng trong tương lai mười năm sau, Minh sẽ có một robot không chỉ hiểu được chỉ dẫn bằng giọng nói để hoàn thành việc nhà, mà còn tham gia các cuộc họp gia đình để đưa ra đề xuất cho cả chuyến đi. Bạn bè của bác sĩ Minh cũng sẽ có một robot thông minh có thể lấy kết quả kiểm tra theo hướng dẫn bằng giọng nói, và thậm chí tham gia thảo luận về các lựa chọn điều trị. Công nghệ nhận dạng giọng nói sẽ tạo nhiều điều kiện thuận lợi hơn nữa cho con người (như trong *Hình 4-11*).



Hình 4-11: Ứng dụng rộng rãi của nhận dạng giọng nói

Nguyên tắc nhận dạng giọng nói

Nhận dạng giọng nói là một nhiệm vụ rất phức tạp, và nó không phải là dễ dàng để đạt đến mức độ thực tế. Chúng ta cũng có thể hiểu nhận dạng giọng nói như một nhiệm vụ phân loại, nghĩa là tìm một từ tương ứng với từng giọng nói của một người. Tuy nhiên, nhiệm vụ phân loại này phức tạp hơn nhiều so với phân loại phong cách âm nhạc mà chúng ta vừa hoàn thành. Phân loại kiểu âm nhạc chỉ cần phân loại toàn bộ đoạn âm thanh một lần và số lượng thể loại nhỏ. Nhận dạng giọng nói cần phải phân loại từng âm thanh, số lượng từ là hàng nghìn và số lượng danh mục có thể cũng nhiều. Như bạn có thể hình dung, nhiệm vụ phân loại như vậy là rất khó. Nhưng nhận dạng giọng nói cũng có mặt đơn giản của nó. Ngôn ngữ của con người rất đều đặn. Chúng ta nên xem xét các quy tắc này khi thực hiện nhận dạng giọng nói. Đầu tiên, mỗi ngôn ngữ có một số đặc điểm về âm thanh. Trong trường hợp tiếng Việt, mỗi tiếng đều có một cách phát âm riêng. Chúng ta có thể cải thiện độ chính xác của nhận dạng giọng nói bằng cách sử dụng các đặc điểm âm thanh của tiếng Việt. Thứ hai, biểu hiện ngôn ngữ tiếng Việt cũng có các quy tắc nhất định. Tùy vào ngữ cảnh mà một từ có thể hiểu theo nhiều nghĩa khác nhau..

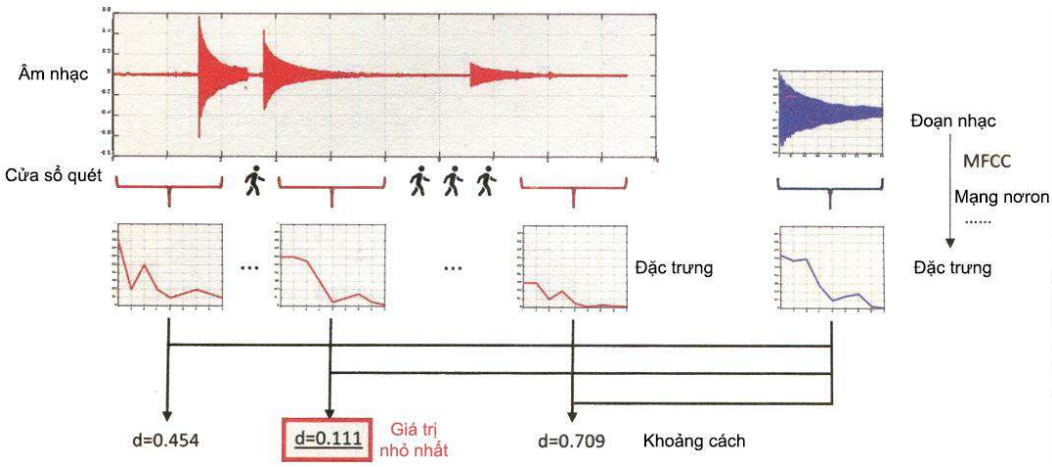
Đầu tiên, chia một đoạn lời nói thành các phân đoạn nhỏ. Quá trình này được gọi là đóng khung. Mỗi khung sau đó được gọi là một trạng thái và các trạng thái được kết hợp thành các âm vị. Kết hợp các trạng thái vào âm vị ngữ âm thường được gọi là chữ cái đầu và kết thúc. Trạng thái là một đơn vị ngữ âm chi tiết hơn so với âm vị, và một âm vị thường chứa ba trạng thái. Quá trình chuyển đổi một loạt các khung ngữ âm thành nhiều âm vị sẽ tận dụng lợi thế của các thuộc tính âm thanh của ngôn ngữ, vì vậy phần này được gọi là **mô hình âm thanh** (acoustic model). Quá trình từ ngữ âm đến văn bản yêu cầu sử dụng các biểu thức ngôn ngữ để chọn các từ chính xác trong các từ đồng âm và tạo thành câu có ý nghĩa. Phần này được gọi là **mô hình ngôn ngữ** (language model).

Suy nghĩ và thảo luận: Độ chính xác của nhận dạng giọng nói là gì?

Độ chính xác của nhận dạng giọng nói liên quan chặt chẽ đến mô hình âm thanh và mô hình ngôn ngữ. Ví dụ, một hệ thống nhận dạng giọng nói, mô hình âm thanh của nó có thể mô tả các đặc tính phát âm của ngôn ngữ chung, còn mô hình ngôn ngữ có thể mô tả sự diễn đạt ngôn ngữ của các chủ đề. Nếu hệ thống nhận dạng giọng nói được sử dụng để xác định tiếng nói của người thông báo để phát sóng tin tức thì tỷ lệ chính xác cao. Nhưng nếu sử dụng nó để xác định giọng nói với giọng điệu diễn dịch trong các bài phát biểu cổ điển, độ chính xác của nó sẽ thấp hơn nhiều.

4.4. Công nghệ phục hồi âm nhạc

Có một tính năng thú vị trong ứng dụng âm nhạc là tìm thấy bài hát tương ứng dựa trên phân khúc được người dùng hát, đó là tìm kiếm nhạc. Đầu vào cho nhiệm vụ truy xuất này thường là một đoạn nhạc ngắn và đầu ra là bài hát có nhạc giống đầu vào nhất có trong cơ sở dữ liệu. Trong phần này, chúng ta tìm hiểu một cách đơn giản để thực hiện điều đó.



Hình 4-12: Cửa sổ quét và tính toán khoảng cách

Nhiệm vụ truy xuất nhạc giống như chức năng "tìm" mà chúng ta sử dụng khi chỉnh sửa tài liệu. Chúng ta cũng có thể sử dụng cùng một ý tưởng để tìm clip nhạc đầu vào trong tất cả các bản nhạc. Nếu tìm thấy ca khúc muốn, chắc chắn đó là bản nhạc cần tìm. Nhưng không giống như tra cứu chính xác trong tài liệu, tra cứu ở đây rất mơ hồ, chẳng hạn như cùng một ca khúc có thể được hát bởi các ca sĩ khác nhau, mặc dù tương tự nhưng không giống nhau.Vì vậy, ta không thể đánh giá "tìm" hoặc "không tìm thấy" một cách trực tiếp mà xét trên sự giống nhau.

Chúng ta thường đo lường sự giống nhau cùng với khoảng cách. Khoảng cách càng gần, độ tương đồng càng lớn. Nhìn lại khoảng cách đã học trong Chương 2, với hai bộ đặc trưng $x = (x_1, x_2, x_3), y = (y_1, y_2, y_3)$, khoảng cách của chúng là:

$$d = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2}$$

Với khái niệm khoảng cách, chúng ta có thể tra cứu sự giống nhau.Như trong Hình 4-12, các đoạn có độ dài của clip nhạc được cắt tuần tự theo thứ tự thời gian trên nhạc. Khoảng thời gian giữa các đoạn liền kề có thể lớn hoặc nhỏ, và nó thường được đảm bảo rằng chúng có một chồng chéo

lớn trong thời gian. Quá trình này được gọi là "quét cửa sổ".Sau đó, tính toán các tính năng của phân khúc và các đoạn văn và tính khoảng cách của chúng, lấy khoảng cách tối thiểu của khoảng cách này làm khoảng cách giữa phân đoạn âm nhạc và nhạc. Bài hát cuối cùng với khoảng cách nhỏ nhất từ clip nhạc là kết quả của tìm kiếm.

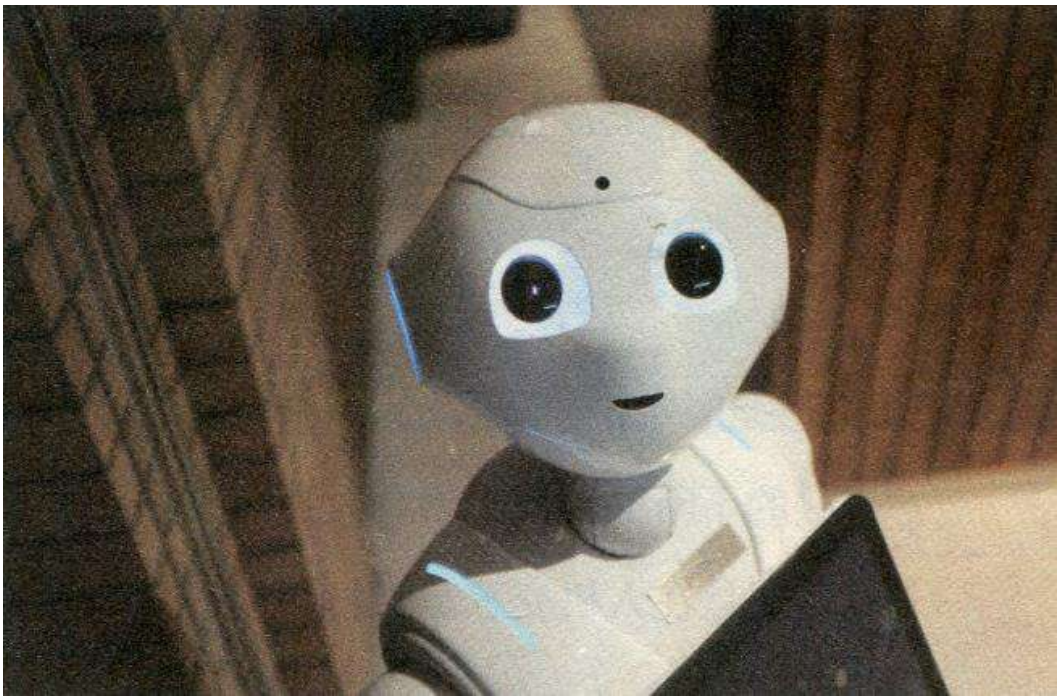
4.5. Tóm tắt chương này

Thông qua các nghiên cứu, chúng ta đã hiểu được đặc điểm của âm thanh và học cách làm cho máy tính cảm nhận và hiểu âm thanh; thông qua việc số hóa âm thanh, máy tính có thể "lắng nghe" âm thanh; thông qua việc tính toán phổ, máy tính có thể hiểu được tông màu và giai điệu của âm thanh; đặc trưng MFCC là tái tổng hợp phổ, và máy tính có thể thể hiện các đặc điểm quan trọng của âm thanh như formant với các vectơ kích thước nhỏ hơn.

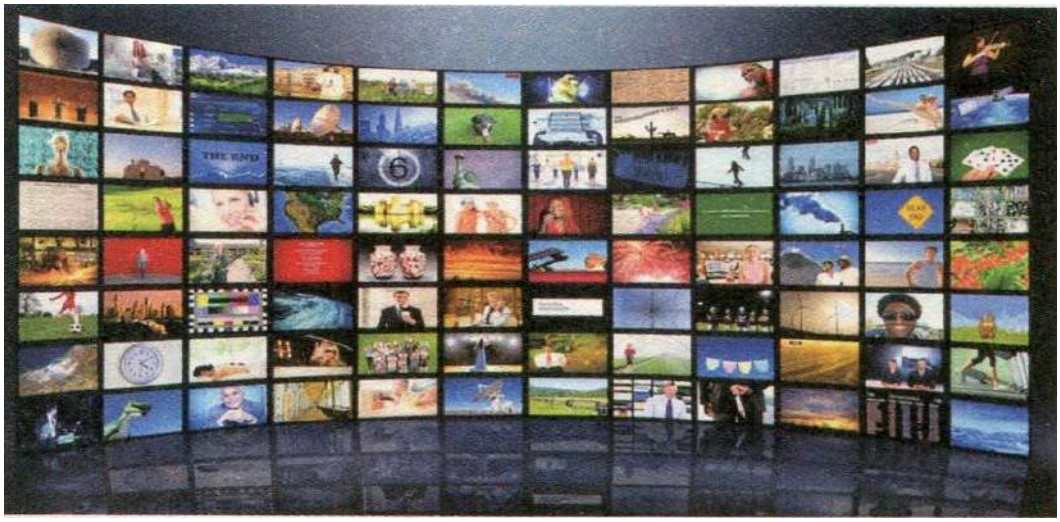
Với sự giúp đỡ của các mạng nơron, chúng ta đã tạo ra được một bộ phân loại tổng quan các thể loại nhạc của một đoạn nhạc. Chúng ta cũng nhận thấy mô hình âm thanh và mô hình ngôn ngữ là cánh tay phải của nhận dạng giọng nói. Ứng dụng truy xuất nhạc cuối cùng đã giúp ta hiểu sâu hơn về khoảng cách.

Âm thanh rất phong phú và đầy màu sắc, công nghệ xử lý âm thanh là vô tận. Các công nghệ và ứng dụng khác vẫn đang chờ chúng ta khám phá.

Chương V. Hiểu về video



Một robot tên VBOT gia nhập gia đình Minh, khiến cho Minh ấn tượng bởi trí thông minh nhân tạo của chú. VBot không chỉ thông minh, mà còn rất hiểu biết. Khi Minh từ trường học về nhà, VBot ngay lập tức chuyển chế độ chào đón và mang đến một cốc nước nóng; khi Minh đắm mình trong nghiên cứu, VBot luôn bật chế độ im lặng “học” và không làm phiền Minh. Minh chưa thể hiểu tại sao chú robot có thể hiểu hành vi của mình như thế nào và không thể không ngưỡng mộ sự thông minh đó.



Trong các chương trước, chúng ta đã dạy máy tính cách nhận biết hình ảnh và hiểu âm thanh. Trong chương này, chúng ta sẽ dạy máy tính hiểu về video.

Qua nhiều năm, số lượng video trên Internet đã phát triển, nội dung video ngày càng trở nên phong phú và việc áp dụng công nghệ video ngày càng trở nên phổ biến. Khi đối mặt với các tài nguyên video rộng lớn, làm sao để máy tính tự động phân tích chính xác nội dung để chúng ta có thể sử dụng nó? Sự hiểu biết video đã trở thành một xu hướng nóng trong lĩnh vực thị giác máy tính. Từ đặc trưng optical flow đến đặc trưng quỹ đạo, từ các phương pháp truyền thống đến học tập sâu, sự xuất hiện của các phương pháp mới đã thúc đẩy sự phát triển của công nghệ hiểu video. Ngày nay, công nghệ hiểu video đã đạt được những kết quả thú vị trong nhiều lĩnh vực như phân tích nội dung video, giám sát video, robot thông minh và tương tác giữa con người với máy tính.

5.1. Tĩnh và động: Từ hình ảnh đến video

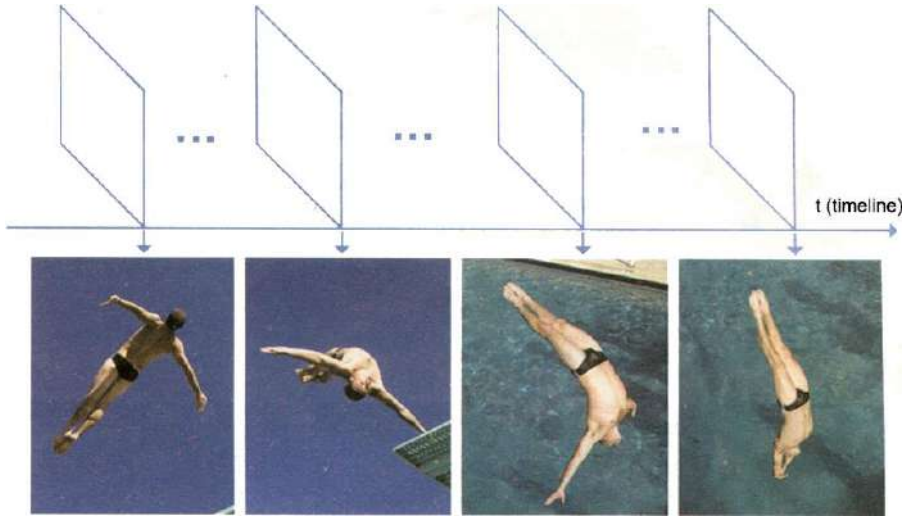
Trên TV, chương trình thể thao yêu thích của Minh được chiếu, và VBot cũng lặng lẽ theo dõi.

"VBot, bạn có hiểu không?" Minh hỏi.

"Tất nhiên, hình ảnh môn lặn đang chiếu trên TV."

"Sao lại là hình ảnh được?" Minh hơi uể oải. “Trên TV rõ ràng là hoạt động thể thao.”

Trong ra, VBot đã đúng. Video trong mắt của con người thực chất là các bức ảnh phát liên tục. Lý do tại sao chúng ta có thể thấy sự chuyển động trong bức tranh là "giả mạo" bởi hiện tượng lưu ảnh của mắt người. Video thực sự là một chuỗi gồm hàng trăm bức ảnh được chụp liên tiếp, mỗi bức ảnh được gọi là một khung hình (frame) của video. Để thấy rõ những thay đổi trong hình, Hình 5-1 hiển thị ví dụ 4 khung hình tiêu biểu. Khi hàng trăm hình ảnh được phát ở tốc độ hơn 24 khung hình / giây, ảnh gốc vẫn có thể chuyển động mà không ảnh hưởng đến cơ chế lưu ảnh. Tĩnh và năng động, một hành động môn nhảy cầu được trình bày trước chúng ta.



Hình 5-1: Thành phần của video là các hình ảnh liên tục

Kiến thức bổ sung: Lưu ảnh

Lưu ảnh (persistence of vision) là một cơ chế của mắt người: sau khi ánh sáng được chiếu xạ vào võng mạc, nó có thể được giữ lại trong một khoảng thời gian, tạo ấn tượng về sự liên tục của bức tranh.

Chúng ta thấy video ở mọi nơi trong cuộc sống của mình: phim trong rạp chiếu phim, chương trình trên TV, video được lưu trữ trên DVD, video có thể phát trực tuyến trong ứng dụng dành cho thiết bị di động và hơn thế nữa. Chúng ta đã học được cách các hình ảnh được biểu diễn trong các máy tính trong Chương 3, vậy còn video được biểu diễn như thế nào?

Trong máy tính, video là hình ảnh được sắp xếp theo thứ tự thời gian. Khi phát video, chỉ cần hiển thị hình ảnh theo thứ tự ở một tốc độ nhất định để hiển thị hình ảnh video đang chuyển động. So với biểu đồ của hình ảnh ba chiều, chúng ta có thể coi video có một chiều - chiều thời gian. Do đó, chúng ta có thể sử dụng hàm $I(x, y, t)$ để biểu diễn thông tin của video, trong đó t là thời gian tương ứng với khung video và x, y là vị trí (tọa độ hai chiều) của video tương ứng với pixel. Cách biểu diễn này liên kết chặt chẽ video và hình ảnh, cho phép ta sử dụng nhiều kỹ thuật trong lĩnh vực hình ảnh để nghiên cứu video. Khi đã hiểu cách biểu diễn của video, sau đây chúng ta bắt đầu tìm hiểu cách nhận biết hành vi trong video.

5.2. Mắt đại bàng: Nhận diện hành vi video

Hành vi (behavior) là một loạt các hành động xảy ra khi con người thực hiện một công việc. Nhận dạng hành vi trong video là quá trình mà máy tính phân tích dữ liệu video để xác định hành vi của con người. Hành vi bao gồm một loạt các hành động, máy ảnh ghi lại các hành động theo thứ tự và sử dụng làm đầu vào cho nhiệm vụ nhận dạng hành vi, còn đầu ra của nhận dạng hành vi là tên của hành vi đó trong một tập hợp cho trước. Nhận dạng hành vi video, cũng như phân loại hình ảnh, là một vấn đề cơ bản trong lĩnh vực thị giác máy tính.

Hinh 5-2: Nhận diện hành vi video

Nhận dạng hành vi video có giá trị ứng dụng quan trọng trong nhiều lĩnh vực. Ví dụ, trong lĩnh vực tương tác giữa con người và máy tính, nhận dạng hành vi có thể làm cho hệ thống tương tác máy hiểu chính xác hơn về hành vi của con người, do đó đưa ra một phản ứng chính xác; trong lĩnh vực giám sát video, nhận dạng hành vi có thể xác định các hành vi đặc biệt và bất thường trong video giám sát, giúp giảm khối lượng công việc của cảnh sát; trong việc lập chỉ mục video dựa trên nội dung, nhận dạng hành vi có thể tự động phân loại video dựa trên những gì xảy ra với các ký tự trong video.

Các khó khăn trong nhận dạng hành vi

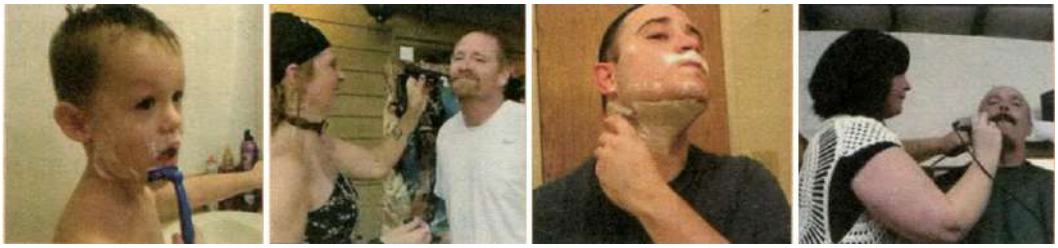
Hành vi của con người là một quá trình rất phức tạp, làm cho máy tính khó hiểu. Hơn nữa, các yếu tố như khoảng cách, ánh sáng, góc và sự ăn khớp khi quay video cũng có thể có tác động đáng kể đến nhận dạng hành vi video.

Nhìn chung, độ khó của nhận dạng hành vi video chủ yếu bao gồm các khía cạnh sau.

Đầu tiên, sự khác biệt nội bộ trong hành vi là lớn. Sự khác biệt trong nội bộ nghĩa là sự khác biệt lớn giữa các hành vi của cùng một loại. Như thể hiện trong *Hình 5-3*, hành vi "cạo râu" của những người khác nhau không giống nhau. Làm thế nào để có được một máy tính trích xuất các đặc điểm phổ biến từ một loạt các hành vi như vậy là một nhiệm vụ đầy thử thách.

Thứ hai, sự thiếu rõ ràng trong định nghĩa hành vi dẫn đến việc thiếu sót trong miêu tả video. *Hình 5-4* cho thấy một đoạn video của một bữa ăn trong một tập hợp dữ liệu, nhưng xen kẽ với hành vi của việc cho ăn, dẫn đến một sự thiên vị trong sự hiểu biết của máy tính về hành vi ăn uống.

Cuối cùng, nền môi trường và những khác biệt lớn là rất lớn. Như trong *Hình 5-5*, cùng là việc xem TV nhưng nền của môi trường video được lấy từ các góc khác nhau rất khác nhau: một số video xuất hiện màn hình TV, một số khác thì không.



Hình 5-3: Sự khác biệt lớn trong cùng một loại

Hình 5-4: Định nghĩa hành vi không rõ ràng

Hình 5-5: Môi trường nền có sự khác biệt lớn

Ngoài ra, số lượng mẫu dữ liệu hành vi ngày nay rất hạn chế. Ví dụ, trong một cơ sở dữ liệu hành vi thường được sử dụng UCF101, chỉ có 13320 video hành vi được thu thập từ các trang web video YouTube, bao gồm 101 danh mục; so với bộ dữ liệu hình ảnh ImageNet có hơn 14 triệu hình ảnh gồm hơn 20.000 danh mục. Những thách thức như vậy đóng góp lớn vào động lực phát triển. Trong quá trình vượt qua những khó khăn này, công nghệ nhận dạng hành vi video đang dần được cải thiện.

Đặc trưng quan trọng của nhận diện hành vi: Chuyển động

Sau khi học từ các chương trước, chúng ta đều biết rằng việc lựa chọn các tính năng sẽ có tác động lớn đến độ chính xác của phân loại. Trong phân loại hoa, chúng ta trích xuất chiều dài và chiều rộng của cánh hoa là một đặc trưng quan trọng để phân biệt các loài khác nhau của hoa diên vĩ, nếu chọn màu của cánh hoa làm đặc trưng, khó có thể phân biệt các loại hoa khác nhau dựa trên đặc trưng này. Làm cách nào để thiết kế các đặc trưng tốt hơn để nhận dạng hành vi video?

Trước hết, chúng ta hãy nghĩ về loại thông tin mà con người sử dụng để đánh giá hành vi của một người trong cuộc sống. Như thể hiện trong hình 5-6, một số sinh viên trong lớp giáo dục thể chất đang thực hành nhảy cao, và một số sinh viên đang thực hành nhảy xa. Bạn có thể xác định loại hành vi vì các sinh viên nhảy cao và nhảy xa có các quá trình khác nhau: Một là nhảy lên, nâng đầu gối, nhấc chân, băng qua xà ngang, và chân rơi xuống; hai là để nhảy về phía trước, cả hai chân cong về phía trước trước khi chạm đất. Có

thể thấy rằng **chuyển động** (motion) là một đặc trưng quan trọng trong đánh giá hành vi.

Hình 5-6: Sơ đồ nhảy cao và nhảy xa

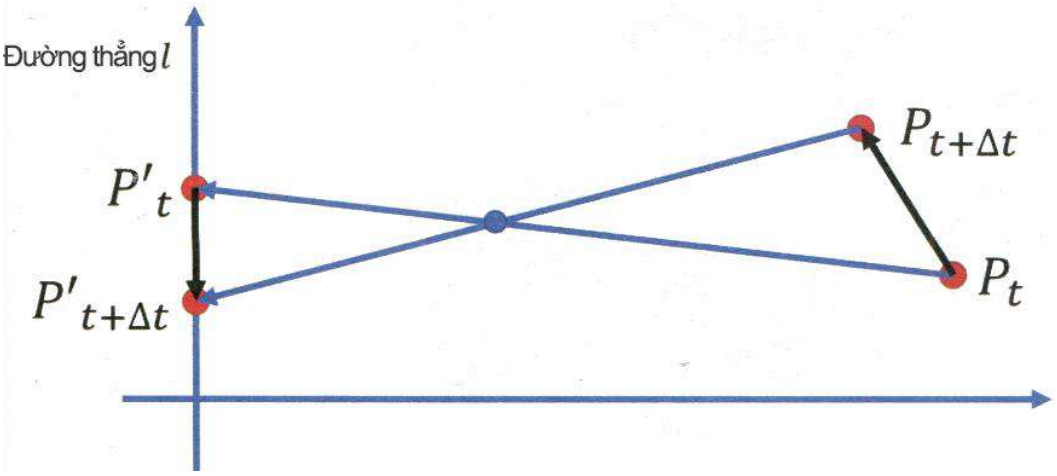
Đặc điểm của chuyển động: optical flow

Cơ sở quan trọng để phân biệt các hành vi hành vi khác nhau là động tác. Chúng ta nên trích xuất thông tin chuyển động trong video như thế nào? Đối với con người chúng ta, nhận ra sự chuyển động của mục tiêu trong không gian ba chiều khá đơn giản. Nhưng trong con mắt của máy tính, video chỉ là một chuỗi các khung hình. Nó không biết địa điểm của những người trong những hình ảnh này và loại chuyển động mà những mục tiêu này đang thực hiện. Điều này yêu cầu chúng ta thiết kế một thuật toán cho phép máy tính lấy được các đặc trưng chuyển động của cơ thể con người từ hình ảnh được tuần tự hóa.

Từ kiến thức vật lý, chúng ta biết rằng trong không gian ba chiều của thế giới thực, đại lượng vật lý như sự dịch chuyển và vận tốc có thể được sử dụng để mô tả chuyển động của một điểm trong không gian từ vị trí này sang vị trí khác. Trong xử lý video, chúng ta sử dụng optical flow để mô tả chuyển động. Cụ thể, optical flow mô tả chuyển động của điểm chiếu từ chuyển động trên không gian ba chiều thành hình ảnh hai chiều. Vì chúng ta đang xử lý dữ liệu hình ảnh hai chiều sau khi chuyển động được chụp, chúng ta chỉ có thể mô tả gián tiếp chuyển động ba chiều trong thế giới thực bằng chuyển động của điểm chiếu hai chiều.

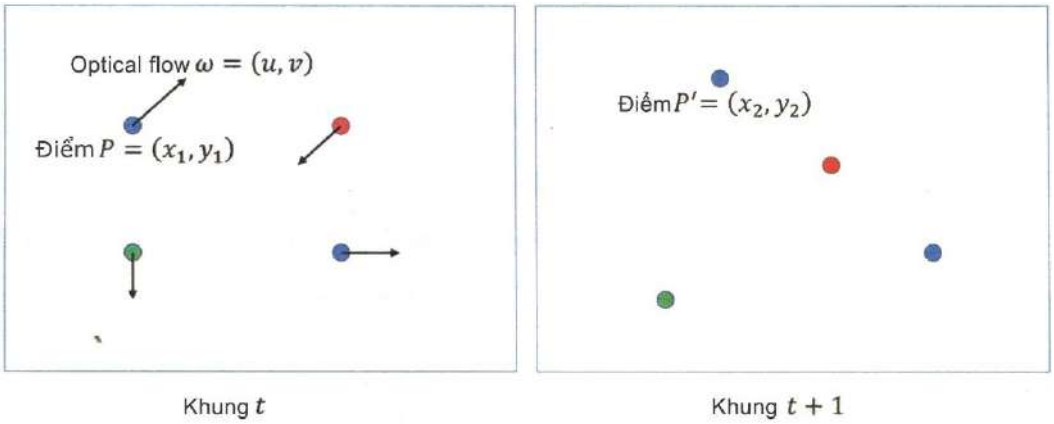
Để hiểu được optical flow là gì, chúng ta có thể đưa ra trường hợp chiếu một điểm chuyển động trong mặt phẳng hai chiều lên một đường thẳng một chiều làm ví dụ. Như thể hiện trong *Hình 5-7*, tại thời điểm t, một điểm được đặt tại điểm P_t trong mặt phẳng hai chiều. Sau khi chụp bằng camera, điểm chiếu P'_t của nó được lấy trên đường một chiều l. Sau thời gian Δt, nó

di chuyển đến $P_{(t+\Delta t)}$, và điểm chiếu trên đường thẳng di chuyển đến $P'_{(t+\Delta t)}$. Vecto $\overrightarrow{P'_t P'_{t+\Delta t}}$ mô tả chuyển động của điểm chiếu của điểm trên đường thẳng l, mô tả xấp xỉ trạng thái chuyển động của điểm trong mặt phẳng hai chiều thực. Khi khoảng thời gian dt đủ nhỏ, vector $\overrightarrow{P'_t P'_{t+\Delta t}}$ có thể được xem như là dịch chuyển tức thời của điểm chiếu, đó là thứ chúng ta gọi là optical flow.



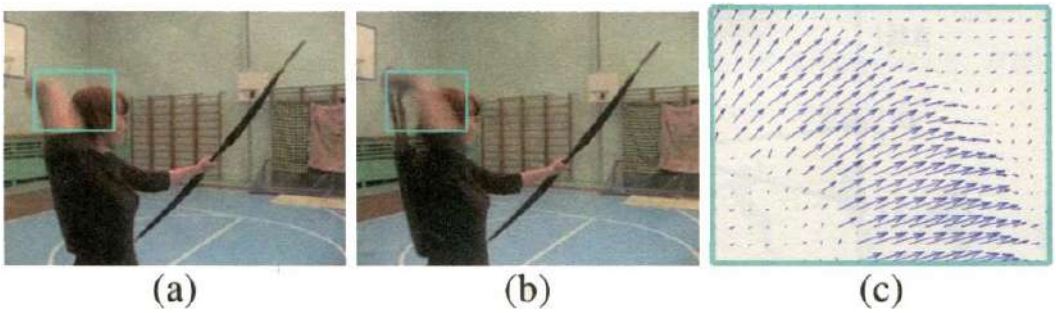
Hình 5-7: Chiếu sự di chuyển của điểm trong mặt phẳng 2 chiều lên đường thẳng một chiều

Hãy xem cách tính optical flow trong video. *Hình 5-8* cho thấy hai khung liền kề $I\,(x, y, t)$ và $I\,(x, y, t+1)$. Vì optical flow là sự dịch chuyển của cùng một điểm trong hai khung liền kề nên chìa khóa để tính toán optical flow là khớp với các điểm giống nhau giữa hai khung hình. Để có thể tìm các điểm tương ứng với nhau, chúng ta cần hai giả định chính:(1) Chuyển động của các đối tượng trong hai khung liền kề là tương đối nhỏ;(2) Màu sắc của hai khung liền kề về cơ bản không thay đổi. Với hai giả định này, chúng ta biết rằng vị trí, màu sắc và độ sáng của pixel trong hình ảnh sẽ không thay đổi nhiều sau khi di chuyển từ t đến $t+1$. Tức là, đối với điểm pixel $P = (x_1, y_1)$ ở thời điểm t nằm trong khung $I\,(x, y, t)$, chúng ta chỉ cần tìm điểm $P' = (x_2, y_2)$ ở vị trí tương ứng ở thời điểm $t+1$ khung $I(x, y, t\,1)$ có màu phù hợp với màu pixel P và quan sát P' để xem chuyển động P ở đâu. Sau khi có được điểm tương ứng, optical flow ω tại điểm giữa đầu tiên P có thể được tính toán: $\omega: (u, v) = (x_2, y_2) - (x_1, y_1)$.



Hình 5-8: Tính toán optical flow dựa trên hai khung liền kề

Trong các ứng dụng thực tế, việc ước tính optical flow cũng cần phải xem xét nhiều yếu tố khác, chẳng hạn như tắc nghẽn, thay đổi ánh sáng và bị mờ do chuyển động. Do đó, việc áp dụng cụ thể optical flow là phức tạp về mặt kỹ thuật. Như trong *Hình 5-9*, (a) và (b) là hình ảnh của hai khung liền kề. Chúng ta chọn phần của hình chữ nhật màu xanh để tính toán lưu lượng quang học, và vector optical flow cho mỗi điểm được biểu thị bằng một mũi tên trong Hình (c). Trong đó, hướng của mũi tên là hướng chuyển động của điểm ảnh, và kích thước của mũi tên là sự dịch chuyển của chuyển động của điểm ảnh. Như có thể thấy từ sơ đồ optical flow này, nó mô tả chuyển động của vật thể sang phía trên bên phải.

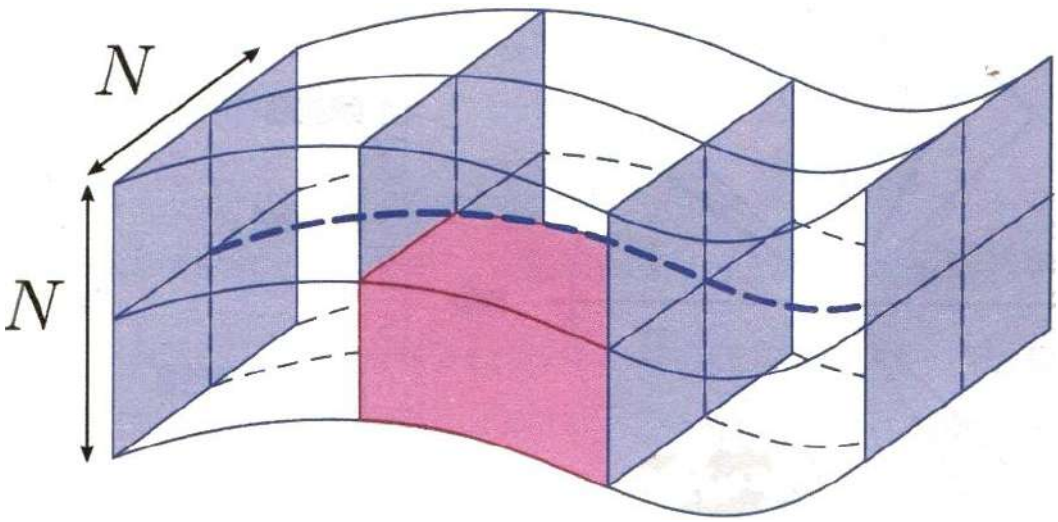


Hình 5-9: (c) là sơ đồ optical flow được tính từ hai khung liền kề (a) và (b)

HOF

Trong Chương 2, chúng ta đã học về khái niệm HOG. Bằng cách phân loại thông tin gradient trong hình ảnh, HOG có thể đại diện cho thông tin đường viền của các đối tượng trong hình ảnh, do đó tạo điều kiện cho máy tính phân biệt các đối tượng trong hình ảnh. Tương tự, các nhà nghiên cứu

đã đề xuất đặc trưng HOF (Histogram Optical Flow). HOF tính thông tin optical flow trong video để biểu thị thông tin chuyển động của đối tượng, qua đó máy tính có thể phân biệt các hành vi trong video. Bây giờ chúng ta hãy khám phá các đặc trưng thực sự của HOF.



Hình 5-10: 12 đơn vị space-time

Đầu tiên, chúng ta viết thời điểm bắt đầu là t, chọn một điểm trong khung hình video của thời điểm này và ghi lại vị trí của nó là $P = (x, y)$. Trong mỗi khung hình video từ thời gian t đến thời gian $t + L$, chúng ta chặn vùng có kích thước $N \times N$ có tâm là điểm (x, y). Kết quả là có một space-time volume gồm L hình ảnh có cùng kích thước.

Sau đó, chúng ta tiếp tục chia space-time volume này: chia nó thành 4 khu vực nhỏ hơn trên mỗi hình ảnh với một lưới 2x2. Theo chiều thời gian, chúng ta chia nó thành 3 phần bằng nhau. Do đó, chúng ta có thể nhận được $12\ (-2 \times 2 \times 3)$ ô space-temporal như trong *Hình 5-10*.

Tiếp theo, bên trong mỗi ô, chúng ta tính optical flow tại mỗi vị trí pixel. Giả sử rằng optical flow tại điểm ảnh (x, y) trong ảnh là $\omega\ (x, y) = (u, v)$. Đây là một vectơ hai chiều trong đó u, v đại diện cho các thành phần optical flow theo trục x và trục y tương ứng.

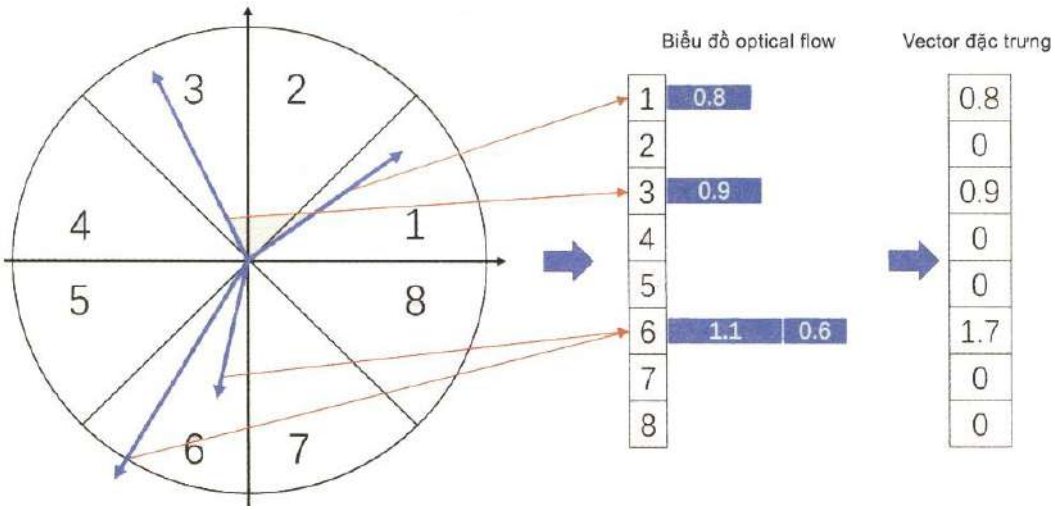
Optical flow tại điểm ảnh (x, y) là:

$$H(x, y) = \sqrt{u^2 + v^2}$$

Hướng của luồng ánh sáng tại điểm ảnh (x, y) là:

$$\theta(x, y) = \tan^{-1}\left(\frac{v}{u}\right)$$

Để tạo điều kiện thống kê, chúng ta chia phạm vi $[0°, 360°)$ trong hệ tọa độ hai chiều thành tám phần bằng nhau, mỗi phần trong số đó là một góc 45°. Như trong *Hình 5-11*, vectơ optical flow (u, v) tại tất cả các điểm ảnh trong một đơn vị space-time được vẽ trong hệ tọa độ trên theo kích thước và hướng. Sau đó, với "vector optical flow" chứa trong mỗi khu vực, các số liệu thống kê biểu đồ được thực hiện. Ví dụ, nếu sector có số 1 chứa một vector optical flow với kích thước là 0.8, thì thêm 0.8 vào vị trí đầu tiên trong HOF; và khu vực hình quạt có nhãn 6 chứa hai vectơ optical flow là 1.1 và 0.6 tương ứng, sau đó giá trị 1.7 (tức là 1.1 + 0.6) được thêm vào vị trí thứ sáu trong biểu đồ optical flow. Sau khi đếm vector optical flow trong mỗi sector, chúng ta thu được optical flow tương ứng với đơn vị space-time. Thông tin của biểu đồ được biểu diễn bằng một vectơ 8 chiều, vì vậy chúng ta có được vectơ đặc trưng 8 chiều của một đơn vị space-time.



Hình 5-11: Biểu đồ optical flow

Như chúng ta đã nói, một space-time volume chứa 12 đơn vị space-time. Đối với mỗi đơn vị, chúng ta có thể tính toán vectơ đặc trưng HOF 8 chiều theo cách này. Sau đó, chúng ta ghép 12 vectơ 8 chiều thành một vectơ 96 chiều (12x8) theo thứ tự nhất định, được sử dụng làm vectơ đặc trưng HOF tương ứng với toàn bộ space-time volume

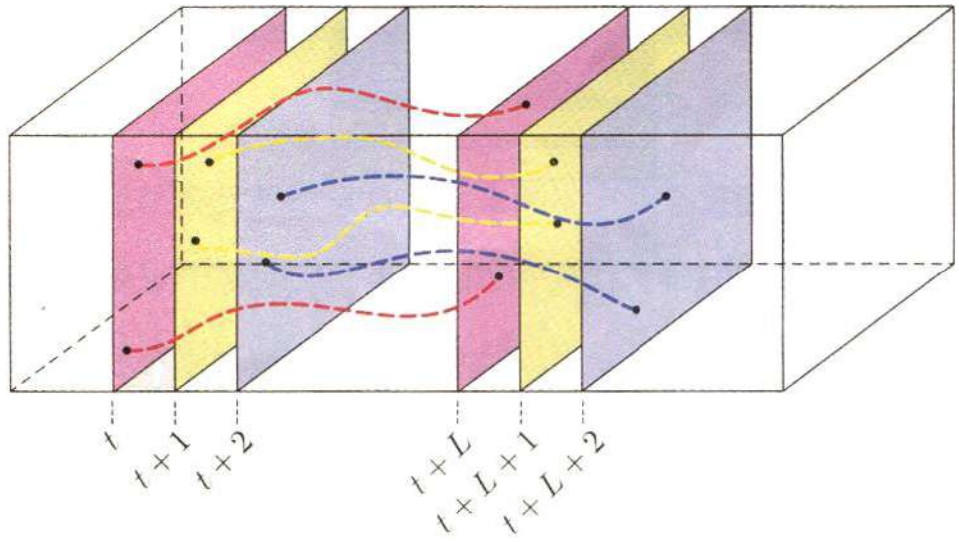
Suy nghĩ và thảo luận

Vừa rồi là một ví dụ đơn giản. Trong thực tế, số lượng vectơ optical flow trong một đơn vị space-time là nhiều hơn bốn. Tuy nhiên, ngay cả với nhiều vector dòng quang hơn, chúng ta vẫn tính toán theo cách này. Vector đặc trưng kết quả của space-time volume vẫn là 96 chiều. Tại sao lại như vậy?

Kiến thức bổ sung: Từ optical flow đến quỹ đạo dày đặc

Sự xuất hiện của một hành động thường kéo dài trong một thời gian dài từ đầu đến cuối. Chúng ta lưu ý rằng optical flow chỉ mô tả chuyển động của mục tiêu giữa hai điểm lân cận, trong khi HOF tương đối thô đối với phân đoạn thời gian. Những phương pháp này có những hạn chế tương đối lớn trong việc mô tả thông tin kích thước thời gian của chuyển động. Để mô tả chính xác hơn các chuyển động dài hạn, chúng ta cần kết hợp thông tin nhiều khung liên tục. Trong phần này, chúng ta học một đặc trưng mô tả trạng thái chuyển động của một đối tượng trong một khoảng thời gian - đặc trưng quỹ đạo.

Như thể hiện trong *Hình 5-12*, thiết lập tọa độ của một điểm trong hình ảnh khung thứ t là $P = (x, y)$. Chúng ta có thể sử dụng optical flow để có thể tính toán vị trí P_{t+1} của điểm này trong khung tiếp theo. Loạt sự thay đổi vị trí có thể mô tả một quá trình hành động, do đó chúng ta có thể sử dụng công thức $\Delta P_t = P_{t+1} - P_t = (x_{t+1} - x_t, y_{t+1} - y_t)$ để lần lượt tính toán sự dịch chuyển của điểm đặc trưng P trong khung L, và sau đó thu được vector $(\Delta P_t, \dots, \Delta P_{t+L-1})$. Chúng ta gọi vector có chiều dài $2 \times L$ này là quỹ đạo (trajectory), sử dụng để mô tả chuyển động của điểm đặc trưng P trong một khoảng thời gian.



Hình 5-12: Theo dõi chuyển động các điểm lấy mẫu trong L khung liên tiếp

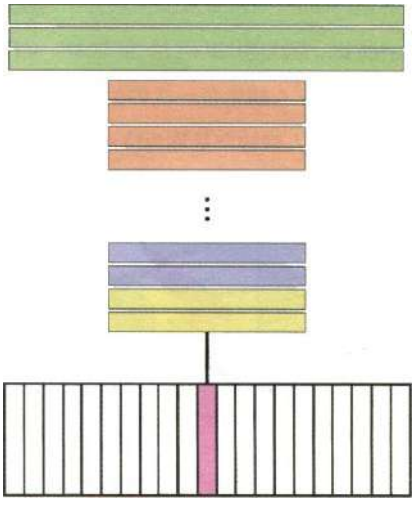
5.3. Nhận dạng hành vi video dựa trên học tập sâu

Như có thể thấy trong phần giới thiệu ở Chương 3, mạng nơ-ron tích chập (CNN) có thể trích xuất các đặc trưng của hình ảnh để nhận dạng hình ảnh, trong khi đó video bao gồm các khung hình liên tiếp. Làm thế nào để áp dụng CNN để nhận dạng hành vi video? Sau đây sẽ giới thiệu các phương pháp nhận dạng hành vi video chính trong những năm gần đây.

Phương pháp nhận dạng dựa trên khung hình đơn

Khi chúng ta không xét đến việc thay đổi thông tin hình ảnh trong video theo chiều thời gian, chúng ta có thể sử dụng khung hình trong video để thể hiện thông tin của toàn bộ video.

Hình 5-13 là sơ đồ nhận dạng hành vi video bằng cách sử dụng một phương pháp dựa trên khung đơn. Lớp dưới cùng trong hình thể hiện chuỗi các khung hình trong video. Ở đây, ta trích xuất ngẫu nhiên một khung hình nhất định của video để biểu diễn toàn bộ video và gửi nó đến CNN để xác định. Khi video mô tả hành vi tương đối tĩnh (chẳng hạn như xem TV hoặc viết một công việc), ta có thể đạt được kết quả tốt bằng cách chỉ sử dụng các đặc trưng hình ảnh của một khung để phân loại hành vi. Trong trường hợp này, sự khác biệt giữa các khung ảnh là không đáng kể và ảnh có thể đại diện cho hầu hết thông tin của video. Tuy nhiên, trong trường hợp di chuyển tương đối nhiều, nhận dạng hành vi cần phải kết hợp một loạt các hành động. Ví dụ, nhảy cao và nhảy xa có giai đoạn chạy đà giống nhau. Rất khó để phân biệt giữa các hình ảnh với một khung hình duy nhất, điều này sẽ dẫn đến tỷ lệ phân loại chính xác thấp hơn nhiều.

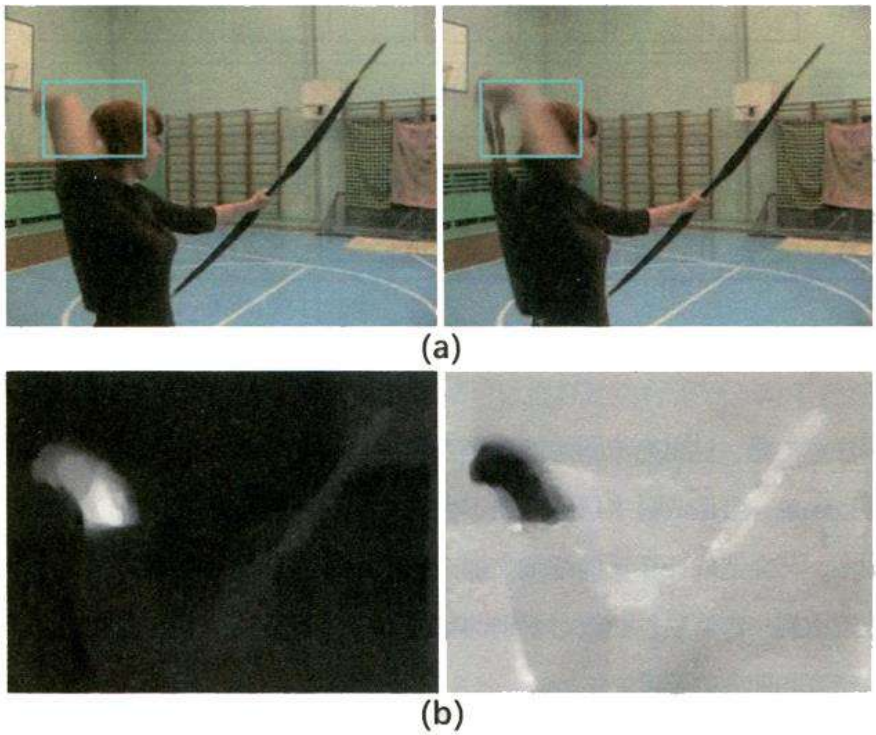


Hình 5-13: Sơ đồ nhận dạng hành vi khung đơn

CNN hai luồng

Như có thể thấy từ phần trên, khi chúng ta bỏ qua sự thay đổi của khung hình theo thời gian và coi video là ảnh tĩnh, thông tin tĩnh của video thu được và thông tin động bị mất sẽ ảnh hưởng đến độ chính xác của nhận dạng hành vi. Trong nhận dạng hành vi video, thu thập thông tin động, tức là biểu diễn video thời gian, trở thành yếu tố để cải thiện độ chính xác của nhận dạng. Điều này trở nên quan trọng hơn đối với các video thể thao. Vậy chúng ta mô tả thông tin chuyển động trong video như thế nào? Chúng ta có thể nghĩ đến việc sử dụng thông tin optical flow được giới thiệu trước đó, vậy trích xuất các đặc trưng chuyển động từ optical flow để nhận dạng hành vi như thế nào?

Trong *Hình 5-9 (c)* trước đó, chúng ta thấy rằng optical flow có hai thành phần trên mỗi pixel, thể hiện sự dịch chuyển theo hướng ngang và dọc. Như thể hiện trong *Hình 5-14 (b)*, nếu chúng ta lấy tất cả sự chuyển dịch theo phương ngang ra và sau đó sắp xếp các giá trị của chúng trong khoảng từ 0 đến 255, chúng ta sẽ nhận được một ảnh thang độ xám của optical flow theo chiều ngang. Tương tự, ta thu được hình ảnh thang độ xám optical flow theo chiều dọc. Ta có thể sử dụng hình ảnh thang độ xám ngang và dọc của optical flow làm đầu vào cho CNN để trích xuất các đặc trưng chuyển động trong video.

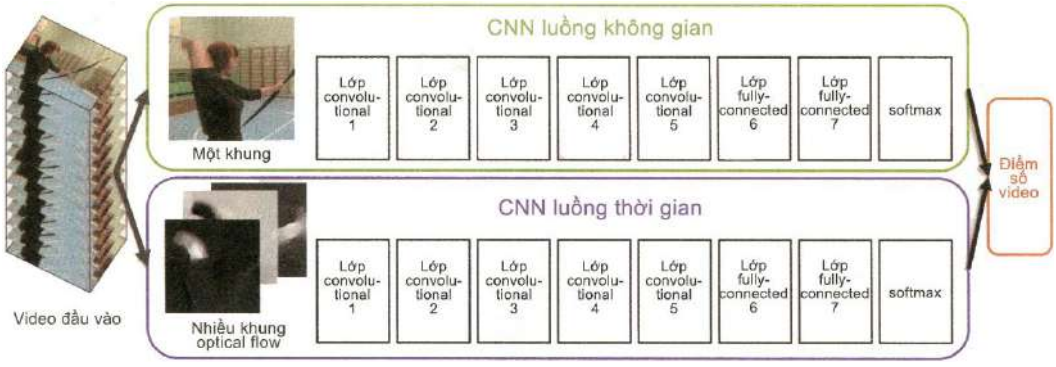


Hình 5-14: Các khung video liền kề và sơ đồ optical flow

(a) là hai khung video liền kề, (b) là sơ đồ optical flow và optical flow phương thẳng đứng của chúng

Chúng ta có thể chia thông tin video thành hai khía cạnh, tĩnh và động. Thông tin tĩnh đề cập đến sự xuất hiện của các đối tượng trong hình ảnh, bao gồm các cảnh và đối tượng liên quan, có thể thu được thông qua các khung hình tĩnh. Thông tin động đề cập đến thông tin chuyển động của một đối tượng trong chuỗi video, bao gồm chuyển động của người quan sát và đối tượng, có thể thu được bằng cách truyền trực tuyến hình ảnh đen trắng. CNN hai luồng (two-stream CNN) là việc sử dụng hai mạng khác nhau để đồng thời xử lý cả thông tin tĩnh và động, được sử dụng rộng rãi trong nhận dạng hành vi video. Như trong Hình 5-15, mạng có một khung hình màu được chọn ngẫu nhiên làm đầu vào được gọi là CNN luồng không gian (spatial stream CNN), và mạng có nhiều khung (chẳng hạn 10) làm đầu vào là CNN luồng thời gian (temporal stream CNN). Vì CNN hai luồng sử dụng hai CNN độc lập, sau khi chúng ta lấy được hai điểm số cho mỗi hành vi, chúng ta cần áp dụng phương pháp lấy trung bình theo thể loại hoặc phương pháp lấy giá trị lớn nhất, sau đó điểm số hành vi của hai luồng được hợp nhất.

Chúng ta biết rằng trong CNN hai luồng, CNN luồng thời gian trích xuất các đặc trưng từ sơ đồ optical flow đầu vào. Hãy suy nghĩ về các đặc điểm của HOF được đề cập trước đó. Cả hai đều trích xuất các đặc trưng từ optical flow. Vậy sự khác biệt trong phương pháp là gì? Trên thực tế, HOF là một thống kê có trọng số của hướng optical flow. Thông tin HOF thu được là đặc trưng thiết kế thủ công. Trong CNN hai luồng, optical flow được phân tách thành 2 sơ đồ optical flow ngang và dọc, và sau đó được gửi đến CNN để trích xuất các đặc trưng chuyển động. Điều này cho phép máy tính tự động tìm hiểu thông tin chuyển động trong sơ đồ optical flow. So với HOF được thiết kế thủ công, CNN có khả năng biểu diễn đặc trưng hiệu quả hơn, và trích xuất các đặc trưng có ý nghĩa cao từ các pixel cơ bản qua từng bước trong mạng, qua đó thực hiện nhận diện hành vi hiệu quả hơn.



Hình 5-15: Sơ đồ hai luồng nhận dạng hành vi

Cần lưu ý rằng việc xếp chồng sơ đồ optical flow là để nắm bắt thông tin chuyển động giữa các khung ảnh liền nhau trong chuỗi thời gian. Nếu số lượng khung đầu vào quá nhỏ, việc thu thập thông tin có thể không đầy đủ và không thể đại diện cho một chuỗi video dài. Nếu nhập quá nhiều khung hình, nó sẽ tăng số lượng phép tính. Vậy cần bao nhiêu khung hình là phù hợp? Kinh nghiệm thực tế cho thấy khi số lượng khung của sơ đồ optical flow được chọn đến một giá trị nhất định, số khung hình ít ảnh hưởng đến độ chính xác của nhận dạng, vì vậy ngưỡng này có thể được lựa chọn để cân bằng các yêu cầu về độ chính xác và tốc độ .

Xử lý video dài: Mạng phân đoạn chuỗi thời gian

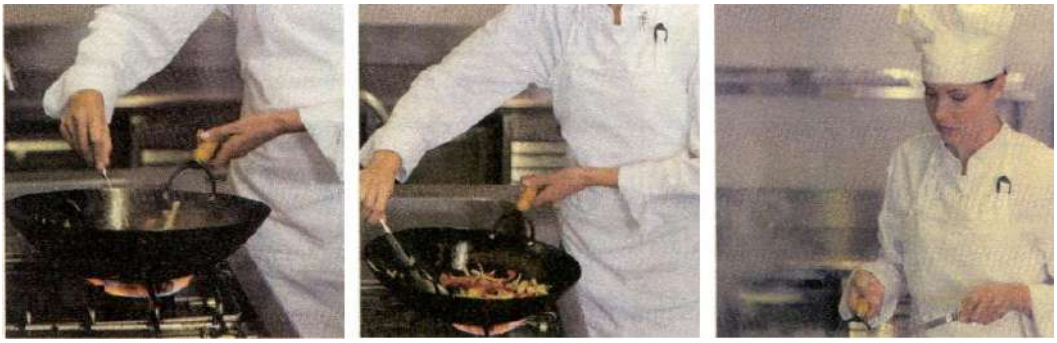
Đối với các video ngắn (khoảng 10 giây), CNN hai luồng có thể nhận dạng tốt. Để nhận dạng video dài, thách thức mà nó đặt ra là làm thế nào để lập mô hình thời gian dài hơn. Ở đây quyển sách giới thiệu một mạng nơron khác xử lý dữ liệu video dài (vài phút) - mạng phân đoạn thời gian (temporal segment networks - TSN). Nó được đề xuất để giải quyết vấn đề nhận dạng hành vi trong thời gian dài.

Để có được thông tin chuyển động trong một thời gian dài trong video, nếu việc lấy mẫu dày đặc (dense) được thực hiện trên luồng thời gian, cần có nhiều khung hình optical flow hơn để bao phủ đầu và cuối của chuyển động. Vấn đề với việc có quá nhiều optical flow là nó tạo ra quá nhiều tính toán và không thích hợp cho việc quảng bá ứng dụng. Ngoài ra, do sự dư thừa liên tục trong video, như trong *Hình 5-16*, các khung liền kề rất giống nhau. Nếu lấy mẫu dày đặc được thực hiện, các hình ảnh lấy mẫu sẽ rất giống nhau. Ta nghĩ đến chiến lược lấy mẫu thưa thớt (sparse), không chỉ tiết kiệm chi phí tính toán mà còn không bỏ qua thông tin quan trọng của video và khả thi hơn là lấy mẫu dày đặc.

Hình 5-16: Các khung hình giống nhau liên tục

Chiến lược lấy mẫu thời gian thưa thớt (Sparse temporal sampling) là chia dữ liệu thành các độ dài khác nhau theo chuỗi thời gian. Ví dụ, bất kể có bao nhiêu học sinh trong mỗi lớp, các chỗ ngồi được chia thành 6 nhóm. Nếu số lượng sinh viên tương đối lớn, số lượng người trong mỗi nhóm sẽ tăng lên tương ứng và ngược lại. Bằng cách trích xuất một đặc trưng từ mỗi đoạn, bạn có thể nhận được một đặc trưng có độ dài cố định, theo sau là một mạng xử

lý dữ liệu có độ dài cố định. Có nhiều cách để trích xuất đặc trưng từ phân đoạn video. Từ phần giới thiệu trước, nó có thể đơn giản như việc trích xuất một mẫu hoặc sử dụng CNN hai luồng. Mạng phân đoạn theo thời gian giống như có một nhóm sinh viên trong lớp gửi một tác phẩm, bạn có thể chọn một người hoàn thành, bạn cũng có thể phân công lao động cho từng người khác nhau để mọi người cùng nhau hoàn thành.

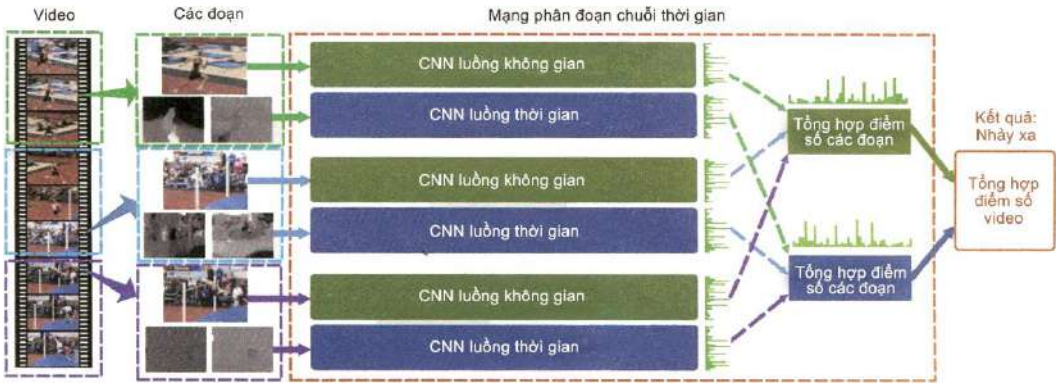


Hình 5-17: Ví dụ video dài

Đối với hành vi nấu ăn phức tạp, sẽ có một loạt các hành động liên tục, cũng có lúc chuyển đổi góc ống kính, đó là các video có hành vi dài hạn. Điểm số hành vi cuối cùng phải được kết hợp với hành vi của từng khoảng thời gian để có được danh mục hành vi của toàn bộ video.

Hình 5-18 là sơ đồ của mạng phân đoạn chuỗi thời gian. Phương pháp cụ thể là chia một video đầu vào thành nhiều đoạn theo thời gian (hình được chia thành ba phân đoạn). Sơ đồ optical flow liên tục và khung hình được chọn ngẫu nhiên từ mỗi phân đoạn, và mỗi phân đoạn phải sử dụng CNN hai luồng; cuối cùng, điểm danh mục của ba phân đoạn được hợp nhất để có được danh mục hành vi của toàn bộ video. Mỗi phân đoạn đóng góp vào dự đoán của video và mỗi phân đoạn đầu tiên tạo lớp hành động dự đoán sơ bộ của riêng mình và sau đó kết hợp các dự đoán của từng phân đoạn để có được dự đoán toàn bộ về video.

Chìa khóa để giải quyết nhận dạng video dài bởi mạng phân đoạn chuỗi thời gian là phân đoạn video dọc theo trục thời gian sao cho các mẫu được lấy mẫu có thể được phân phối đồng đều hơn trong toàn bộ khoảng thời gian. Vì vậy, mạng có thể mô phỏng toàn bộ cấu trúc dài hạn và mô hình có thể bao gồm toàn bộ video.

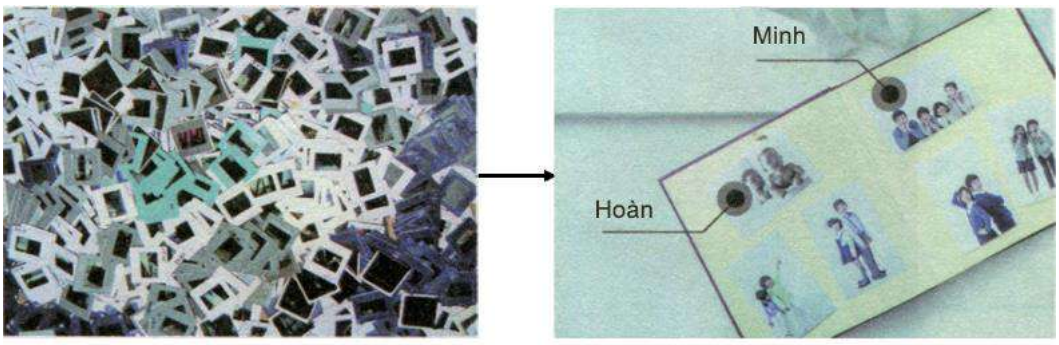


Hình 5-18: Sơ đồ của mạng phân đoạn chuỗi thời gian

5.4. Tóm tắt chương

Thông qua việc tìm hiểu nhận dạng hành vi video, chúng ta hiểu sự khác biệt và kết nối giữa video và hình ảnh, nhận ra tầm quan trọng của việc mô hình hóa cấu trúc thời gian của video và nắm vững cách trích xuất hiệu quả các đặc trưng theo thời gian và không gian của video. Cụ thể, sơ đồ optical flow và quỹ đạo mô tả các đặc tính chuyển động của hành vi trong video theo những cách khác nhau, do đó đặt nền tảng cho nhận dạng hành vi video. Trong phương pháp học sâu, chúng ta sử dụng CNN để trích xuất và biểu diễn cho đặc trưng optical flow của video và sử dụng hai luồng để kết hợp thông tin chuyển động video và thông tin hiển thị để phân loại. Cuối cùng, đối với nhiệm vụ nhận dạng hành vi trong video dài, chúng ta được giới thiệu ngắn gọn chiến lược lấy mẫu thời gian thưa thớt và mạng phân đoạn thời gian. Video tích hợp nhiều dạng thông tin khác nhau như hình ảnh, âm thanh và văn bản. Làm thế nào để tiếp tục phân tích và hiểu video kết hợp với các khía cạnh khác nhau vẫn còn để được khám phá và nghiên cứu.

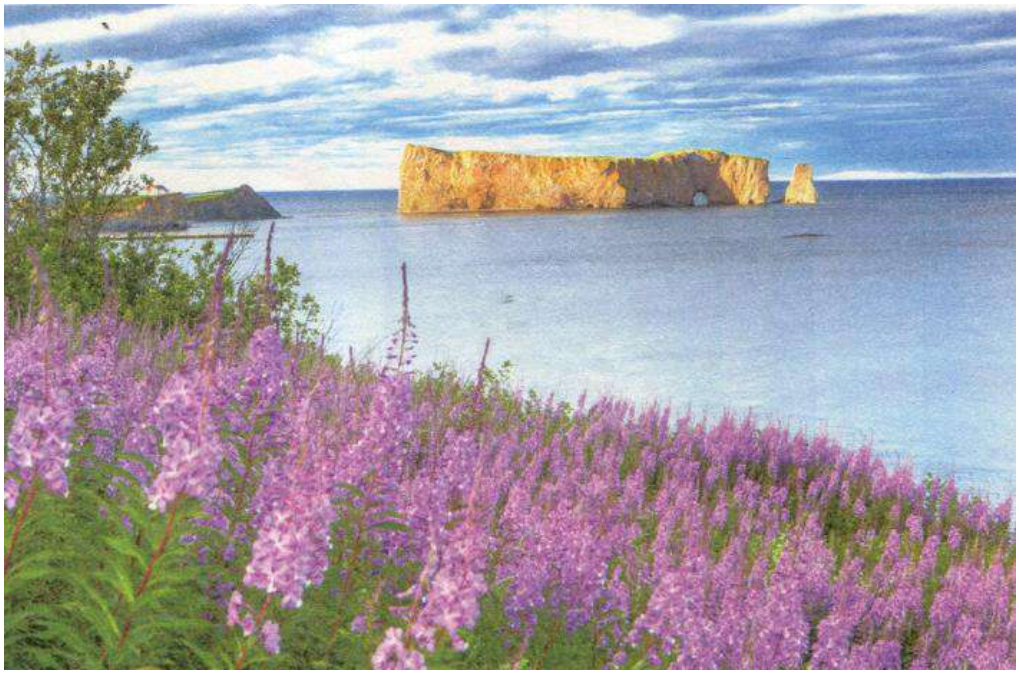
Chương VI. Tự học: Phân loại



Ảnh Album ảnh hoàn thành

Từ ngày Minh được sinh ra, cha bắt đầu ghi lại cuộc đời của Minh bằng một chiếc máy ảnh kỹ thuật số. Sau khi lớn lên, Minh thích mở máy ảnh cũ của bố, nhìn vào người cha và người mẹ trẻ tuổi, bản thân, và những người bạn nhỏ giờ không còn tiếp xúc nữa. Tuy nhiên, ngoài những người biết Minh, máy ảnh cũng được trộn với ảnh đồng nghiệp của bố, bạn bè của mẹ và những bức ảnh phong cảnh chụp bằng tay. Trong khi chọn nút lật trang để bỏ qua những bức ảnh nhàm chán này, Minh nghĩ: "Máy tính có thể tự động sắp xếp ảnh được không?"

6.1. Khi trí tuệ nhân tạo chưa bao giờ nghe tên loài hoa



Khoảng 100 năm trước, 150 bông hoa diên vĩ trên Bán đảo Gaspar ở Canada được đo bởi nhà thực vật học Edgar Anderson. Ông đã cẩn thận ghi lại sự xuất hiện và sự đa dạng của 150 bông này. Dữ liệu này đã được chuyển xuống cho đến ngày nay và đã trở thành "cuốn sách giác ngộ" được sử dụng bởi nhiều trí tuệ nhân tạo khi họ lần đầu tiên nhận ra thế giới. Trong Chương 2, đó là dữ liệu mà bộ phân loại học cách phân biệt các loại hoa.

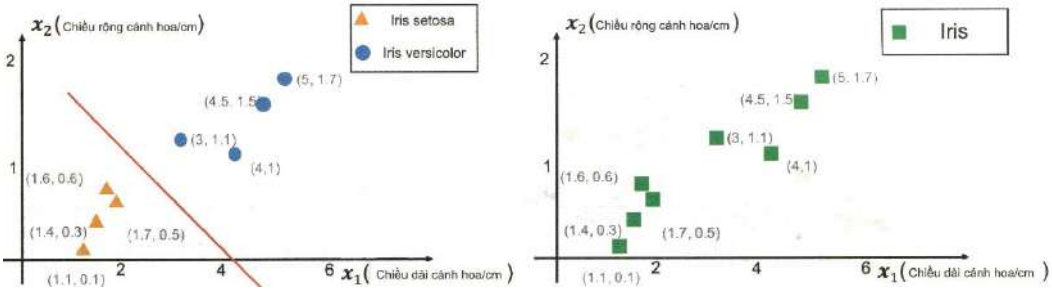
Có thể lập luận rằng kiến thức của bộ phân loại hoa iris xuất phát từ kiến thức thực vật học của Anderson. Vì bộ phân loại nằm dưới sự hướng dẫn của thông tin ghi nhãn do Anderson cung cấp, nó được học để phân biệt các loài iris. Quá trình học tập này, đòi hỏi thông tin chú thích cho dữ liệu huấn luyện, được giám sát học tập. Việc phân loại hoa hình ảnh, âm thanh và video trong các chương trước cũng yêu cầu thông tin chú thích cho các danh mục. Tất cả đều thuộc về việc học có giám sát.

Mặc dù lịch sử nhân loại có niên đại hàng triệu năm, nhưng cho đến ba trăm năm trước, chúng ta mới có một hệ thống phân loại khoa học hoàn chỉnh. Tác giả của hệ thống này là Carl von Linné, cha đẻ của phân loại hiện đại. Trong một thời đại khi một số lượng lớn các sinh vật không được đặt tên, Linné đã viết trong ghi chú của mình: "Tại sao có một mối quan hệ thú vị giữa

các sinh vật và sinh vật, giữa các sinh vật và thiên nhiên?" Nếu trí thông minh nhân tạo một mình đạt đến bán đảo Gaspar, liệu nó có thể khám phá mối quan hệ giữa những sinh vật như Linné tò mò; nếu trí tuệ nhân tạo chưa nghe thấy tên của các loại iris, nó có thể phân loại một cách độc lập không?

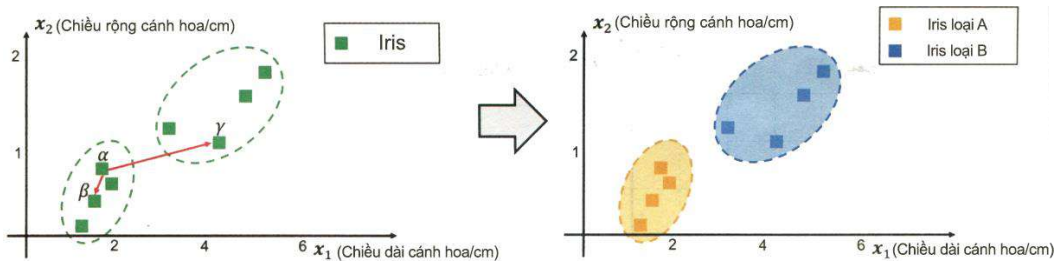
Khác với nhiệm vụ phân loại hoa trong Chương 2, khi phân loại các sinh vật, trí tuệ nhân tạo sẽ không có thông tin về danh mục.

Chúng ta gọi quy trình học tập mà không có thông tin được chú thích này là học không giám sát. *Hình 6-2* bên trái cho thấy dữ liệu hoa trông như thế nào trong trí thông minh nhân tạo khi học được giám sát. Dưới sự hướng dẫn của thông tin về danh mục của hoa, chúng ta có thể dễ dàng tìm thấy một đường thẳng tối ưu, tách không gian đặc trưng thành hai, sao cho 2 loại hoa ở 2 phần. Hình 6-2 bên phải cho thấy việc học không giám sát. Có thể thấy rằng nếu không có hướng dẫn về thông tin thể loại, rất khó để xác định loài hoa nào có cùng giống, chưa kể đến việc sử dụng một đường thẳng để phân loại hoa iris.



Hình 6-2: Phân loại hoa trong mắt trí tuệ nhân tạo: Hình ảnh bên trái là học tập có giám sát và bên phải là học tập không giám sát.

May mắn thay, mặc dù không có hướng dẫn, chúng ta biết rằng nếu hoa của cùng một loài, chiều rộng cánh hoa và chiều dài cánh hoa của chúng phải giống nhau để hai điểm mẫu trong không gian đặc trưng có khả năng là cùng một loại hoa iris. Như có thể thấy từ hình bên trái trong *Hình 6-3*, hoa trong không gian đặc trưng có thể được nhóm thành hai cụm. Hai mẫu thuộc cùng một cụm, chẳng hạn như alpha và beta, có chiều rộng cánh hoa và chiều dài cánh hoa tương tự nhau. Hai mẫu thuộc về các cụm khác nhau, chẳng hạn như alpha và gama, có chiều dài cánh hoa rất khác nhau, vì vậy có thể thuộc về hai giống khác nhau. Như thể hiện trong hình bên phải của *Hình 6-3*, chúng ta gọi 2 loại là loại A và loại B.



Hình 6-3: Phân loại được thực hiện trên sự tổng hợp các hoa trên không gian đặc trưng

Có thể thấy bằng cách phân tích tổng hợp dữ liệu trong không gian đặc trưng, một nhóm dữ liệu cũng có thể được chia thành các lớp khác nhau. Chúng ta gọi loại phương thức này là **phân cụm** (clustering). Clustering được thiết kế để chia một nhóm các mẫu thành nhiều bộ sao cho các phần tử trong cùng một tập hợp giống như "tương tự" nhau. Một giả định quan trọng về phân cụm là hai mẫu trong không gian đặc trưng tương tự nhau thì rất có thể thuộc cùng một loại. Giả định này không nhất thiết phải đúng trong tất cả các dữ liệu, chúng ta nên đặc biệt chú ý đến điều này khi sử dụng các thuật toán phân cụm. Là một quá trình học tập không giám sát, phân cụm không yêu cầu ghi nhãn danh mục dữ liệu, và thậm chí không cần định nghĩa trước các loại. Nó là một phương pháp phân tích rất thiết thực.

Trong chương này, trí thông minh nhân tạo sẽ giống như các nhà khoa học thực thụ, khám phá các quy luật và các giống khác nhau từ dữ liệu hoa.

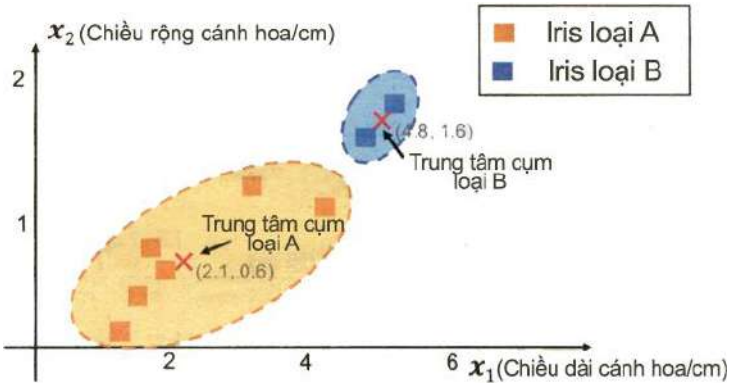
6.2. Sự tích lũy: Phân nhóm K-means các loài hoa

Bài toán: Có N bông hoa, tọa độ của bông hoa thứ n trong không gian đặc trưng

$$x_n = (a_n, b_n), n = 1,2,3, \dots, N$$

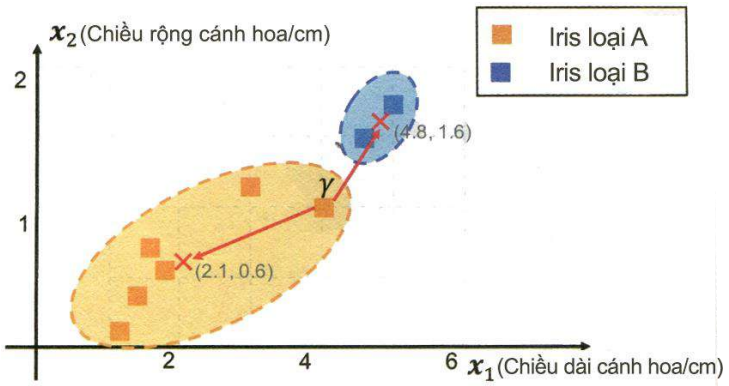
Trong số đó, a, b tương ứng đại diện cho chiều rộng cánh hoa và chiều dài cánh hoa của hoa iris thứ n. Chúng ta hy vọng trí tuệ nhân tạo có thể phân chia N bông hoa thành K loại dù không biết đến sự đa dạng của chúng, khiến những mẫu cùng loại sẽ có những đặc trưng giống nhau, và những mẫu khác loại thì ít giống nhau. Dưới đây quyển sách sẽ giới thiệu một phương pháp phân cụm. Ý tưởng chính của nó là bắt đầu từ một nhóm bất kỳ và dần dần đạt được các mục tiêu trên thông qua điều chỉnh.

Đối với phương pháp phân chia thể hiện trong *Hình 6-4*, tính chiều rộng trung bình và chiều dài trung bình của mỗi loại cánh hoa iris, chúng ta có thể xác định trung tâm của mỗi cụm (cluster center), chúng nằm trong dấu đỏ trong hình. Vì trung tâm cụm được xác định bởi các đặc trưng trung bình của một loại hoa, nó có thể được sử dụng như một đại diện của loại hoa này. Khoảng cách từ hoa đến điểm trung tâm của loại hoa đó càng nhỏ, thì nó càng giống với hoa của loại hình này, và có nhiều khả năng nó thuộc về loại này.



Hình 6-4: Phân cụm mỗi loại hoa tại trung tâm

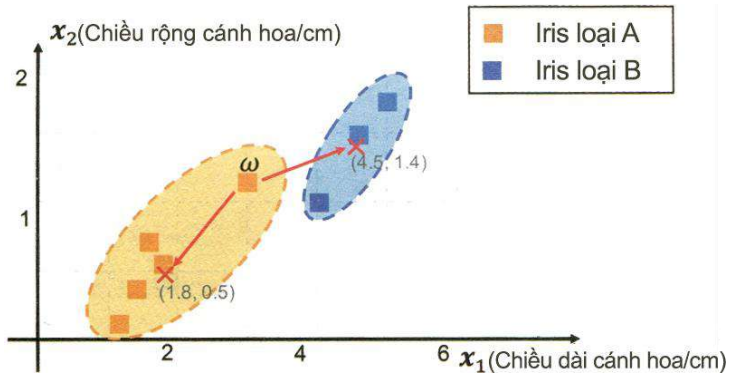
Nhìn vào *Hình 6-5*, chúng ta thấy rằng mẫu γ thuộc loại A nhưng gần với trung tâm phân nhóm B hơn, chỉ ra rằng nó giống với hoa B hơn. Điều này trái với mục tiêu phân cụm: tính tương đồng của các đặc trưng của cùng một loại mẫu là cao, và sự giống nhau của các đặc trưng của các mẫu tương tự là thấp. Giải pháp rất đơn giản. Bạn có thể phân loại mẫu γ thành lớp B.



Hình 6-5: Mâu thuẫn của mẫu γ

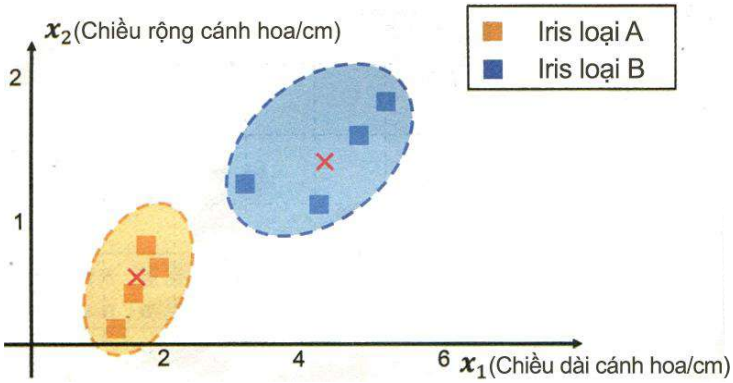
Do sự thay đổi trong cách phân chia, trung tâm phân nhóm của danh mục đã thay đổi và chúng ta tính toán lại trung tâm phân cụm. Kết quả được thể hiện trong *Hình 6-6*. Tuy nhiên, trong bộ phân loại mới, chúng ta đã tìm

thấy một mâu thuẫn mới: Mẫu ω của lớp A gần với trung tâm cụm của lớp B hơn. Để cải thiện hơn nữa kết quả phân loại, chúng ta phân loại mẫu này thành lớp B.



Hình 6-6: Mâu thuẫn của mẫu ω

Như trong *Hình 6-7*, sau nhiều lần điều chỉnh, cuối cùng chúng ta cũng có được cách phân chia thỏa đáng, tất cả các mẫu đều đáp ứng được mục tiêu phân cụm.



Hình 6-7: Kết quả phân cụm K-means

Phương pháp phân cụm ở trên là thuật toán K-means clustering. Trong thuật toán phân cụm này, cách phân vùng của các mẫu được biết để tính toán trung tâm phân cụm của từng loại mẫu. Ngược lại, một trung tâm phân cụm đã biết cũng có thể phân chia tốt hơn. Bằng cách cải thiện các điểm vừa và nhỏ và phương pháp phân chia theo chu kỳ, chúng ta có thể nhận được kết quả phân cụm tốt hơn và tốt hơn cho đến khi trung tâm cụm và chế độ phân chia không còn thay đổi nữa.

Vậy làm thế nào để bạn có được trung tâm cụm ban đầu? Đầu tiên, chúng ta cần xác định số K của các cụm, và sau đó chọn ngẫu nhiên K mẫu làm trung tâm của K cụm trong tất để hoàn thành việc khởi tạo trung tâm cụm. Thuật toán K-means clustering như sau:

Thuật toán phân cụm K-means:

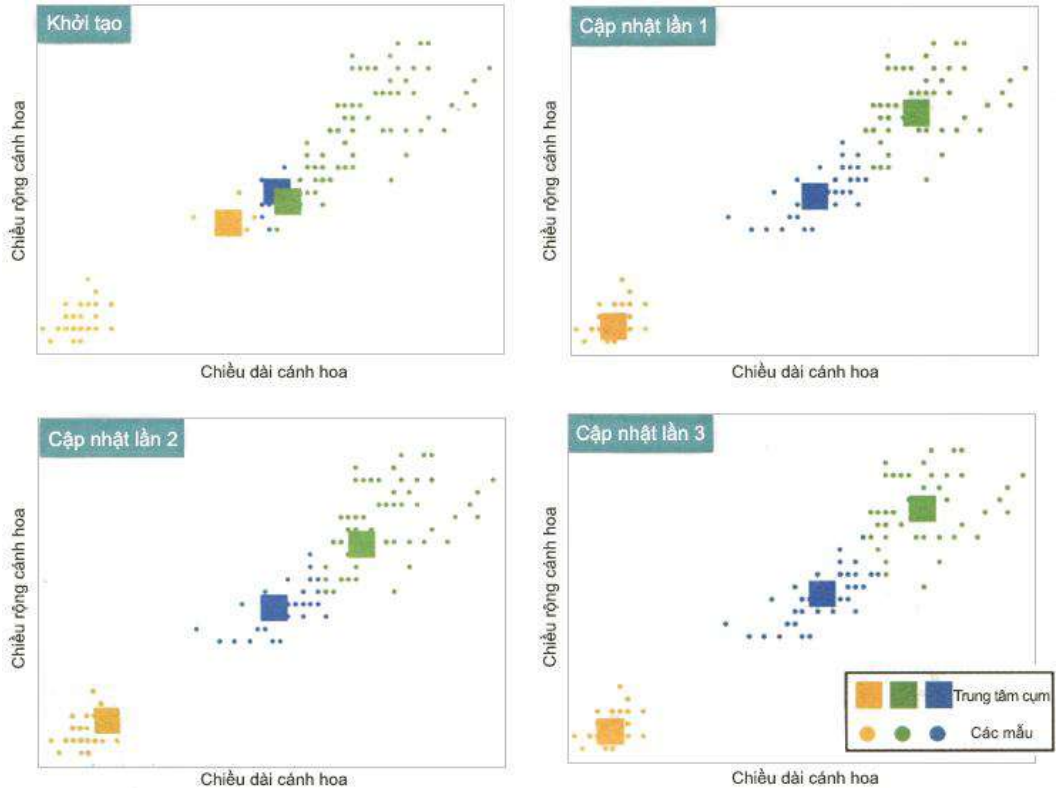
Bước 1: chọn ngẫu nhiên K mẫu từ tất cả các mẫu làm trung tâm phân cụm ban đầu cho K loại.

Bước 2: mỗi mẫu được phân chia vào cụm có trung tâm gần nhất và thu được danh mục tương ứng.

Bước 3: trung tâm cụm của từng loại mẫu được tính toán lại.

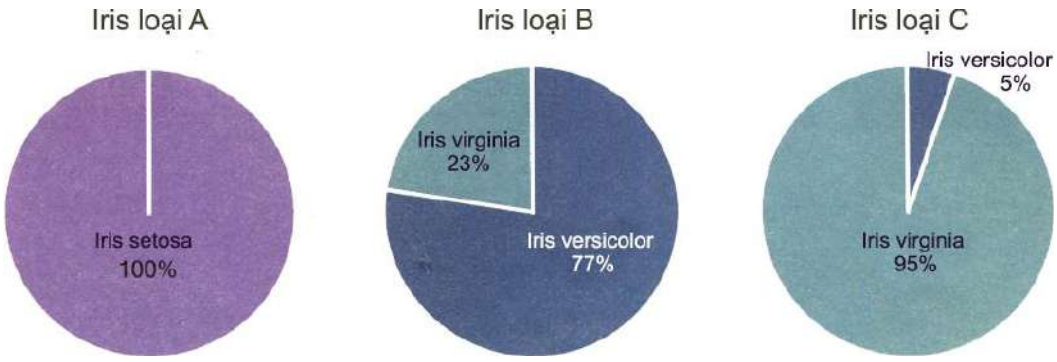
Lặp lại các bước thứ hai và thứ ba cho đến khi trung tâm cụm và chế độ phân chia không còn thay đổi.

Thuật toán phân cụm K-means được thể hiện bên dưới với tập dữ liệu hoa Iris. Trong chương thứ hai và các ví dụ trước, chúng ta chỉ sử dụng hai loại dữ liệu là hai loại hoa trong tập dữ liệu. Trong thực tế, bộ dữ liệu hoàn chỉnh cũng bao gồm thêm một loại Virginia iris nữa, tổng cộng ba loại. Để tăng độ khó của vấn đề, chúng tôi sử dụng cả ba loại dữ liệu (thông tin danh mục ẩn không được sử dụng) để phân cụm. Hình 6-8 cho thấy quá trình phân cụm K-means trong tập dữ liệu hoa dày đặc với K = 3. Vòng tròn đại diện cho mẫu iris, hình vuông biểu thị trung tâm cụm của mỗi loại hoa và màu sắc biểu thị các danh mục khác nhau được thu thập. Có thể thấy rằng sau ba lần cập nhật, hoa được chia thành 3 loại tương đối tập trung.



Hình 6-8: Quá trình hội tụ phân cụm K-means

Sau khi phân cụm, mẫu được phân thành 3 danh mục A, B, C. Tỉ lệ các loài hoa ở mỗi danh mục là:



Hình 6-9: Kết quả phân tích thành phần phân cụm K-means

Trong số đó, danh mục A bao gồm toàn bộ hoa iris loại 1, danh mục B chủ yếu là loại hoa thứ 2 và danh mục C chủ yếu bao gồm loại thứ 3 Virginia. Mặc dù ta không cung cấp thông tin về nhiều loại hoa, thuật toán phân cụm K-means vẫn tìm thấy sự tồn tại của ba loài hoa bằng cách phân tích các đặc trưng của các mẫu, và chia các mẫu ba loại. Đây là lần đầu tiên trí tuệ nhân tạo trong cuốn sách này không dựa vào kiến thức của con người, quan sát thế giới một cách độc lập và có câu trả lời của riêng mình.

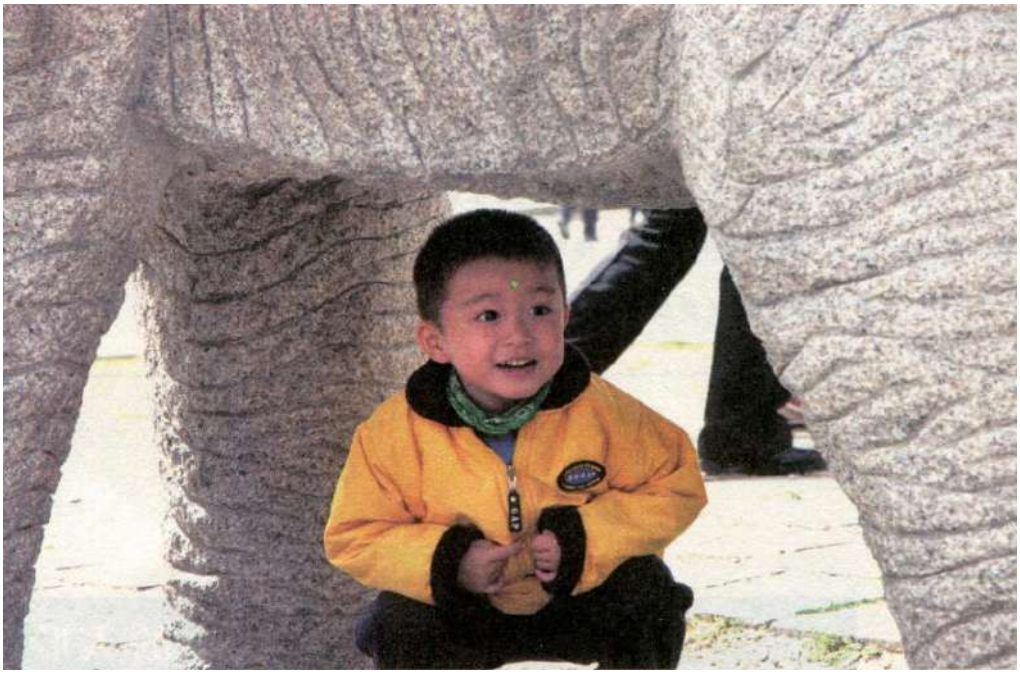
6.3. Chia nhóm người: Phân cụm khuôn mặt trong album

Minh đã có nhiều hình ảnh từ bé. Những bức ảnh quý giá này ghi lại những kỷ niệm của Minh và gia đình, bạn bè. Tuy nhiên, khi anh muốn nhớ lại kỷ niệm với ai đó, hoặc chỉ muốn chọn một bức ảnh cho mình để làm avatar, anh phải xem qua tất cả các bức ảnh và mất thời gian và công sức. Bây giờ, trí thông minh nhân tạo sẽ giúp Minh tổ chức album và tự động nhóm ảnh theo người có trong ảnh.

Hình 6-10: Quá trình phân cụm album

Trong phần trước, chúng ta coi hoa iris là điểm đặc trưng trong không gian đặc trưng. Thuật toán K-means sau đó được sử dụng để nhóm các đặc trưng của hoa. Tương tự, miễn là chúng ta có thể trích xuất các đặc trưng từ khuôn mặt trong ảnh, sử dụng các điểm đặc trưng trong không gian đặc trưng để biểu diễn từng khuôn mặt, ta có thể sử dụng thuật toán K-means để thu thập các khuôn mặt "tương tự" nhau.

Việc trích xuất các đặc trưng rất quan trọng đối với hiệu ứng phân cụm. Làm thế nào để có được những đặc điểm của khuôn mặt? Như trong *Hình 6-10*, với một album, mỗi khuôn mặt trong album được kiểm tra riêng biệt để phát hiện khuôn mặt, chỉnh sửa khuôn mặt và trích xuất đặc trưng, sau đó ta thu được đặc trưng khuôn mặt để phân cụm khuôn mặt. Dưới đây, chúng ta sẽ trích xuất các đặc trưng khuôn mặt của các bức ảnh được hiển thị trong *Hình 6-11* theo từng bước theo quy trình được thể hiện trong *Hình 6-10*.



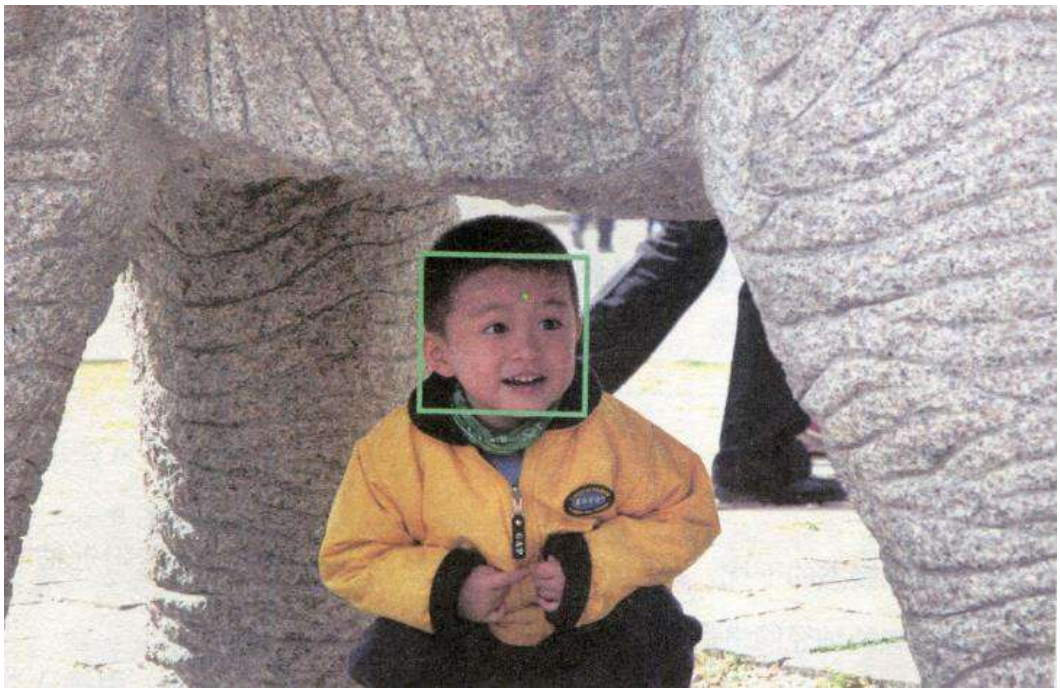
Hình 6-11: Ảnh của một người

Phát hiện khuôn mặt

Mục đích của phát hiện khuôn mặt là xác định vị trí của khuôn mặt trong một hình ảnh. Khi phân nhóm ảnh, chúng ta chỉ quan tâm đến khuôn mặt xuất hiện trong ảnh, vì đặc trưng "khuôn mặt" là thông tin quan trọng được sử dụng để xác định danh tính của người khác. Nền bức ảnh, trang phục của người mặc..., sẽ không liên quan và không ảnh hưởng đến việc

đánh giá. Do đó, như được hiển thị trong *Hình 6-12*, ta sử dụng phát hiện khuôn mặt được huấn luyện trước để tìm vị trí khuôn mặt trong ảnh. Trong các bước tiếp theo, chỉ có khu vực có chứa khuôn mặt được phân tích.

Cách phát hiện khuôn mặt được mô tả trong phần đọc mở rộng của Chương 2. Phát hiện khuôn mặt là một công nghệ đã hoàn thiện, nhiều thiết bị phát hiện khuôn mặt có hiệu năng vượt trội và khả năng chống nhiễu mạnh có thể được sử dụng trực tiếp.



Hình 6-12: Phát hiện khuôn mặt: Ô vuông màu xanh chỉ ra vị trí của khuôn mặt

Điều chỉnh khuôn mặt

Mục đích của việc điều chỉnh khuôn mặt là làm cho khuôn mặt với những tư thế khác nhau đều quay về phía trước. Trong phân cụm khuôn mặt, các bức ảnh khác nhau của cùng một người có đặc điểm càng giống nhau, thuật toán phân cụm để nhóm các ảnh này thành một nhóm càng dễ dàng hơn. Tuy nhiên, như trong *Hình 6-13*, mặc dù phát hiện khuôn mặt đã giúp ta thu hẹp diện tích phân tích xuống còn phần của khuôn mặt, hướng của khuôn mặt sẽ vẫn làm cho các bức ảnh của cùng một người trông rất khác nhau và ảnh hưởng tới kết quả trích xuất đặc trưng.

Hình 6-13: Các khuôn mặt với tư thế khác nhau

Để giải quyết vấn đề này, đầu tiên chúng ta tìm những điểm chính trên mặt người (như mắt, mũi, miệng). Sau đó, theo vị trí của các điểm chính, ta thực hiện các phép biến đổi hình học thích hợp (chẳng hạn như phóng to, kéo giãn và cắt) trên hình ảnh sao cho khuôn mặt quay về phía hướng mặt về phía trước. Hình 6-14 cho thấy quá trình hiệu chỉnh khuôn mặt và quay mặt. Có thể thấy khuôn mặt trong hình thay đổi từ xiên về phía trước để phía trước. Hiệu chỉnh khuôn mặt giúp chống lại sự can thiệp của các nét cử chỉ trên khuôn mặt.

Hình 6-14: Hiệu chỉnh và điều chỉnh khuôn mặt: Những điểm màu xanh lá cây là những điểm quan trọng.

Cũng giống phát hiện khuôn mặt, hiệu chỉnh và điều chỉnh khuôn mặt là những công nghệ hoàn thiện có thể được tự động hóa bởi máy tính.

Trích xuất đặc trưng

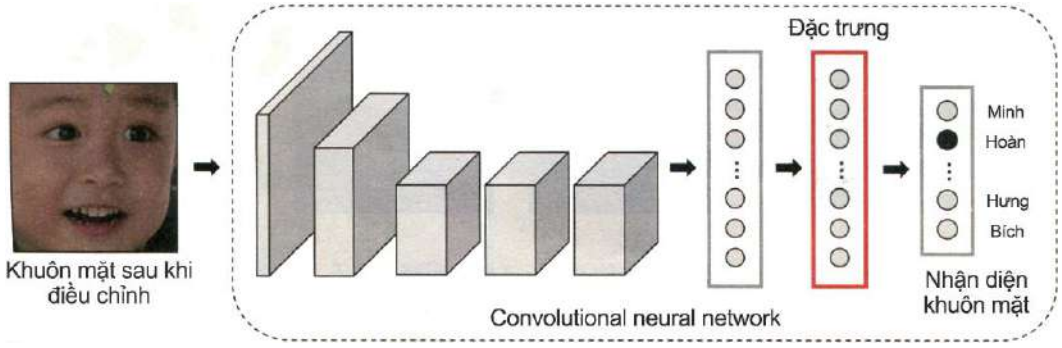
Bằng cách loại bỏ nhiễu tư thế khỏi khuôn mặt, chúng ta có thể sử dụng mạng nơron để trích xuất các đặc trưng gần mỗi điểm chính. Có mạng nơron tùy ý nào giúp chúng ta trích xuất các đặc trưng không?

Trong các chương trước, chúng ta đã học cách sử dụng các mạng nơron để phân loại. Trong quá trình huấn luyện, mỗi lớp của mạng nơron sẽ

tự động tìm kiếm các đặc trưng phù hợp nhất, tùy thuộc vào nhiệm vụ phân loại. Ví dụ, trong nhiệm vụ phân loại ảnh trong Chương 3, các đặc trưng được tìm thấy bởi mạng nơron chứa thông tin về danh mục mà phân biệt tốt nhất các đối tượng trong hình ảnh. Trong nhiệm vụ nhận dạng khuôn mặt, mạng nơron là cần thiết để xác định danh tính của chủ sở hữu khuôn mặt. Các đặc trưng mà nó tìm thấy có nhiều khả năng phân biệt giữa các khuôn mặt.

Trong nhiệm vụ phân cụm album ảnh, ta quan tâm nhiều hơn đến danh tính của những người trong ảnh chứ không phải các đối tượng xuất hiện trong ảnh. Vì vậy, chúng ta nên chọn mạng nơron cho việc nhận dạng khuôn mặt để trích xuất các đặc trưng từ khuôn mặt.

Hình 6-15 cho thấy quá trình trích xuất đặc trưng. Chúng ta gửi hình ảnh khuôn mặt vào CNN, lấy đầu ra của lớp áp chót làm đặc trưng mô tả khuôn mặt.



Hình 6-15: Trích xuất đặc trưng: Các đặc trưng được hiển thị trong đường viền màu đỏ là các đặc trưng đã được trích xuất.

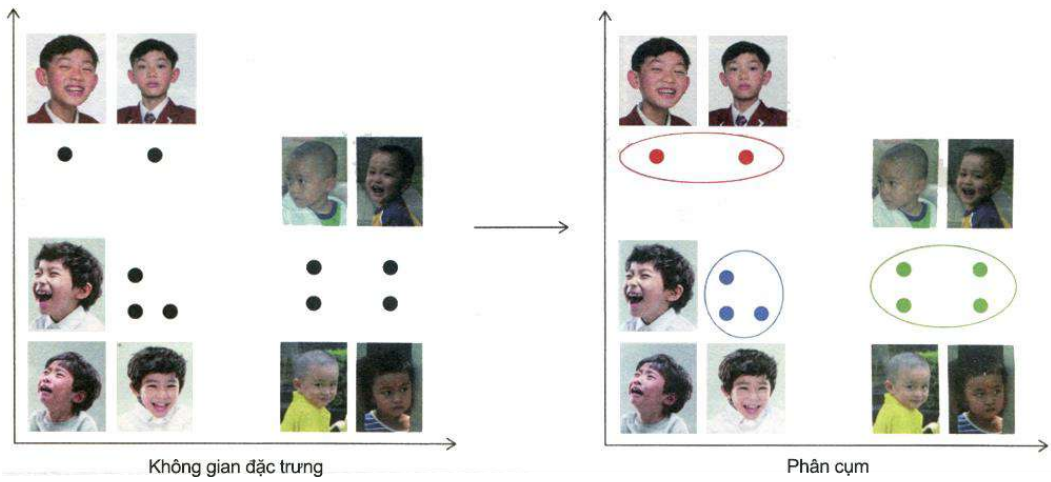
Lưu ý rằng mạng thần kinh được sử dụng ở đây để trích xuất các đặc trưng hình ảnh khuôn mặt là một mô hình được huấn luyện trước trên các bộ dữ liệu khác. Mặc dù mạng không nhìn thấy Minh và bạn bè của mình trong quá trình huấn luyện, các đặc trưng được trích xuất vẫn có thể được sử dụng để nhóm các khuôn mặt mới này. Điều này là do một mạng thần kinh tốt sẽ có khả năng promotion tốt hơn.

Suy nghĩ và thảo luận

Chúng ta thường chọn đầu ra của lớp áp chót của mạng thần kinh làm đặc trưng thay vì đầu ra của lớp cuối cùng. Tại sao vậy?

Phân cụm khuôn mặt

Sau khi trích xuất các đặc trưng khuôn mặt trong mỗi bức ảnh, chúng ta có thể thực hiện phân cụm K-means trên các khuôn mặt.



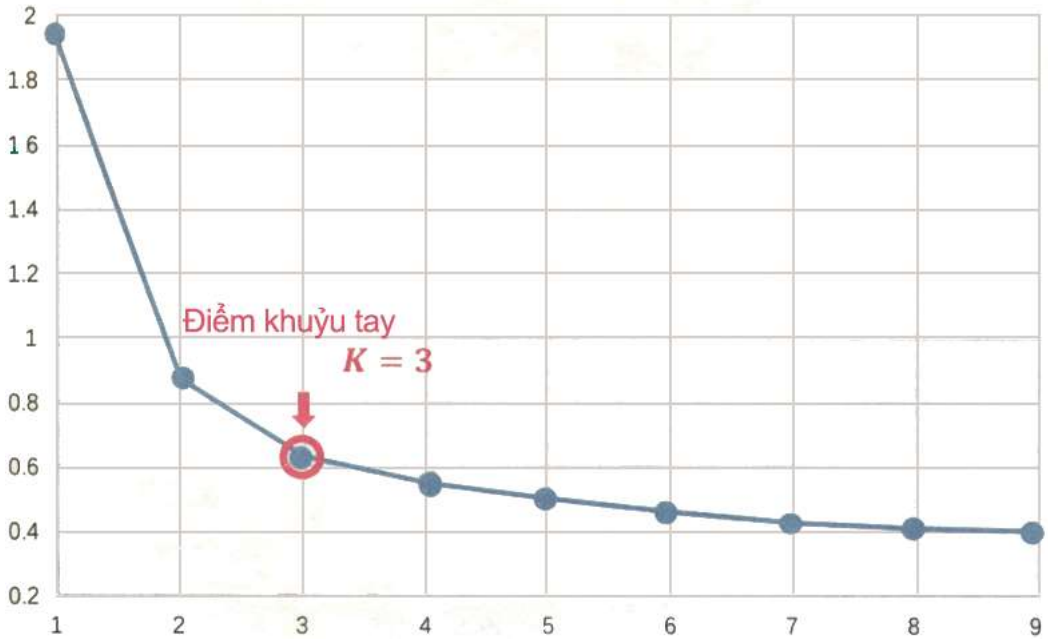
Hình 6-16: Phân cụm khuôn mặt

Như trong *Hình 6-16*, giống như phân cụm hoa, phân cụm khuôn mặt là quá trình khai thác các khuôn mặt giống nhau trong không gian đặc trưng. Thông qua phân cụm khuôn mặt, ta chia những khuôn mặt đã xuất hiện trong album thành một vài loại. Vì mỗi hình ảnh khuôn mặt được chụp từ một bức ảnh nhất định, ảnh được chia thành nhiều loại một cách tự nhiên. Lưu ý rằng một số ảnh chứa nhiều khuôn mặt, do đó, cùng một ảnh có thể được sắp xếp thành nhiều loại. Sử dụng máy tính để tự động sắp xếp các ảnh của cùng một loại, bạn có thể có được một bộ ảnh đẹp như trong *Hình 6-17*.

Hình 6-17: Kết quả phân cụm khuôn mặt trong album

Trước khi thực hiện phân cụm K-means, trước tiên chúng ta cần xác định số lượng của cụm là số K. Trong các cụm hoa, chúng ta có thể quan sát trực quan mức độ hội tụ của dữ liệu và đoán rằng hoa nên được chia thành nhiều nhóm. Tuy nhiên, khi phân cụm các album ảnh, chúng ta thường không biết rằng dữ liệu nên được chia thành bao nhiêu danh mục, đặc biệt là trong không gian dữ liệu nhiều chiều, rất khó để trực tiếp quan sát và đếm. Vì vậy, làm thế nào để xác định kích thước của K? Chúng ta sử dụng các giá trị K khác nhau để thực hiện phân cụm K-means, và khoảng cách trung bình giữa mỗi mẫu và trung tâm cụm tương ứng có các giá trị khác nhau khi thống kê.

Khoảng cách trung bình từ mẫu đến trung tâm cụm tương ứng có thể đo sự ảnh hưởng của phân cụm đến một mức độ nào đó. Như có thể thấy trong Hình 6-18, khi số lượng các cụm K tăng lên, khoảng cách trung bình liên tục giảm. Tuy nhiên, nếu số cụm quá lớn, ảnh sẽ được chia quá mỏng và mỗi danh mục chỉ chứa một vài ảnh, điều này sẽ mất tính thực tiễn của nó. Làm thế nào để cân bằng khoảng cách trung bình và số lượng các cụm? Như có thể thấy trong *Hình 6-18*, tại K = 3, đường cong tạo ra một điểm uốn đáng kể. Sau điểm uốn, khi K tăng, khoảng cách trung bình giảm rất chậm. Do đó, K = 3 tại điểm uốn là một lựa chọn phù hợp. Vì đường cong này có hình dạng tương tự với khuỷu tay, phương pháp này được gọi là phương pháp khuỷu tay (elbow method), và điểm uốn này được gọi là điểm khuỷu tay.

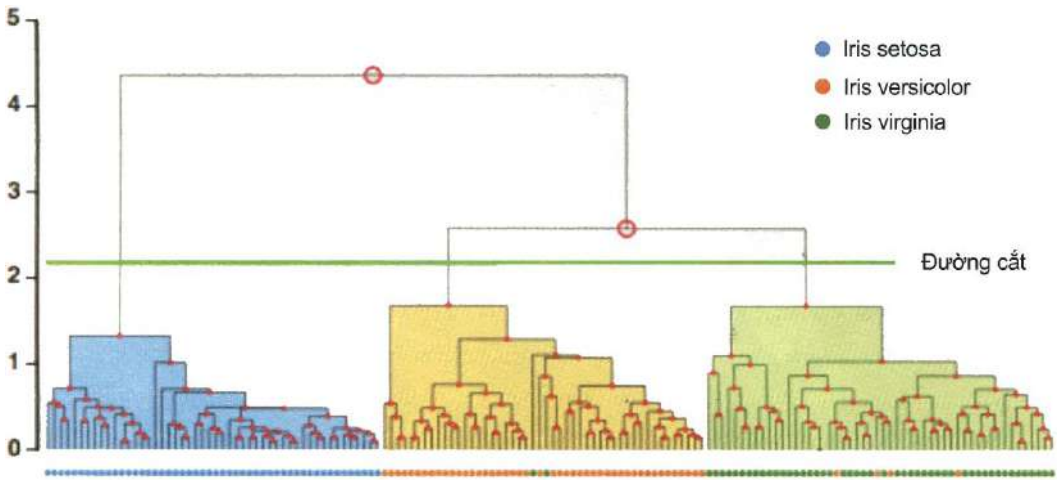


Hình 6-18: Phương pháp khuỷu tay xác định số lượng cụm K

6.4. Phân cụm thứ bậc và phân cụm sinh học

Ngoài phân cụm K-means, **phân cụm thứ bậc** (hierarchical clustering) cũng là một thuật toán phân cụm cổ điển. Phân cụm thứ bậc đầu tiên coi từng mẫu là một loại riêng biệt, và sau đó liên tục kết hợp hai loại giống nhau nhất.

Phân cụm theo cấp bậc kết thúc khi khoảng cách giữa tất cả các danh mục vượt quá khoảng cách được xác định trước.

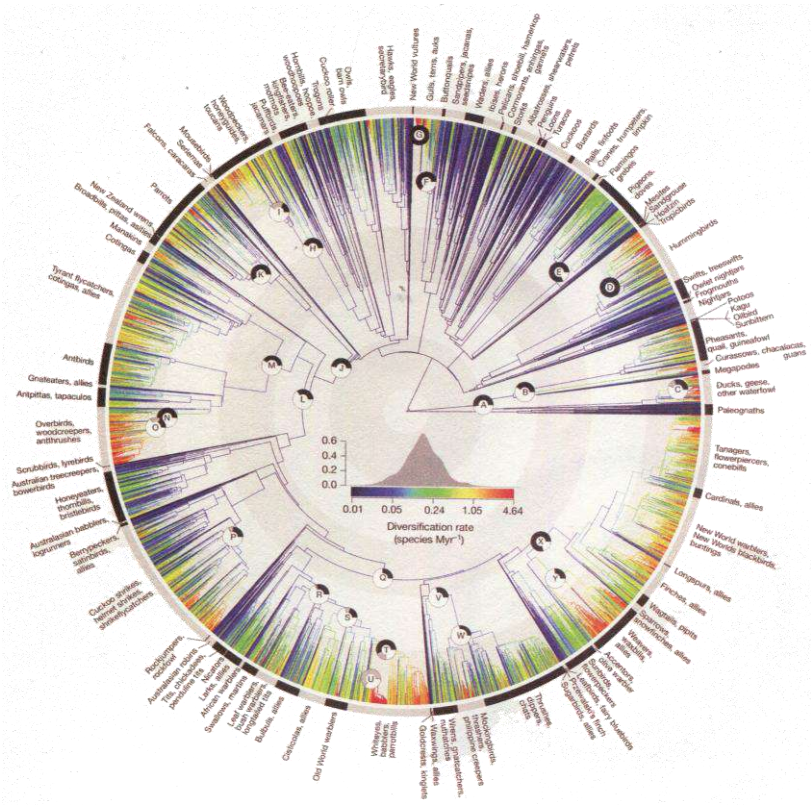


Hình 6-19: Phân loại hoa bằng thuật toán phân cụm thứ bậc

Hình 6-19 cho thấy các kết quả phân cụm hoa bằng cách phân cụm theo thứ bậc. Như bạn có thể thấy, phân cụm thứ bậc tạo ra một sơ đồ cây được xây dựng trên tất cả các mẫu. Kết quả phân cụm càng thấp, phân loại càng mịn thì số lượng các danh mục phân loại càng nhiều. Kết quả phân cụm bên dưới đường màu xanh lá cây cho biết khoảng cách cắt trùng với ba loài hoa.

Trong sinh học, phân cụm thứ bậc có thể giúp chúng ta phân biệt các giống khác nhau của hoa và cũng có thể được sử dụng để phân tích gen, phác họa sự phân loại và thậm chí sự tiến hóa của thực vật và động vật. Sử dụng trình tự DNA sinh học như một đặc trưng, và liên tục kết hợp các loài có tính tương đồng về di truyền cao, chúng ta có thể có được "cây phân loại" của sinh vật.

Thật thú vị, “cây phân loại” thu được bằng cách phân cụm theo thứ bậc được kết hợp với sự tiến hóa của sinh vật thể hiện trong Hình 6-20. Với sự phát triển của kỹ thuật sắp xếp gen và sự phát triển của học tập không giám sát, trong tương lai, trí thông minh nhân tạo cũng sẽ giúp chúng ta khám phá ra nhiều kết nối chưa biết giữa các loài sinh vật.



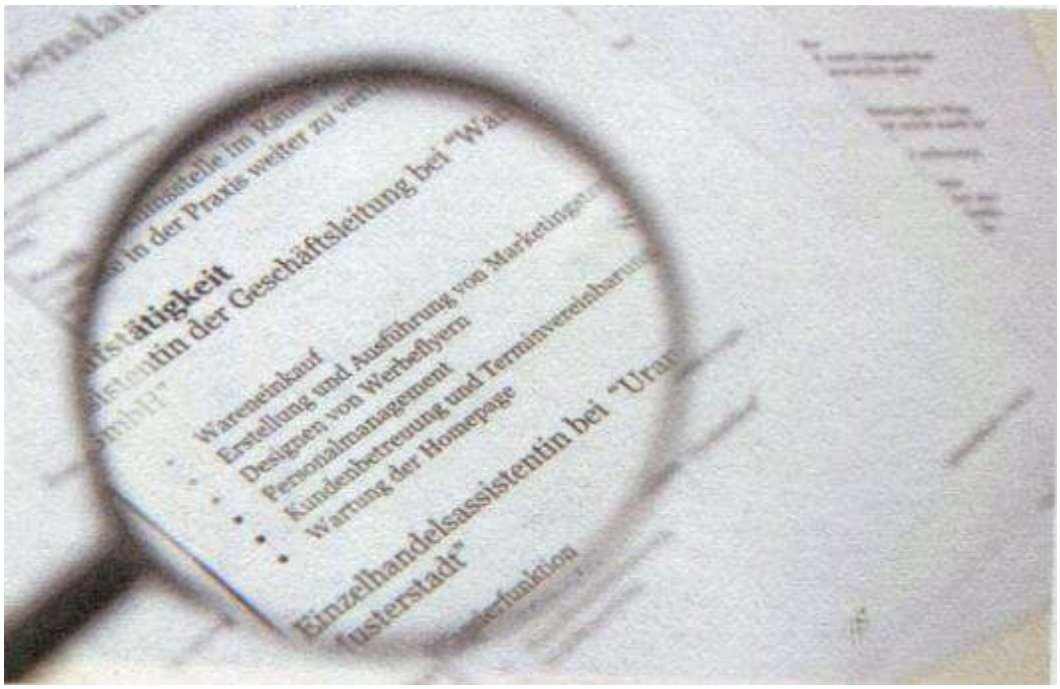
Hình 6-20: Cây tiến hóa sinh học

6.5. Tóm tắt chương

Trong chương này, chúng ta đã tìm hiểu về phương pháp học không giám sát đầu tiên. Không giống với học có giám sát, học không giám sát yêu cầu phát hiện các quy tắc trong dữ liệu mà không có thông tin về nhãn. Như một thuật toán học không giám sát cơ bản, thuật toán phân cụm K-means được tìm hiểu trong chương này có thể phân cụm dữ liệu mà không có chú thích. Vì các thuật toán phân cụm như K-means rất đơn giản và hiệu quả, chúng có ứng dụng quan trọng trong các lĩnh vực tài chính, y tế, khai thác dữ liệu lớn. Thêm vào đó, trong hai chương sau, ta sẽ được tìm hiểu các mô hình chủ đề và tạo ra học không giám sát như mạng confrontation.

Mạng internet ngày nay ngập tràn dữ liệu ảnh, văn bản, audio, video và nhiều hơn nữa. Hầu hết dữ liệu đều thiếu chú thích và việc học không giám sát đã nhận được nhiều sự chú ý hơn. Sự phát triển hưng thịnh của học không giám sát sẽ mang đến một bước nhảy vọt chất lượng cho tương lai của trí tuệ nhân tạo.

Chương VII. Hiểu văn bản



Kể từ khi sự ra đời của nền văn minh nhân loại, từ ngữ đã là phương tiện cơ bản để mọi người truyền tải thông tin. Trong Internet ngày nay phát triển cao, thông tin văn bản cũng đang phát triển với tốc độ bùng nổ. Các phương tiện truyền thông tiếp tục đăng tin tức mới nhất trên Internet mọi lúc và mọi người nói về những thứ xung quanh họ trên điện thoại di động của họ mọi lúc mọi nơi. Mỗi khoảnh khắc, một lượng lớn văn bản được tạo ra từ nhiều kênh khác nhau và được thu thập trên Internet. Khi đối mặt với dữ liệu văn bản lớn, chúng ta có thể sử dụng công nghệ thông minh nhân tạo để tự động phân tích và hiểu nó, do đó tiết kiệm thời gian đọc và năng lượng giới hạn của con người?

Trong chương này, chúng ta sẽ tìm hiểu về các kỹ thuật **phân tích ngữ nghĩa tiềm ẩn** (latent semantic analysis). Với công nghệ này, máy tính có thể tự động khám phá các chủ đề tiềm năng từ dữ liệu văn bản lớn và sau đó hoàn thành việc tổng quát và tinh chỉnh nội dung văn bản. Trước khi chúng ta tìm hiểu các công nghệ liên quan, trước tiên chúng ta hãy khám phá các đặc điểm của nhiệm vụ "khám phá các chủ đề tiềm năng từ văn bản".

7.1. Đặc điểm của nhiệm vụ

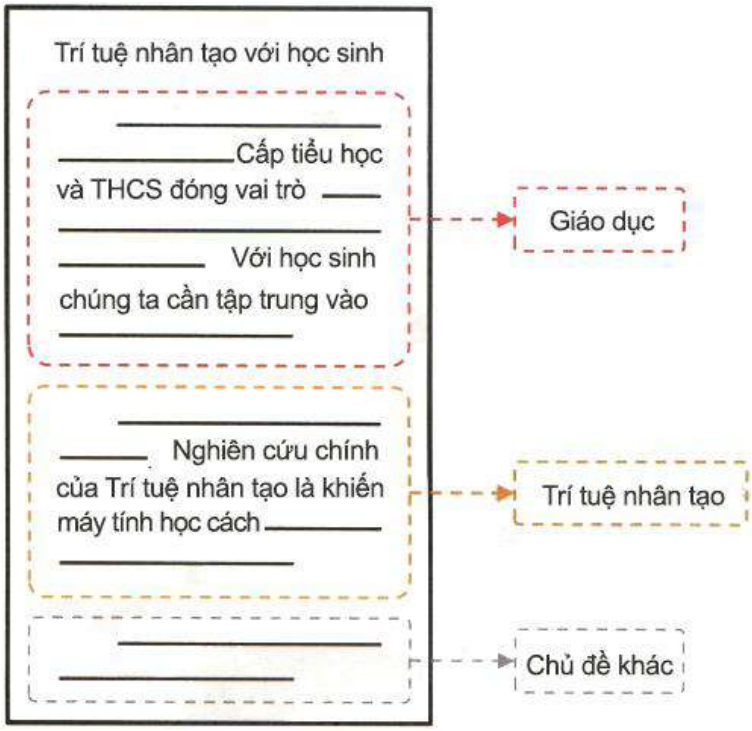
Dữ liệu văn bản thường không chứa thông tin chú thích bổ sung. Ví dụ, ta đăng một trạng thái lên mạng xã hội: “Tôi đã học được các khóa thông minh nhân tạo ở trường.” thì câu này tập trung vào “học tập” hoặc “trí thông minh nhân tạo” nhưng chúng ta sẽ không đánh dấu cụ thể các chủ đề này khi đăng lên. Nếu chúng tôi muốn phân tích tất cả các thông điệp trên mạng xã hội, thì thông tin chúng tôi có thể nhận được thường chỉ là nội thông điệp, mà không có bất kỳ đánh dấu bổ sung nào.

Chúng tôi có thể lấy thông tin về chủ đề của văn bản bằng chú thích thủ công không? Điều này thường bất khả thi. Kích thước của dữ liệu văn bản thường lớn hơn nhiều so với thông tin đa phương tiện như video và hình ảnh, trong khi chi phí ghi nhãn thủ công quá cao. Trong trường hợp này, phân tích dữ liệu là việc sử dụng các thuật toán học tập không giám sát.

Vì nó là nhiệm vụ của việc học không giám sát, chúng ta có thể sử dụng thuật toán K-means được giới thiệu trong chương trước để phân cụm dữ liệu dựa trên văn bản để trích xuất các chủ đề tiềm năng không? Điều này nghe có vẻ khả thi, nhưng chúng ta đã bỏ qua một đặc trưng của dữ liệu văn bản. Trong thuật toán K-means, chúng ta phân loại một mẫu thành một danh mục cụ thể và một đoạn văn bản thường có thể được trải rộng trên nhiều chủ đề. Ví dụ, một tin tức về “thúc đẩy giáo dục trí thông minh nhân tạo ở các trường tiểu học và trung học” sẽ tập trung vào ít nhất hai chủ đề “trí thông minh nhân tạo” và “giáo dục tiểu học và trung học.” Không việc phân loại nó thành một chủ đề duy nhất là không phù hợp.

Công nghệ phân tích ngữ nghĩa tiềm ẩn được thiết kế cho các đặc điểm của dữ liệu văn bản "đa chủ đề". Kỹ thuật này có thể phân tích nhiều chủ đề tiềm năng từ văn bản theo cách không giám sát, hoàn thành các nhiệm vụ mà thuật toán phân cụm không làm được.

Để thảo luận, bây giờ chúng ta sẽ được giới thiệu một số danh từ thích hợp có liên quan. Chúng ta thường gọi dữ liệu văn bản khổng lồ được đề cập ở trên là **kho văn bản** (corpus), văn bản độc lập trong kho được gọi là **tài liệu** (document) và ý tưởng chính hoặc nội dung chính của tài liệu được gọi là chủ đề (topic). Ví dụ, tất cả các bài báo trong năm 2017 có thể tạo thành một kho văn bản và mỗi bài viết trên báo tạo thành một tài liệu có thể xoay quanh “chính trị”, “kinh tế”, “giáo dục”, “công nghệ” và “sinh kế của người dân” và các chủ đề khác.



Hình 7-1: Một tài liệu gồm nhiều chủ đề

7.2. Đặc trưng của văn bản

Mô hình túi từ

Mô hình **túi từ** (bag-of-words) là một mô hình toán học đơn giản để mô tả văn bản và là một phương pháp phổ biến của trích xuất đặc trưng văn bản. Các mô hình túi xử lý một tài liệu như là một "túi với một vài từ", chỉ xem

xét số lần một từ xuất hiện trong tài liệu, bỏ qua thứ tự của các từ và cấu trúc của câu.

Ví dụ: đối với văn bản sau:

"Minh thích chơi bóng rổ và thích chơi bóng bàn."

Chúng ta có thể biểu diễn nó như một tập hợp các bộ dữ liệu (tuple) theo dạng (từ: số lần xuất hiện).

Tập hợp:

$$\{(Minh: 1)\ (Thích: 2)\ (Chơi: 2)\ (bóng\ rổ: 1)\ (và: 1)\ (bóng\ bàn: 1)\}$$

Bộ sưu tập này là "túi từ" tương ứng với văn bản trên.

Mô hình túi từ giản hóa tài liệu đáng kể mà vẫn giữ lại thông tin chủ đề của tài liệu ở một mức độ nào đó. Theo những từ "Minh", "bóng rổ" và "bóng bàn" trong túi, chúng ta vẫn có thể biết rằng tài liệu này liên quan đến hai chủ đề có thể có là Minh và Thể thao. Bỏ qua cấu trúc của các từ khó mô hình hóa và giữ lại số từ thể hiện chủ thể là ý tưởng cơ bản của mô hình túi.

Với túi từ, chúng ta có thể xây dựng một từ điển (từ vựng) có chứa một số từ, và sử dụng từ điển này để chuyển đổi túi từ thành một vector đặc trưng. Ví dụ, chúng ta có thể xây dựng một từ điển với sáu từ:

Số thứ tự	1	2	3	4	5	6
Từ	Minh	thích	chơi	bóng rổ	và	bóng bàn

Ta sắp xếp số lần xuất hiện của mỗi từ trong tài liệu theo số lượng của từ đó và thu được vectơ đếm từ (term counting vector) $n = (1,2,2,1,1,1)$ của tài liệu này. Chúng ta cũng có thể chuẩn hóa vectơ đếm từ (ví dụ, chia tỷ lệ chiều dài của vectơ sao cho tổng của tất cả các phần tử là 1) và lấy vector tần số từ (term frequency vector) $f = (⅛, ¼, ¼, ⅛, ⅛, ⅛)$.

Thông thường, ta không yêu cầu từ điển chứa tất cả các từ đã xuất hiện trong văn bản. Nếu một từ trong tài liệu không xuất hiện trong từ điển, chúng ta sẽ bỏ qua nó. Ví dụ: nếu bạn sử dụng từ điển sau có chứa bốn từ, vectơ đếm từ và vectơ tần số từ của tài liệu này trở thành $n = (1,2,1,1)$ và $f = (⅕, ⅖, ⅕, ⅕)$ tương ứng.

Số thứ tự	1	2	3	4
Từ	Minh	thích	bóng rổ	bóng bàn

Trong thực tế, chúng ta sử dụng một từ điển chung để thực hiện thống kê tần số từ trên tất cả các tài liệu trong kho văn bản. Hãy lấy một kho văn bản với ba tài liệu làm ví dụ:

Tài liệu 1: Minh thích chơi bóng rổ và thích chơi bóng bàn.

Tài liệu 2: Minh đến công viên để thả diều.

Tài liệu 3: Trường của Minh cung cấp các khóa học trí tuệ nhân tạo.

Đầu tiên, ta trích xuất tất cả các từ đã xuất hiện từ kho văn bản và tạo từ điển.

Số thứ tự	1	2	3	4
Từ	Minh	thích	chơi	bóng rổ
Số thứ tự	5	6	7	8
Từ	và	bóng bàn	đến	công viên
Số thứ tự	9	10	11	12
Từ	để	thả diều	Trường	của
Số thứ tự	13	14	15	16
Từ	cung cấp	các	trí tuệ nhân tạo	khóa học

Tiếp theo, chúng ta đếm số lần xuất hiện của mỗi từ trong mỗi tài liệu, như trong *Hình 7-2*.

	Minh	thích	chơi	bóng rổ	và	bóng bàn	đến	công viên
Tài liệu 1	1	2	2	1	1	1	0	0
Tài liệu 2	1	0	0	0	0	0	1	1
Tài liệu 3	1	0	0	0	0	0	0	0

	để	thả diều	Trường	của	cung cấp	các	trí tuệ nhân tạo	khóa học
Tài liệu 1	0	0	0	0	0	0	0	0
Tài liệu 2	1	1	0	0	0	0	0	0
Tài liệu 3	0	0	1	1	1	1	1	1

Hình 7-2: Các từ xuất hiện trong tài liệu thống kê

Kết quả thống kê là vectơ đếm từ của ba tài liệu.:

$$n_1 = (1, 2, 2, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)$$

$$n_2 = (1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0)$$

$$n_3 = (1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1)$$

Các mô hình túi từ rất đơn giản, nhưng nó cần phải được kết hợp với một số kỹ thuật xử lý văn bản để đạt được kết quả tốt hơn trong ứng dụng. *Hình 7-3* cho thấy luồng cơ bản của việc xây dựng một đặc trưng văn bản bằng cách sử dụng mô hình túi từ. Chúng ta sẽ tìm hiểu ngắn gọn các công nghệ liên quan trong phần tiếp theo.

Hình 7-3: Luồng cơ bản của ứng dụng mô hình túi từ

Phân đoạn từ

Đầu tiên chúng ta cần tách các từ trong câu để xây dựng một túi từ dựa trên các từ. Quá trình này rất dễ dàng đối với tiếng Anh. Chúng ta chỉ cần phân tách tất cả các từ dựa trên dấu cách và dấu chấm câu. Nhưng trong văn bản tiếng Việt, tất cả các từ được kết nối với nhau. Máy tính không biết liệu một từ có nên được kết nối với các từ trước và sau đó hay không, hoặc nên tự tạo thành một từ. Vì vậy, trước khi chúng tôi xây dựng từ túi cho văn bản, chúng ta cần tách các từ trong văn bản bằng các phương tiện bổ sung. Kỹ thuật này được gọi là phân đoạn từ (word segmentation) tiếng Việt. Phương pháp phân đoạn từ tiếng Việt chủ yếu dựa trên phương pháp so khớp và thống kê, sẽ không được giới thiệu ở đây.

Stop word và từ tần số thấp

Trong ví dụ trên, chúng ta thấy từ điển chứa các từ như "của", "và", "các"... Những từ này là những từ cơ bản tạo thành một câu tiếng Việt. Những từ này xuất hiện với số lượng lớn bất chấp chủ đề của tài liệu, nhưng chúng không giúp phân biệt chủ đề của các tài liệu khác nhau. Một từ tần số cao như thế này không mang bất kỳ thông tin chủ đề nào được gọi là stop word. Khi xây dựng một từ điển, chúng ta thường loại bỏ các stop word này.

Khi xây dựng một từ điển, chúng ta thường loại bỏ các từ tần số thấp rất hiếm khi xuất hiện. Những từ như vậy thường là một số danh từ ít phổ biến hơn. Chúng có thể xuất hiện trong các bài báo cụ thể (chẳng hạn như tên của một người trả lời ngẫu nhiên trong một cuộc phỏng vấn), nhưng chúng không đại diện cho một loại chủ đề nhất định. Nếu chúng ta dựa quá nhiều vào những từ như vậy để phân loại chủ đề của bài viết, thì có thể xảy ra hiện tượng overfitting. Mặt khác, nếu chúng ta bao gồm tất cả các từ tần số thấp trong từ điển, nó sẽ làm tăng đáng kể kích thước của từ điển và đặc tính của vectơ đặc trưng, điều này sẽ gây ra những khó khăn về tính toán. Do đó, thường sau khi chúng tôi thu thập tất cả các từ trong kho văn bản, chúng tôi sẽ giữ lại hàng nghìn hoặc hàng chục nghìn từ thông dụng và loại bỏ các từ tần số thấp.

Tần số từ và tần suất tài liệu nghịch đảo

Tần số từ và tần số tài liệu nghịch đảo là hai chỉ số phản ánh tầm quan trọng của một từ đối với tài liệu. Tần số mà từ đó xuất hiện trong tài liệu là tần số từ (term frequency), bằng với thương số của số lần xuất hiện của từ trong văn bản và tổng số từ trong văn bản. Chúng ta đếm số từ có số i xuất hiện trong tài liệu j là n_{ij}, tổng số từ trong tài liệu j được tính $n_j = \sum_{i=1}^{V} n_{ij}$, trong đó V là kích thước của từ điển. Tần số từ của từ i trong tài liệu j được tính $tf_{ij} = n_{ij}/n_j$. Ví dụ, có tổng cộng bốn từ trong tài liệu đầu tiên, trong đó từ đầu tiên "Minh" xuất hiện một lần trong tài liệu này, vậy tần số từ của từ đầu tiên trong tài liệu đầu tiên là 1/4.

Theo suy nghĩ thông thường một từ thường xuyên xuất hiện trong một tài liệu, nó càng quan trọng đối với tài liệu. Ví dụ: nếu từ "Minh" xuất hiện trong một số lượng lớn văn bản, thì "Minh" có thể là nội dung chính của tài liệu này. Nhưng giả định này không hợp lý trong một số trường hợp. Ví dụ, stop word sẽ xuất hiện với số lượng lớn trong mỗi tài liệu, nhưng tầm quan trọng của những từ này đối với tài liệu là rất thấp. Ví dụ: trong ba tài liệu đầu tiên, trong tài liệu đầu tiên, từ đầu tiên "Minh" và từ thứ tư "Bóng rổ" chỉ xuất hiện một lần và tần số từ giống nhau. Nhưng nếu cả ba tài liệu đều từ blog cá

nhân của Minh, thì từ "Minh" tương tự như stop word. Cho dù tài liệu xoay quanh thể thao hay giáo dục ở trường, từ “Minh” là không thể tránh khỏi và ít quan trọng hơn “bóng rổ” hay “khóa học” - là những từ có thể phân biệt giữa các chủ đề khác nhau. Tại thời điểm này, chúng ta cần phải sử dụng tần số tài liệu nghịch đảo (inverse document frequency) để điều chỉnh tầm quan trọng của mỗi từ trong mỗi tài liệu.

Chúng ta gọi tần số tài liệu (document frequency) của một từ là thương của tổng số các văn bản có chứa từ đó trong kho văn bản và tổng số tất cả tài liệu trong kho văn bản. Nếu có tổng số D tài liệu trong kho văn bản, và từ thứ i xuất hiện trong tổng số D_i tài liệu, tần suất tài liệu của từ thứ i là $df_i = D_i/D$. Tần số tài liệu nghịch đảo của thuật ngữ này là logarit âm của tần số tài liệu, tức là $idf_i = \log(D/D_i)$. Để tránh trường hợp mẫu số là 0, đôi khi chúng ta xác định tần số tài liệu nghịch đảo $idf_i = \log(\frac{D}{1+D_i})$. Tần số tài liệu nghịch đảo cũng mô tả tầm quan trọng của các từ trong văn bản, giá trị càng cao thì tầm quan trọng càng lớn.

Chúng ta vẫn sử dụng ba tài liệu trước đó làm ví dụ. Sau khi xóa stop word "của", "và", "các", mỗi phần văn bản có thể được biểu diễn dưới dạng vectơ đếm từ 13 chiều:

$$n_1 = (1, 2, 2, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0)$$

$$n_2 = (1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0)$$

$$n_3 = (1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1)$$

Chúng ta có thể thấy từ "Minh" xuất hiện trong cả ba tài liệu và tần suất tài liệu ngược của từ "Minh" là $log\ (\frac{3}{3}) = 0$. Từ "bóng rổ" chỉ xuất hiện trong 1 tài liệu, vì vậy tần số tài liệu nghịch đảo của nó là $log(\frac{3}{1}) \approx 0{,}47$. Giá trị tính toán của tần suất tài liệu nghịch đảo phù hợp với ý tưởng trực quan của chúng ta rằng từ "Minh" ít quan trọng hơn từ "bóng rổ".

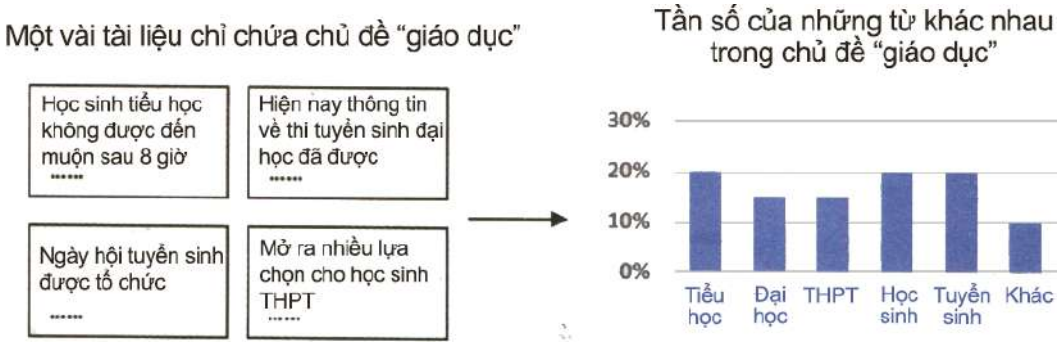
Bằng cách nhân tần số của một từ trong một bài báo với tần số tài liệu nghịch đảo của từ ($tf \times idf$), chúng ta có thể nhận được tần số của từ trong bài viết này — tần số tài liệu nghịch đảo ($tf - idf$). Tần số tài liệu nghịch đảo là một sửa đổi của tần số từ làm nổi bật thông tin quan trọng hơn trong văn bản. Chúng ta có thể thay thế giá trị tần số trong vector tần số từ của tài liệu với giá trị tần số tài liệu đảo ngược để thu được vectơ tần số tài liệu đảo ngược từ của tài liệu này như là một đặc trưng của tài liệu.

7.3. Khám phá các chủ đề tiềm năng trong văn bản

Mô hình chủ đề

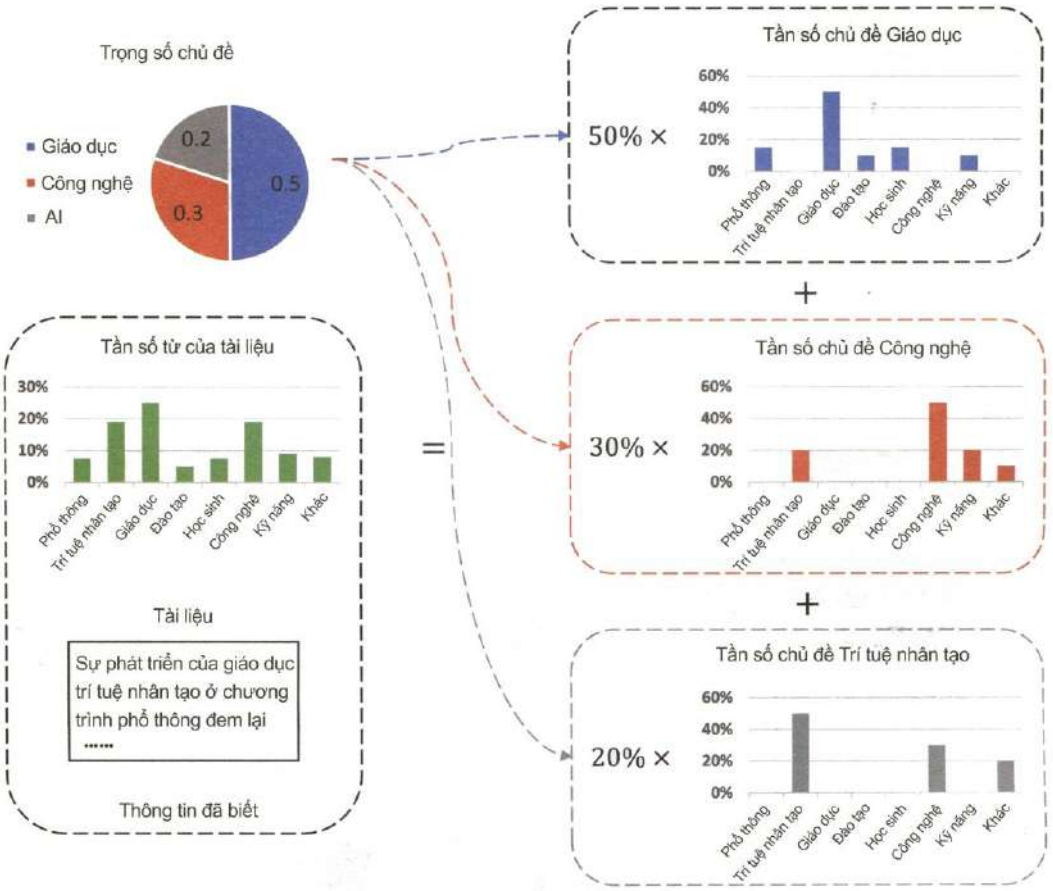
Mô hình chủ đề (topic model) là một loại mô hình toán học mô tả một kho văn bản và các chủ đề cơ bản của nó. Một trong những câu hỏi đầu tiên chúng ta xem xét trong mô hình chủ đề là làm thế nào để mô tả một chủ đề trong toán học. Khi giới thiệu mô hình túi từ, chúng ta đã biết rằng các từ xuất hiện trong văn bản có thể phản ánh chủ đề của văn bản. Vì vậy, nếu chúng ta có thể thu thập một số tài liệu chỉ chứa một chủ đề duy nhất (như chủ đề giáo dục trong *Hình 7-4*) và đếm tần suất xuất hiện của các từ, thì kết quả thống kê có thể được sử dụng làm đại diện cho chủ đề này.

Cụ thể, nếu kích thước của từ điển là V, với mỗi từ ta tính được có tổng n_i lần xuất hiện trong tất cả các tài liệu, chia cho tổng số từ n trong tài liệu để thu được tần số từ tương ứng. Sau đó, ta kết hợp tất cả các tần số từ để thu được vectơ tần số từ $t = (x_1, x_2, \ldots, x_V)$. Vector tần số từ này là biểu diễn toán học của chủ đề giáo dục.



Hình 7-4: Tần số từ biểu diễn cho chủ đề

Phương pháp thống kê tần số từ cung cấp cho ta ý tưởng mô hình hóa chủ đề, nhưng phương pháp này có những thiếu sót trong thực tế. Một mặt, mỗi tài liệu thường chứa nhiều hơn một chủ đề, việc chỉ có một chủ đề duy nhất là rất hiếm. Mặt khác, không có thông tin chú thích về chủ đề của tài liệu trong kho văn bản. Ngay cả khi có một tài liệu với một chủ đề duy nhất, rất khó để chúng ta khám phá nó từ một kho dữ liệu khổng lồ. Do đó, trong thực tế, chúng ta phải sử dụng các kỹ thuật bổ sung để thu được vectơ tần số từ tương ứng với từng chủ đề.



Hình 7-5: Tần số từ của mỗi tài liệu là một hỗn hợp tần số từ của các chủ đề liên quan.

Hãy suy nghĩ về mối quan hệ giữa tài liệu và chủ đề. Mỗi tài liệu thường chứa một số lượng chủ đề, mỗi chủ đề tương ứng với một vectơ từ. Ví dụ trong *Hình 7-5* có ba chủ đề "giáo dục", "công nghệ" và "trí tuệ nhân tạo". Các vectơ từ của chủ đề này được vẽ ở phía bên phải của hình. Vậy mối quan hệ giữa vector tần số của tài liệu này và vector tần số từ tương ứng với các chủ đề này là gì? Thông thường trọng số của mỗi chủ đề có trong một tài liệu có giá trị khác nhau. Trong ví dụ của *Hình 7-5*, tỷ lệ chủ đề "giáo dục" lớn hơn hai chủ đề khác. Trong mô hình chủ đề, chúng ta coi vectơ tần số từ của tài liệu là trung bình của các vectơ tần số từ của tất cả các chủ đề mà nó chứa, và trọng số của từng chủ đề biểu diễn trọng số của chủ đề đó trong tài liệu.

Cụ thể, ta giả định rằng có tổng số T chủ đề tiềm năng (số lượng chủ đề thường được chỉ định theo cách thủ công). Điều này tương tự như lựa chọn số cụm K trong thuật toán K-means. Mỗi chủ đề tương ứng với một vectơ tần số từ $t_j = (x_{j1}, x_{j2}, \ldots x_{jV}), 1 \leq j \leq T$, trong một tài liệu cụ thể, trọng

số của mỗi chủ đề là $w_1, w_2, \ldots, w_T$. Ta đã biết vectơ tần số từ của tài liệu là $d = (y_1, y_2, \ldots, y_V)$ và chúng ta có thể diễn tả mối quan hệ giữa tần số từ tài liệu, trọng số chủ đề và tần số chủ đề là

$$d = w_1 t_1 + w_2 t_2 + \ldots + w_V t_V \qquad (7\text{-}1)$$

Trong đó $w_i t_i$ là phép nhân của trọng số $w_i$và vector t_i.

Với phép nhân của ma trận, chúng ta cũng có thể diễn đạt biểu thức này dưới dạng ngắn gọn hơn. Đầu tiên, ta sắp xếp các vectơ tần số từ của tất cả các chủ đề T thành một ma trận.

$$[T] = \begin{bmatrix} - & t_1 & - \\ - & t_2 & - \\ \ldots & \ldots & \ldots \\ - & t_T & - \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1V} \\ x_{21} & x_{22} & \ldots & x_{2V} \\ \vdots & \vdots & \ddots & \vdots \\ x_{T1} & x_{T2} & \cdots & x_{TV} \end{bmatrix}$$

Sau đó, ta sắp xếp tất cả các chủ đề thành một vectơ $w = (w_1, w_2, \ldots, w_T)$. Với phép nhân ma trận, chúng ta có thể đơn giản hóa công thức (7-1) thành

$$d = wT \qquad (7\text{-}2)$$

Nếu có tổng số D tài liệu trong kho văn bản, vectơ tần số từ của mỗi tài liệu là $d_k = (y_{k1}, y_{k2}, \ldots, y_{kV}), 1 \le k \le D$ và vectơ trọng số chủ đề của mỗi tài liệu là $w_k = (w_{k1}, w_{k2}, \ldots, w_{kV}), 1 \le k \le D$. Chúng ta có thể sắp xếp vector từ và vector trọng số của tất cả các tài liệu vào một ma trận.

$$D = \begin{bmatrix} - & d_1 & - \\ - & d_2 & - \\ \ldots & \ldots & \ldots \\ - & d_D & - \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1V} \\ y_{21} & y_{22} & \ldots & y_{2V} \\ \vdots & \vdots & \ddots & \vdots \\ y_{D1} & y_{D2} & \cdots & y_{DV} \end{bmatrix}$$

$$W = \begin{bmatrix} - & w_1 & - \\ - & w_2 & - \\ \ldots & \ldots & \ldots \\ - & w_D & - \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1T} \\ w_{21} & w_{22} & \ldots & w_{2T} \\ \vdots & \vdots & \ddots & \vdots \\ w_{D1} & w_{D2} & \cdots & w_{DT} \end{bmatrix}$$

Sử dụng phép nhân ma trận, mối quan hệ của tần số tài liệu,chủ đề tiềm năng và tần số từ được biểu diễn theo công thức

$$D = WT \qquad (7\text{-}3)$$

Phương trình này thiết lập mối quan hệ giữa kho văn bản và các chủ đề tiềm năng và là trọng tâm của mô hình chủ đề.

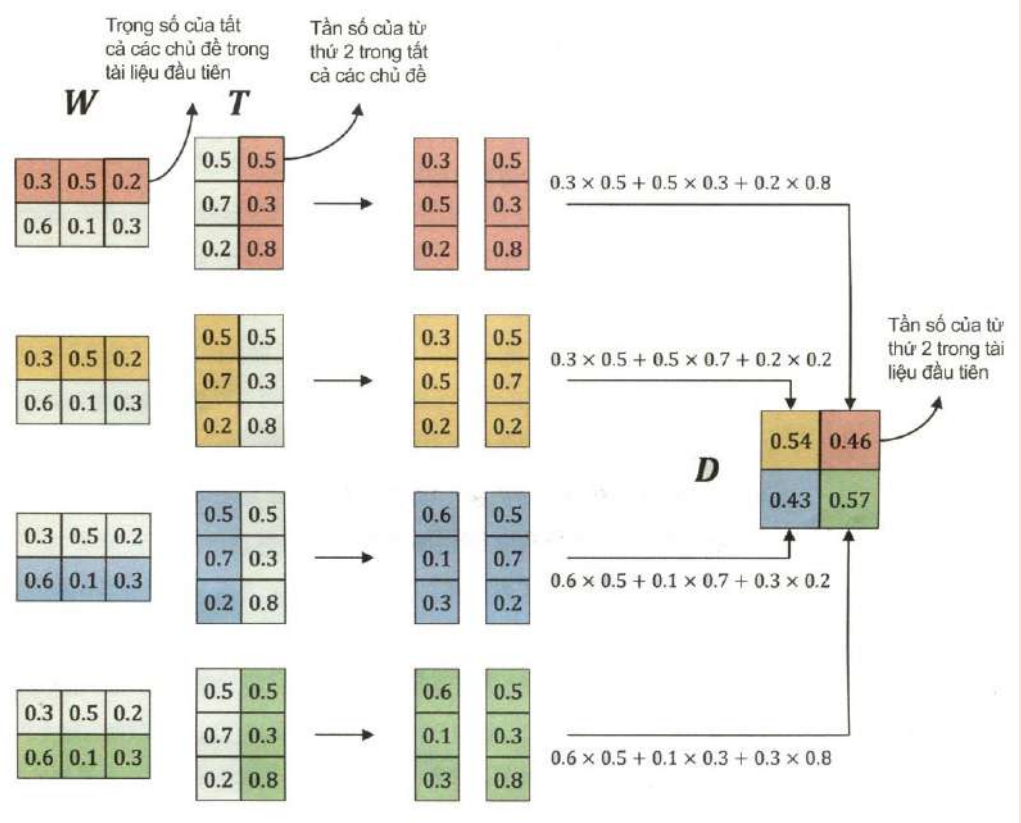
Kiến thức bổ sung: Nhân ma trận

Một ví dụ để hiểu phép nhân của mảng. Giả sử có 2 tài liệu trong kho văn bản của chúng ta, 2 từ trong từ điển và 3 chủ đề tiềm năng.

Trong trường hợp này, ma trận tần số chủ đề T là một ma trận 2 hàng 3 cột, mỗi hàng đại diện cho một vector tần số từ chủ đề, và mỗi cột đại diện cho một ứng dụng khác nhau trong tất cả các chủ đề của một từ hài hòa. Trọng số của đối tượng W là 2 hàng và 3 cột, mỗi hàng đại diện cho trọng số của chủ đề trong tài liệu.

Kết quả của phép nhân ma trận vẫn là ma trận có số hàng bằng số hàng của ma trận đầu tiên, và số lượng cột bằng với số cột của ma trận thứ hai. Trong ví dụ của chúng ta, kết quả nhân hai ma trận là ma trận của 2 hàng và 2 cột.

Như trong *Hình 7-6*, mỗi phần tử trong ma trận được tính như sau. Ta chọn hàng thứ i từ ma trận W, chọn cột thứ j từ ma trận T, và tính tích vô hướng của hai vectơ ba chiều làm giá trị của hàng thứ i và cột thứ j trong ma trận kết quả. Theo công thức trên, chúng ta biết rằng giá trị này bằng với tần số từ của từ thứ j trong tài liệu i.



Hình 7-6: Sơ đồ nhân ma trận

Phân tích ngữ nghĩa tiềm ẩn

Thông qua mô hình chủ đề, chúng ta đã thiết lập mối quan hệ giữa kho văn bản và các chủ đề cơ bản. Phương trình (7-3) xây dựng một hệ phương trình, trong đó ma trận tần số từ tài liệu ở bên trái của dấu bằng có thể thu được thông qua kho dữ liệu thống kê, là số lượng đã biết, và ma trận trọng số từ và chủ đề ma trận tần số từ ở phía bên phải đều không rõ. Bằng cách giải quyết hệ phương trình này, chúng ta có thể nhận được vector tần số từ tương ứng với chủ đề, và các chủ đề chứa trong mỗi tài liệu, để hoàn thành việc khám phá các chủ đề tiềm năng trong kho văn bản. Đây là kỹ thuật phân tích ngữ nghĩa tiềm năng.

Nhưng giải quyết hệ phương trình này không dễ. Chúng ta nhận thấy rằng có những ẩn số DxT trong ma trận trọng số, có những ẩn số TxV trong ma trận chủ đề và số phương trình là DxV, phương trình này có khả năng không thể giải được.

Lý do là vì mô hình chủ đề của chúng ta thiết lập mối quan hệ giữa phần văn bản và chủ đề đơn giản bằng cách lấy trung bình các trọng số, vì vậy không thể tránh khỏi các lỗi. Tuy nhiên, chúng ta có thể tính toán một tập hợp ma trận tần số từ chủ đề T và ma trận trọng số từ W bằng một phương pháp như hệ số ma trận không âm (non-negative matrix factorization), sao cho vế trái và phải của công thức (7-3) gần nhau nhất có thể. Vì việc phân tách ma trận không âm có liên quan đến kiến thức toán học nâng cao hơn, chúng ta sẽ không đi vào chi tiết ở đây.

Ma trận tần số chủ đề T đại diện cho tất cả các chủ đề tiềm năng trong kho văn bản và ma trận trọng số chủ đề chứa trọng số của mỗi chủ đề trong mỗi tài liệu. Sau khi nhận được hai giải pháp này, chúng ta đã hoàn thành việc khám phá các chủ đề tiềm năng trong kho văn bản và khái quát hóa và hiểu biết về từng tài liệu trong kho văn bản.

7.4. Đề xuất tìm kiếm văn bản của chủ đề

Trong cuộc sống hàng ngày và công việc, chúng ta thường cần sử dụng các công cụ tìm kiếm để tìm nội dung thú vị trên Internet. Các công cụ tìm kiếm truyền thống thường dựa trên các kỹ thuật đối sánh từ khóa. Ví dụ: nếu chúng ta muốn tìm kiếm các bài báo khoa học phổ biến về "độ ẩm của thực phẩm", công cụ tìm kiếm sẽ xóa stopword "của" và sử dụng từ khóa

"thực phẩm" và "độ ẩm" để tìm các bài viết phù hợp. Nhưng công nghệ tìm kiếm dựa trên kết hợp từ khóa có hai nhược điểm rõ ràng:

1. Vấn đề đồng nghĩa. Ví dụ sử dụng "dễ thương" làm từ khóa, sau đó những từ như "đáng yêu", "xinh xắn", v.v., từ đồng nghĩa sẽ bị bỏ qua, nhưng trên thực tế chúng có cùng ý nghĩa trong ngữ cảnh nhất định.

2. Từ đồng âm. Ví dụ, từ "đường" có thể có nghĩa là lộ trình cũng có thể là gia vị. Nếu ta sử dụng "đường" làm từ khóa và muốn tìm kiếm các bài viết về giao thông, một số bài viết chủ đề ẩm thực có thể xuất hiện trong kết quả tìm kiếm.

Sau khi tìm hiểu mô hình chủ đề và các kỹ thuật phân tích ngữ nghĩa tiềm năng, chúng ta biết rằng mỗi bài viết sẽ chứa một số chủ đề. Nếu chúng ta nhìn vào chủ đề của tài liệu trong quá trình tìm kiếm, chúng ta có thể khắc phục các giới hạn của từ khóa. Ví dụ: nếu chúng tôi chỉ định chủ đề "giao thông" khi tìm kiếm bằng từ khóa "đường", thì các bài viết có liên quan đến "ẩm thực" sẽ được lọc ra dễ dàng.

Cụ thể, trước tiên chúng tôi sẽ tìm kiếm các bài viết có chứa "đường" thông qua các kỹ thuật đối sánh từ khóa làm kết quả tìm kiếm ứng cử viên. Tiếp theo, chúng tôi sẽ sử dụng ma trận tần số chủ đề T thu được bằng cách phân tích văn bản lớn để thực hiện phân tích chủ đề trên mỗi tài liệu ứng viên và nhận được trọng lượng của từng chủ đề được bao gồm trong mỗi tài liệu. Sau đó, chúng tôi có thể lọc ra các bài viết liên quan đến chủ đề "giao thông" dưới dạng kết quả cuối cùng và trả lại cho người dùng.

Chúng ta cũng có thể sử dụng công nghệ phân tích ngữ nghĩa tiềm năng để đạt được các đề xuất cá nhân cho các bài viết. Nếu Minh thích xem tin tức, trang web tin tức có thể thu thập tin tức mà Minh đã xem và phân tích vectơ trọng lượng của chủ đề có trong tài liệu tin tức này. Các vectơ trọng số chủ thể này thể hiện các ưu tiên của Minh. Nếu Minh thích xem tin tức thể thao và công nghệ, tỷ lệ thể thao và công nghệ có thể cao.

Sau đó, để có tin tức mới nhất, chúng ta cũng có thể sử dụng cùng một công nghệ để phân tích tỷ lệ của mỗi chủ đề. Nếu vector trọng lượng tương tự như vector của Minh, thì bài viết này có thể là chủ đề mà Minh quan tâm và trang web tin tức có thể giới thiệu bài viết này cho Minh.

7.5. Tóm tắt chương

Trong chương này, chúng ta đã nghiên cứu mô hình từ và sử dụng các kỹ thuật phân tích ngữ nghĩa tiềm năng để hoàn thành nhiệm vụ phân tích văn bản và khai thác chủ đề. Mô hình túi từ chỉ xem xét số lần xuất hiện của các từ trong văn bản, bỏ qua mối quan hệ thứ tự giữa các từ, là một mô hình toán học đơn giản để tạo văn bản. Mô hình chủ đề đại diện cho chủ đề về tần số từ và giả định rằng vectơ tần số từ của tài liệu là trọng số trung bình của các vectơ tần số từ cho tất cả các chủ đề có trong tài liệu. Dựa trên các giả định của mô hình chủ đề, chúng ta có thể liệt kê một phương trình về kho văn bản và các chủ đề cơ bản của nó, và giải quyết nó bằng hệ số ma trận không âm.

Dữ liệu văn bản có các đặc điểm của nhiều loại không giám sát và mô hình chủ đề và công nghệ phân tích ngữ nghĩa tiềm ẩn là một phương pháp máy học không giám sát quan trọng được đề xuất cho hai đặc điểm này.

Chương VIII. Cây bút của Chúa: Bản vẽ sáng tạo

Trong Đệ nhị Thế chiến, một số ít kiệt tác của họa sĩ Johannes Vermeer người Hà Lan được Han van Meegeren bán cho trùm Quốc Xã Hermann Göring với giá bằng 200 tác phẩm nghệ thuật nổi tiếng khác. Đó là một trong những báu vật quốc gia Hà Lan. Sau chiến tranh, Van Meegeren bị buộc tội phản quốc vì đã cộng tác với Đức Quốc Xã và đã bán báu vật quốc gia cho chúng và bị tuyên án treo cổ. Thực tế, những bức tranh mà Van Meegeren bán đều được ông làm giả và không phải báu vật. Để chứng minh, trước sự hiện diện của các nhà báo và công chứng tòa án, Han van Meegeren đã vẽ bức tranh giả cuối cùng của mình, bức "Jesus và các nhà thông thái". Bức tranh của ông không chỉ lừa dối những vị tướng quyền lực nhất, mà còn qua mắt được các nhà phê bình nghệ thuật. Kể từ đó, hình ảnh của Van Meegeren trong con mắt của người Hà Lan đã thay đổi từ một kẻ phản bội thành một anh hùng dân tộc, người đã sử dụng kỹ thuật vẽ tuyệt vời để lừa được trùm Đức Quốc Xã Göring bằng tranh giả.

Minh đã rất phấn khởi sau khi đọc câu chuyện của Meegeren, vì vậy anh đã gọi cho người bạn của mình, người đang học vẽ ở Pháp:

"Tôi muốn học vẽ với bạn!"

"Nhưng chúng ta cách xa nhau quá, chúng ta không thể trực tiếp làm điều đó được"

"Vậy có cách nào không?"

"Tôi nghĩ có một cách để làm điều đó. Mỗi ngày, bạn chọn một chủ đề để vẽ một bức tranh và gửi nó cho tôi. Tôi đánh giá liệu nó có tốt hay không. Nếu tôi cảm thấy rằng bức tranh chưa giống, tôi sẽ đưa ra phản hồi. Sau đó, bạn có thể cải thiện kỹ thuật vẽ dựa trên những ý kiến này. Bằng cách này, miễn là chúng ta tuân theo quy trình 'sáng tạo – đánh giá – phản hồi – hoàn thiện', ngày này qua ngày khác, cấp độ sáng tạo của bạn sẽ ngày càng tốt hơn. Khi tôi thấy bức tranh đủ đẹp và không còn gì để phản hồi nữa, bạn sẽ trở thành giáo viên. ”

"Vì vậy, tôi chỉ phải tìm cách để đánh lừa đôi mắt của bạn - giống như Mechelen đã làm - nó thành công, phải không?"

“Phải, nhưng hãy cẩn thận, mức độ nhận diện của tôi đang được cải thiện! Đây thực sự là một trò chơi đối đầu thú vị. ”

Minh lặng lẽ gật đầu và suy nghĩ: "Tôi có thể sử dụng phương pháp giống như vậy để máy tính có thể học cách vẽ !"

Vậy generative adversarial network (GAN - tạm dịch mạng đối nghịch tạo sinh) là gì? Tại sao nó khiến máy móc có thể vẽ tranh ? Trong chương này, chúng ta sẽ tìm hiểu làm cách nào để tạo một bức tranh giả sử dụng GAN.

8.1. Không gian dữ liệu và phân phối dữ liệu

"Generative adversarial network" bao gồm ba từ "generative", "adversarial " và "network". "Generative" có nghĩa là nó là một mô hình tạo sinh (generative model), nó có thể ngẫu nhiên tạo ra dữ liệu quan sát. Ví dụ, nếu tập huấn luyện được đặt cho nó là một bức ảnh ca sĩ, thì một mô hình tạo sinh được huấn luyện có thể "tạo ra", tự sinh ra một bức ảnh ca sĩ mới.

Nó bao gồm hai phần: một mạng tạo sinh (generative network) và một mạng phân biệt (discriminative network). Mạng tạo sinh được sử dụng để tạo dữ liệu và mạng phân biệt được sử dụng để phân biệt liệu dữ liệu là đúng hay sai. Lấy quá trình vẽ tự động của máy tính làm ví dụ, vai trò của mạng tạo sinh là họa sĩ (tạo ra hình ảnh), và vai trò của mạng phân biệt là nhà phê bình nghệ thuật (phân biệt hình ảnh là máy vẽ giả hay họa sĩ vẽ). Ý tưởng cơ bản của việc tạo ra GAN là học bằng cách để mạng tạo sinh và mạn phân biệt "đối nghịch" với nhau. Vậy, tính đặc thù của mạng tạo sinh và mạng phân biệt là gì? Làm thế nào để họ "đối đầu" giữa hai người? Tại sao cuộc đối đầu lại làm cho chất lượng của sản xuất trở nên tốt hơn và tốt hơn? Trước khi trả lời các câu hỏi này, chúng ta hãy xem xét và giới thiệu hai khái niệm cơ bản về không gian dữ liệu và phân phối dữ liệu.

Không gian dữ liệu và phân bố dữ liệu

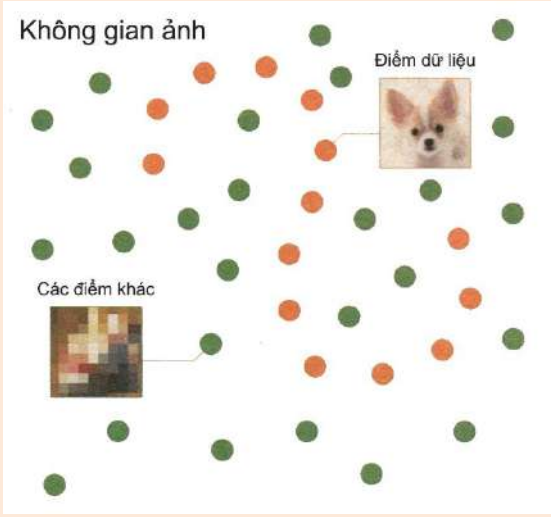
Chúng ta đã biết tầm quan trọng của dữ liệu đối với các hệ thống trí tuệ nhân tạo, việc tạo ra các mô hình cũng không ngoại lệ. Nếu mục tiêu của chúng ta là để cho máy tính tự động tạo ra hình ảnh trông giống như các ca sĩ từ đầu, chúng ta phải cung cấp rất nhiều hình ảnh ca sĩ để tham khảo. Trong con mắt của mô hình được tạo ra, những dữ liệu hình ảnh này tạo thành một tổng thể và cùng xác định các đặc điểm ngoại hình của các ca sĩ. Sự tạo sinh mô hình không phải là học cách tạo ra một bức ảnh của một ca sĩ cụ thể nào mà để nắm được những đặc tính tổng thể của những bức ảnh này để tạo ra một bức tranh về "ca sĩ". Vì vậy, làm thế nào để miêu tả dữ liệu? Điều này giới thiệu khái niệm về không gian dữ liệu và phân bố dữ liệu.

Không gian dữ liệu (data space), là không gian vị trí nơi dữ liệu được đặt. Giả sử độ phân giải của các ảnh ca sĩ được chụp là 128x128. Nhắc lại kiến thức của Chương 3, mỗi bức ảnh có thể được biểu diễn bằng một tensor bậc ba 128x128x3. Không gian dữ liệu trong trường hợp này là tập hợp tất cả các tensors có kích thước 128x128x3 hoặc tập hợp tất cả các hình ảnh có thể có ở độ phân giải này. Trong nhiệm vụ tạo hình ảnh, không gian dữ liệu là tập hợp các hình ảnh, vì vậy nó cũng được gọi là không gian hình ảnh. Trong

không gian hình ảnh, mỗi bức ảnh là một điểm trong không gian này. Như trong Hình 8-1, chúng ta có thể thấy rằng việc tìm kiếm một điểm trong không gian hình ảnh có thể là một hình ảnh không có ý nghĩa (các dấu chấm màu xanh lục trong hình). Các hình ảnh trong tập dữ liệu cũng được phân phối trong không gian này. Chúng ta gọi điểm cụ thể này là một điểm dữ liệu (dấu chấm màu vàng trong hình).

Kiến thức bổ sung: Không gian

"Không gian" ba chiều của cuộc sống của chúng ta là toàn bộ tất cả các địa điểm có thể. Trong toán học, khái niệm về **không gian** (space) được khái quát hóa để biểu diễn một **tập hợp** (set) các phần tử có tính tương đồng. Vì vậy, nó không còn đề cập đến không gian hai chiều hoặc không gian ba chiều, nhưng có thể thể hiện một khái niệm tổng quát hơn. Ví dụ, tất cả các dữ liệu cấu thành một không gian dữ liệu, tất cả các hình ảnh tạo thành một không gian hình ảnh, tất cả các vectơ tạo thành một không gian vectơ, v.v.

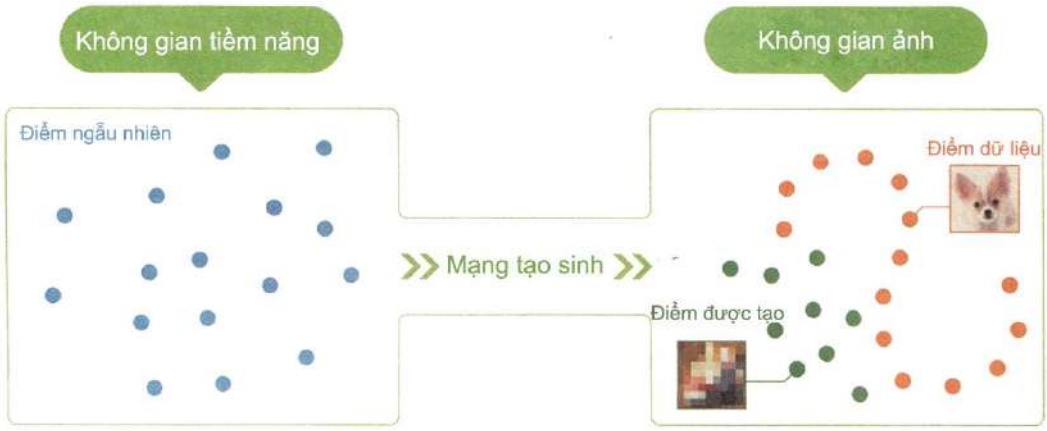


Hình 8-1: Không gian ảnh

Sự phân bố các điểm dữ liệu trong không gian dữ liệu không đều đặn: một số vùng có nhiều điểm dữ liệu, một số vùng có ít. Sự phân bố dữ liệu trong không gian được gọi là **phân bố dữ liệu** (data distribution). Trong toán học, phân bố là một khái niệm cơ bản trong nhánh của lý thuyết xác suất. Nó liên quan chặt chẽ đến sự ngẫu nhiên.

Các hiện tượng ngẫu nhiên có sự phân bố nhất định. Thông qua biểu đồ, chúng ta có thể nhìn thấy các đặc điểm của các sự phân bố khác nhau. Các hình ảnh của tất cả các ca sĩ cũng tạo thành một phân bố dữ liệu phức tạp trong không gian hình ảnh. Phân bố này không thuận tiện khi sử dụng một công cụ như biểu đồ mô tả trực tiếp. Vậy cần mô tả cách phân bố dữ liệu như thế nào? Mạng tạo sinh được mô tả dưới đây có một cách tiếp cận thông minh: Biến một phân bố đơn giản, dễ hiểu thành một phân bố dữ liệu phức tạp, khó hiểu. Điều này làm cho nó có thể gián tiếp nắm bắt sự phân bố dữ liệu phức tạp thông qua một bản phân bố đơn giản. Ví dụ: phân bố đơn giản này có thể được chọn từ phân phối bình thường trong thử nghiệm trước đó. Không gian trong đó mẫu được tạo ra bởi phân phối đơn giản này trong mạng được tạo ra được gọi là **không gian tiềm ẩn** (latent space).

8.2. Người sáng tạo diệu kỳ: Mạng tạo sinh



Hình 8-2: Sơ đồ nhiệm vụ công việc của mạng tạo sinh (một điểm trong không gian ảnh đại diện cho một hình ảnh)

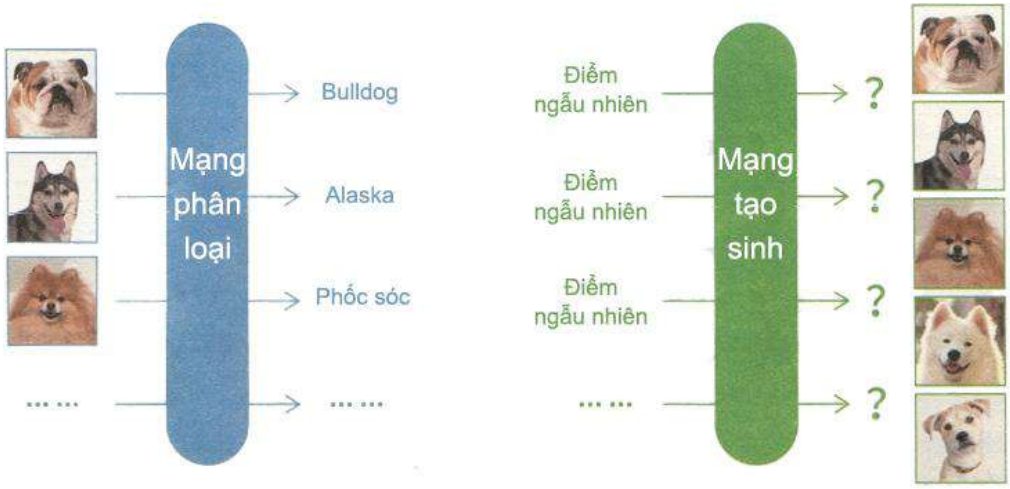
Trách nhiệm của mạng tạo sinh là để biến một điểm ngẫu nhiên thành một hình ảnh tương tự như một tập hợp dữ liệu. Những ngẫu nhiên này được rút ngẫu nhiên từ một không gian tiềm năng. Điều này giống như một họa sĩ biến đổi sự phát triển từ một ý tưởng trừu tượng ban đầu thành một bức tranh với những hình ảnh phức tạp. Như trong *Hình 8-2*, chúng ta thấy rằng mạng tạo sinh là một hàm thực hiện một phép chuyển đổi điểm-điểm, biến một điểm trong một không gian tiềm năng thành một điểm trong một không gian hình ảnh. Điểm mà tại đó được mạng tạo ra được gọi là điểm được tạo. Bằng cách

sử dụng mạng tạo sinh, phân bố trong latent space có thể được chuyển đổi thành một phân bố trong không gian hình ảnh. Chúng ta gọi đây là phân bố tạo sinh. Đôi khi mạng tạo sinh cũng được gọi là **generator**.

Việc phân phối các hình ảnh thực trong không gian hình ảnh rất phức tạp. Rất khó để các số đơn giản thay đổi chính xác các điểm ngẫu nhiên này thành vị trí của hình ảnh thực. Do đó, mạng thần kinh sâu thường được sử dụng trong thực tế. Sức biểu diễn mạnh mẽ của mạng thần kinh sâu làm cho nó có thể tạo ra hình ảnh thực tế. Tuy nhiên, một mạng được thiết lập ngẫu nhiên thường tạo ra những hình ảnh trông vô nghĩa. Vậy làm thế nào để bạn huấn luyện một mạng xây dựng để tạo ra những hình ảnh ý nghĩa?

Ta nhớ lại làm thế nào để huấn luyện mạng lưới phân loại. Khi thực hiện một nhiệm vụ phân loại, đầu vào là một hình ảnh và đầu ra là một thể loại. Trong quá trình huấn luyện, bất kể hình ảnh được nhập vào, mạng phân loại sẽ tìm thấy một mục tiêu đầu ra nhất định cho mỗi điểm đầu vào (như trong hình bên trái của *Hình 8-3*). Với mục tiêu huấn luyện, chúng ta có thể tối ưu hóa mạng bằng cách giảm khoảng cách với mục tiêu.

Chúng ta chỉ cung cấp hình ảnh trong mạng tạo sinh và không có thông tin nào khác. Do đó, không có điểm trong không gian tiềm năng trong không gian hình ảnh, và điểm đích được xác định được thể hiện trong hình bên phải của *Hình 8-3*. Không có mục tiêu so sánh trực tiếp, làm cách nào để tối ưu hóa mạng tạo sinh? Tại thời điểm này, bạn cần tạo một thành phần quan trọng khác của mạng - mạng phân biệt.



Hình 8-3: So sánh mạng phân loại và mạng tạo sinh

8.3. Mạng phân biệt

Nhiệm vụ phân biệt mạng là xác định xem một bức tranh có phải là từ dữ liệu thực hay được tạo ra bởi mạng tạo sinh hay không. Trong quá trình huấn luyện mạng phân biệt, mạng được cải thiện khả năng phân biệt bằng cách liên tục nhập hai loại hình ảnh khác nhau và ghi nhãn hai loại hình ảnh với các giá trị khác nhau. Nếu đầu vào là hình ảnh trong dữ liệu thực thì đánh dấu giá trị 1. Nếu đầu vào là hình ảnh được tạo bởi mạng tạo sinh, giá trị là 0. Sau khi mạng được huấn luyện, hình ảnh được nhập vào mạng đó. Nếu nó xác định rằng hình ảnh được tạo ra bởi mạng tạo sinh, output sẽ là 0. Ngược lại, nếu mạng tin rằng hình ảnh này phải đến từ dữ liệu thực, nó sẽ xuất ra 1. Trong trường hợp này, mạng phân biệt xem xét rằng một hình ảnh có thể được tạo ra bởi một máy tính hoặc có thể từ dữ liệu thực, và sau đó nó sẽ xuất ra xác suất rằng hình ảnh là dữ liệu thực.

Việc phân biệt đầu ra của mạng sử dụng một giá trị để cho biết khả năng một điểm trong không gian sẽ đến từ dữ liệu thực. Giá trị đầu ra 0 chỉ ra rằng mạng phân biệt tin rằng hình ảnh phải được tạo tự động bởi máy tính; giá trị đầu ra 1 chỉ ra rằng mạng phân biệt tin rằng hình ảnh phải đến từ dữ liệu thực; giá trị đầu ra là 0.5 cho biết có thể được máy tính tạo tự động hoặc dữ liệu thật. Đôi khi chúng ta gọi mạng phân biệt là **discriminator**.

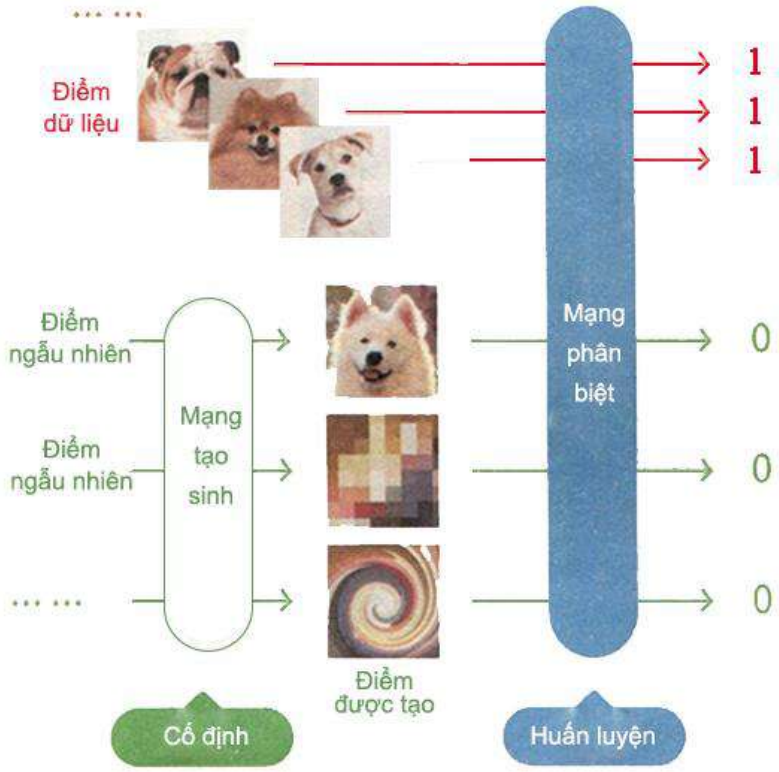
8.4. Hợp tác và tiến bộ trong cuộc đối đầu: Tạo mạng đối đầu

Mạng lưới đối đầu được tạo ra bao gồm hai phần: mạng tạo sinh và mạng phân biệt. Chúng làm việc cùng nhau và chiến đấu với nhau. Chúng được coi là làm việc cùng nhau vì tương tác và hỗ trợ lẫn nhau. Để phân biệt hình ảnh thực và hình ảnh được tạo bởi mạng tạo sinh, cần phải có cả hai loại hình ảnh cùng một lúc. Để tạo ra một hình ảnh tương tự như hình ảnh thực tế, mạng tạo sinh cần phải dựa vào thông tin phản hồi của mạng phân biệt. Đồng thời chúng cũng được coi là đối đầu bởi vì mục tiêu của mạng phân biệt là từ chối để cho những hình ảnh được tạo bởi mạng tạo sinh được trộn lẫn với những hình ảnh thực tế. Mục đích của mạng tạo sinh là tạo ra một hình ảnh giống hình ảnh thực tế càng nhiều càng tốt, để mạng phân biệt không thể

phân biệt được với hình ảnh thực tế. Vậy làm thế nào để chúng tương tác với nhau?

Việc huấn luyện để tạo ra GAN bao gồm hai giai đoạn xen kẽ: Cố định mạng tạo sinh để huấn luyện mạng phân biệt và cho mạng phân biệt cố định để huấn luyện mạng tạo sinh.

Cố định mạng tạo sinh, huấn luyện mạng phân biệt

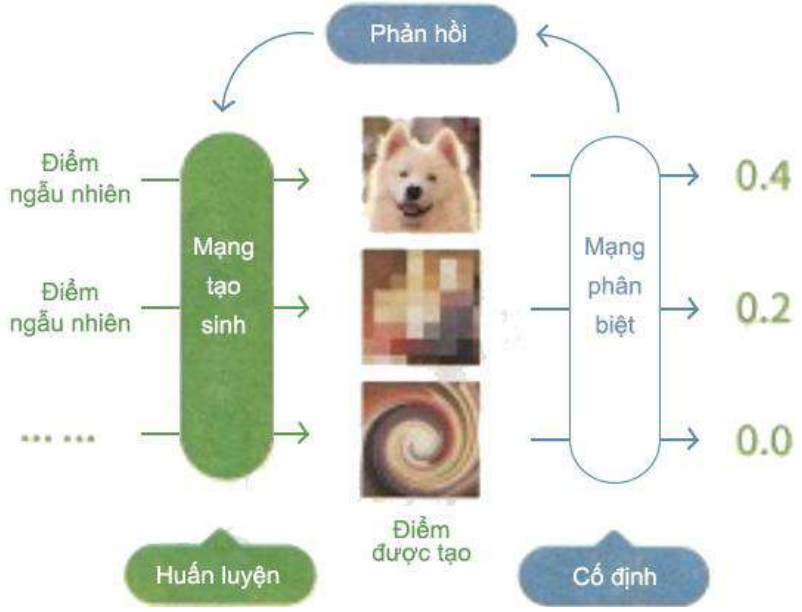


Hình 8-4: Sơ đồ hoạt động giai đoạn Cố định mạng tạo sinh, huấn luyện mạng phân biệt

Hình 8-4 cho thấy các giai đoạn cố định mạng tạo sinh và huấn luyện mạng phân biệt. Đầu tiên, chúng ta tạo ra một số điểm ngẫu nhiên nhất định và sử dụng mạng tạo sinh để biến những điểm ngẫu nhiên này thành hình ảnh được tạo. Sau đó, chúng ta tìm một số lượng hình ảnh thực tế nhất định và tạo bộ dữ liệu hai phân loại với các hình ảnh được tạo. Mục tiêu của phân loại là để cho biết liệu hình ảnh được tạo ra hoặc từ một tập dữ liệu. Trên tập dữ liệu nhỏ này, chúng ta có thể huấn luyện một mạng phân biệt để dự đoán hình ảnh thực là gần 1, và dự đoán của bức ảnh được tạo gần bằng 0, cho phép khả năng phân biệt giữa hình ảnh thực và hình ảnh được tạo ra.

Cố định mạng phân biệt, huấn luyện mạng tạo sinh

Hình 8-5 cho thấy các giai đoạn cố định mạng phân biệt, huấn luyện mạng tạo sinh. Trong giai đoạn này, chúng ta liên tục tạo ra các điểm ngẫu nhiên trong không gian tiềm năng và chuyển đổi các điểm ngẫu nhiên này thành các hình ảnh được tạo bằng mạng tạo sinh. Tiếp theo, chúng ta nhập những hình ảnh được tạo ra vào mạng phân biệt và nhận được xác suất rằng "hình ảnh là một bức tranh thực". Quan trọng hơn, mạng phân biệt cũng sẽ cung cấp phản hồi cho mạng tạo ra cách cải thiện xác suất phân biệt đầu ra. Mạng tạo sinh sử dụng thông tin phản hồi để điều chỉnh các thông số của mạng, sao cho các tác phẩm được tạo ra có thể có được điểm số cao hơn trong mạng phân biệt. Sau một số lượng huấn luyện nhất định, mạng tạo sinh có thể xuất ra một bức ảnh được tạo gần hơn với hình ảnh thực.



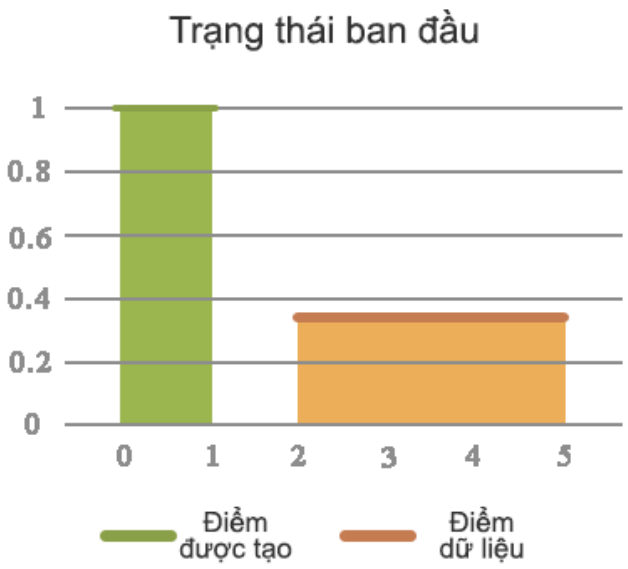
Hình 8-5: Sơ đồ hoạt động giai đoạn Cố định mạng phân biệt, huấn luyện mạng tạo sinh

Suy nghĩ và thảo luận

Sau khi tìm hiểu các phương pháp làm việc cơ bản của GAN, Hãy tưởng tượng khi mạng tạo sinh và mạn phân biệt được huấn luyện luân phiên, khả năng tạo ảnh của mạng tạo sinh có được cải thiện không? Điểm số được tạo ra bởi mạng tạo sinh khi qua mạng phân biệt cuối cùng có tiệm cận 1 không?

Chứng minh sự đối đầu

Chúng ta sử dụng một ví dụ đơn giản hư cấu để chứng minh quá trình hoạt động của mạng tạo sinh và mạng phân biệt. Giả sử rằng dữ liệu được phân bố đều trong khoảng [3.5]. Việc tạo mạng chỉ có thể tạo ra các số giữa các khoảng [0, 1]. Chúng ta hiển thị phân phối của chúng trong không gian như trong *Hình 8-6*.

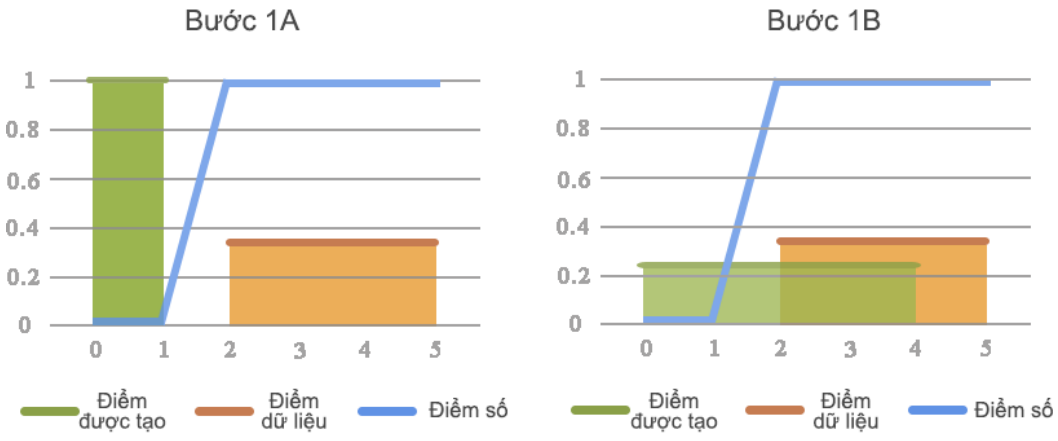


Hình 8-6: Trạng thái phân bố ban đầu của điểm dữ liệu và điểm được tạo

Trong hình, trục hoành đại diện cho không gian nơi các điểm dữ liệu và các điểm được tạo được đặt. Số trên trục tung là giá trị xác suất. Các đường màu xanh lục và màu da cam cho biết xác suất mà điểm tạo và điểm dữ liệu xuất hiện trong không gian tương ứng. Giá trị lớn hơn cho biết khả năng xảy ra lớn hơn.

Tổng các cơ hội của tất cả các giá trị xuất hiện là 1, vì vậy tổng diện tích của hình chữ nhật bên dưới mỗi dòng là 1. Không gian được tạo ra nhỏ, giữa [0,1], vì vậy hình chữ nhật màu xanh lá cây mỏng và cao; phạm vi không gian của các điểm dữ liệu lớn, giữa [2, 5], do đó, khu vực màu da cam là một hình chữ nhật đó là thấp và rộng.

Sau khi bắt đầu học tập, kết quả của vòng đầu tiên được thể hiện trong *Hình 8-7*.

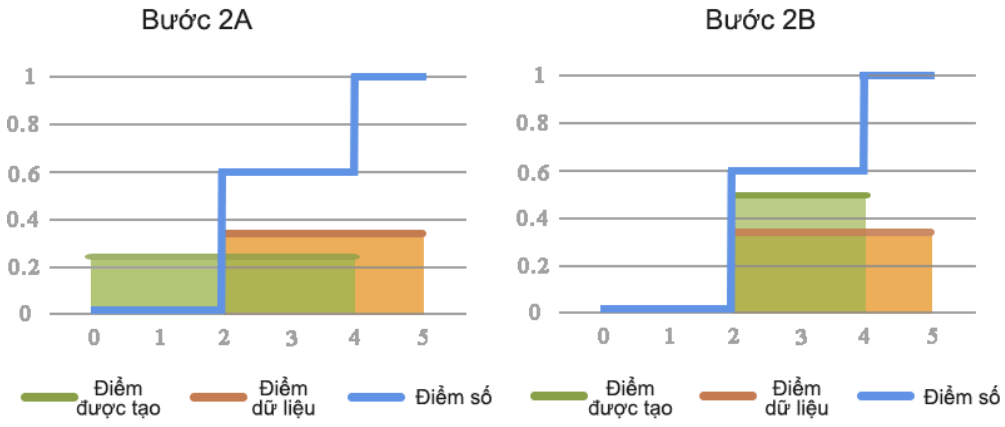


Hình 8-7: Hình bên trái hiển thị kết quả phân biệt mỗi điểm sau khi mạng được huấn luyện. Hình bên phải là những điểm được tạo của mạng tạo sinh sau khi điều chỉnh từ bước 1A.

Bước 1A (thể hiện trong hình bên trái trong *Hình 8-7*) là một sơ đồ về kết quả huấn luyện mạng phân biệt. Đường màu xanh (số điểm nhận dạng) cho biết điểm đầu ra của mạng phân biệt. Chúng ta quan sát thấy điểm được tạo (đoạn màu xanh lục trong hình) nằm xa điểm dữ liệu (đoạn đường màu da cam trong hình), có nghĩa là có thể phân biệt rõ ràng các điểm được tạo ra (điểm đầu ra tương ứng là 0), điểm nào là dữ liệu thực (điểm đầu ra tương ứng là 1).

Mạng tạo sinh muốn đạt một điểm số phân biệt cao hơn, do đó, một chiến lược điều chỉnh mở rộng phạm vi được áp dụng. Bằng cách điều chỉnh ở bước 1B (như trong hình bên phải *Hình 8-7*), phạm vi đầu ra của mạng tạo sinh và khu vực có điểm số là 1 đã chồng lên nhau một phần, cho thấy mạng tạo sinh đã thu được phản hồi hữu ích từ mạng phân biệt và đã tối ưu hóa đến một mức nhất định.

Sau vòng hoạt động đầu tiên, mạng cần phải tự đổi mới trong vòng thứ hai của hoạt động để đánh giá chính xác dữ liệu mới nhất trong mạng tạo sinh “đối thủ” (như trong hình bên trái *Hình 8-8*).



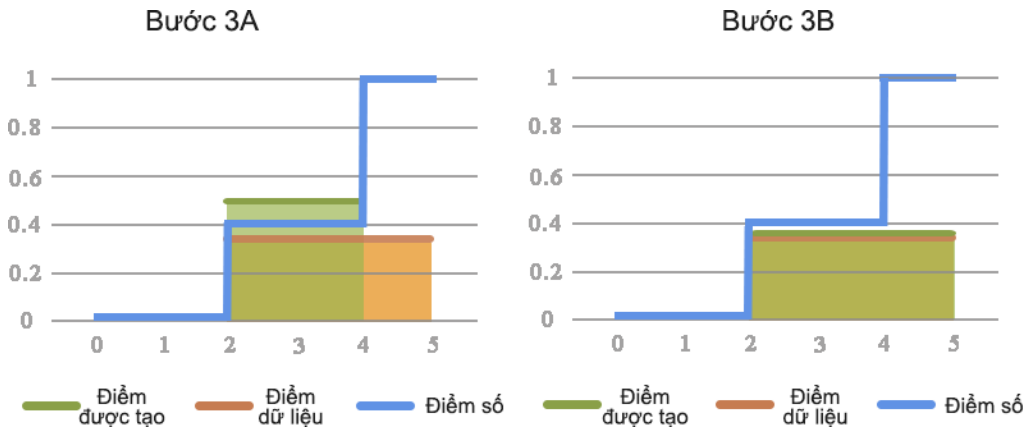
Hình 8-8: Hình bên trái là kết quả phân biệt sau bước 1, hình bên phải hiển thị những điểm được tạo sau khi điều chỉnh từ bước 2A.

Như có thể thấy từ biểu đồ bên trái của *Hình 8-8*, điểm phân biệt đạt gần 0,6 điểm cho các điểm được phân bố giữa các khoảng [2, 4], cho thấy rằng mạng phân biệt tin rằng có khoảng 60% điểm trong khu vực này có thể đến từ một bộ dữ liệu thực sự. Tại thời điểm này, sau khi mạng tạo sinh nhận được phản hồi, nó sẽ chuyển phạm vi điểm mà điểm số quá thấp (tức là điểm trên phạm vi không gian [0, 2]), thành điểm trong khoảng [2, 4] (như thể hiện trong hình bên phải trong *Hình 8-17*), mở rộng khoảng [2, 4]. Như có thể thấy từ hình, xác suất tạo ra một điểm trên khoảng [2,4] này đã cao hơn điểm dữ liệu.

Suy nghĩ và thảo luận

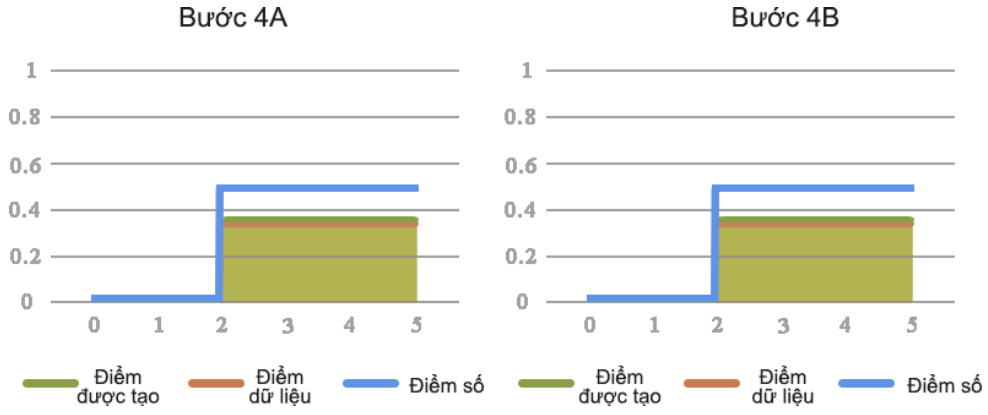
Từ *Hình 8-8*, chúng ta quan sát thấy các điểm được tạo ra bởi mạng tạo sinh đều nằm trong phạm vi không gian của các điểm dữ liệu. Nếu chúng ta tạo ra một hình ảnh, hình ảnh kết quả bây giờ rất gần với hình ảnh thực. Hạn chế duy nhất là bộ tạo dữ liệu giữa [4, 5] trên không gian không thể được tạo ra, thuật toán có thể giải quyết vấn đề này không?

Sau vòng hoạt động cuối cùng, mạng phân biệt sẽ tự cập nhật lại, giảm ngưỡng cho các điểm trong khoảng [2, 4] (hiển thị ở bên trái trong *Hình 8-9*). Mạng tạo sinh nhận được phản hồi về điểm cao hơn cho khoảng [4, 5], do đó mạng sẽ tiếp tục tối ưu hóa, và kết quả tối ưu được thể hiện trong hình bên phải của *Hình 8-18*. Lúc này này, chúng ta thấy rằng khu vực có điểm được tạo đặt chính xác giống với khu vực có điểm dữ liệu. Mọi người có thể sẽ lo lắng, nếu tiếp tục chạy, hệ thống sẽ phá vỡ trạng thái tốt nhất này và trở nên tồi tệ hơn?



Hình 8-9: Hình bên trái là kết quả dữ liệu sau khi được cập nhật, hình bên phải hiển thị kết quả sau khi tối ưu.

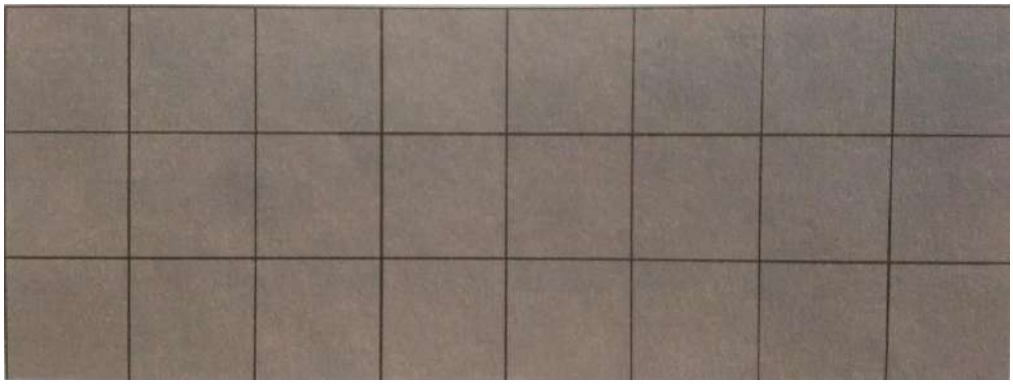
Hình bên trái Hình 8-9 cho thấy mối quan hệ giữa các kết quả dữ liệu sau khi cập nhật mạng. Hình bên phải cho thấy kết quả sau khi tối ưu hóa mạng. Sau vòng hoạt động cuối cùng, đầu ra của các điểm trong mỗi khoảng được xác định là 0,5 (như trong hình bên trái trong Hình 8-19). Điều này có nghĩa là mạng phân biệt hoàn toàn không thể xác định được các điểm là được tạo ra bởi bộ tạo hay từ dữ liệu thực. Đối với mạng tạo sinh, nó sẽ không thay đổi vì điểm số của điểm được tạo ra và điểm dữ liệu giống nhau. Mạng tạo sinh nghĩ rằng hình ảnh được tạo ra không khác với hình ảnh thực (Hình bên phải *Hình 8-10*). Vì vậy, toàn bộ mạng đã đạt đến trạng thái ổn định.



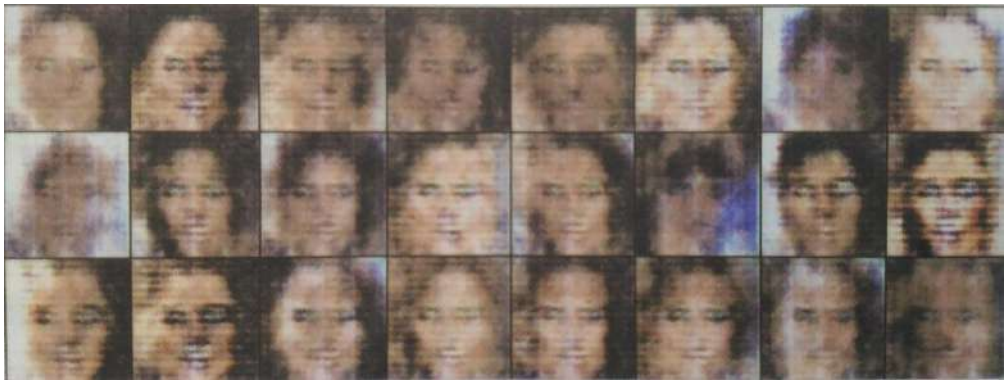
Hình 8-10: Hình bên trái cho thấy kết quả dữ liệu sau khi phân biệt với bản cập nhật mạng tạo sinh cung cấp và hình ảnh bên phải cho thấy rằng mạng tạo sinh không được cải thiện nữa và đã đạt trạng thái cân bằng ổn định.

Thông qua việc chứng minh một quá trình học tập động như vậy, chúng ta đã tìm hiểu trong việc tạo ra GAN, các mạng sẽ tự động cộng tác và cạnh tranh với nhau, cuối cùng đạt được trạng thái ổn định lý tưởng.

Dưới đây là các kết quả trong một thí nghiệm tạo hình ảnh ca sĩ với GAN:



Hình 8-11: Lần chạy 1, hình ảnh được tạo ra là màu xám đơn giản



Hình 8-12: Lần lặp lại thứ 1000, hình ảnh được tạo ra có thể thấy nguyên mẫu của khuôn mặt

Hình 8-22: Lần lặp thứ 3000, kết quả hình ảnh thay đổi theo đúng hướng

Hình 8-23: Lần lặp thứ 5000, một phần hình ảnh tạo ra đã có hiệu ứng khá tốt

8.5. GAN có điều kiện

GAN có thể tạo ra một hình ảnh giống như hình ảnh thực, nhưng nội dung nó tạo ra là ngẫu nhiên. Làm thế nào để ta có thể tạo ra một hình ảnh với các thuộc tính được chỉ định, chẳng hạn như một ca sĩ đeo kính? GAN có điều kiện có thể giải quyết vấn đề này. Nó có thể tạo ra hình ảnh phù hợp với các tiêu chí nhất định. Mặc dù nó chưa bước vào giai đoạn thực tế, công nghệ này đã cho thấy triển vọng ứng dụng tuyệt vời. Chúng ta sử dụng hình ảnh để hiển thị hai ví dụ:

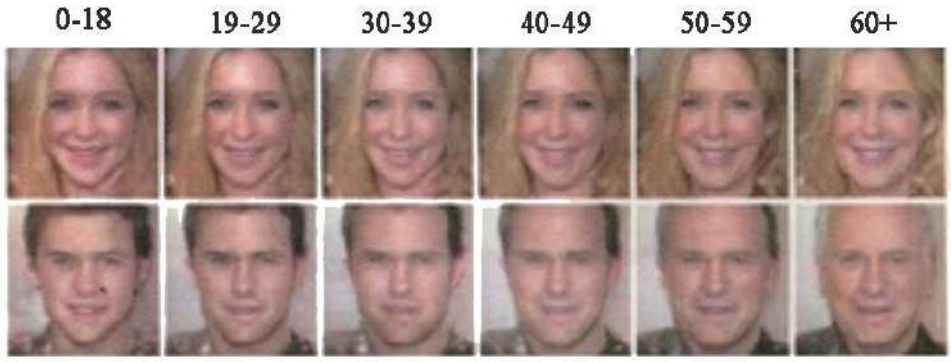
Từ mặt bên đến mặt chính diện: Ứng dụng trong xác định tội phạm

Hình 8-24: Ứng dụng để tạo ảnh mặt chính diện từ các ảnh mặt bên

Trong ví dụ này, điều kiện là ảnh mặt bên của một người và mục tiêu là khuôn mặt nhìn từ chính diện giống người đó (ảnh từ bài báo “Beyond Face

Rotation: Global and local Perierption GAN for Photorealistic and Identity Preserving Frontal View Synthesis").

Từ trẻ đến già: Ứng dụng trong tìm trẻ lạc



Hình 8-25: Ứng dụng trong tạo ảnh với các độ tuổi khác nhau

Trong ví dụ này, điều kiện là ảnh khuôn mặt của một độ tuổi nhất định và mục tiêu tạo sinh của nó là ảnh của cùng một người đó nhưng ở một độ tuổi khác (ảnh từ bài báo “Face Aging with Conditional Generative Adversarial Networks”).

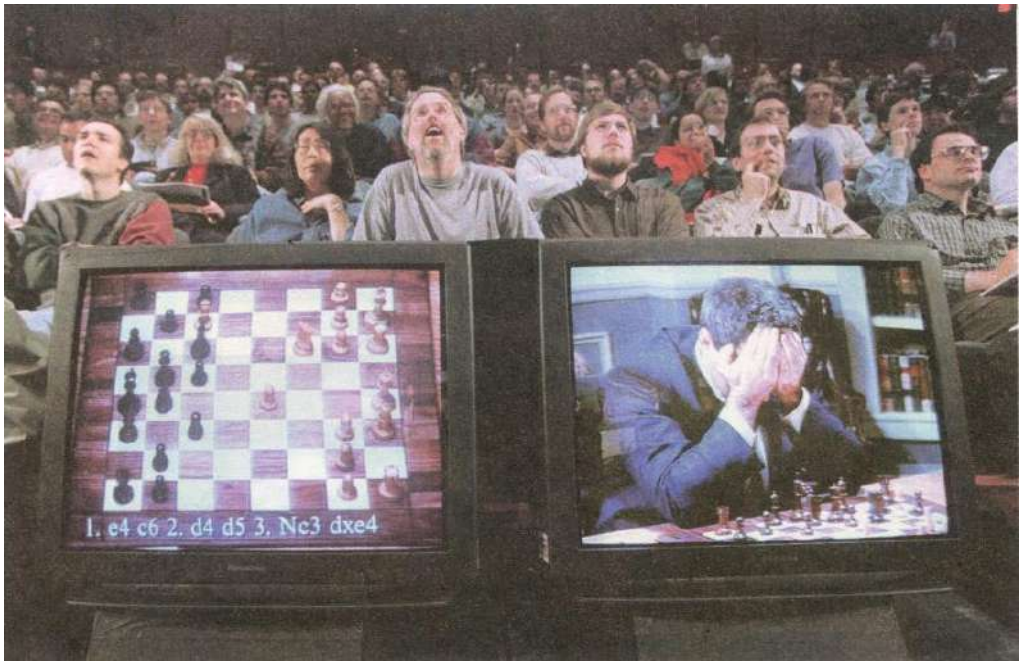
8.6. Tóm tắt chương

Chương này đã giới thiệu cách sử dụng GAN để cho phép máy tính tự động tạo hình ảnh thực tế. Trong quá trình huấn luyện, mạng tạo sinh và mạng phân biệt hợp tác và đối đầu với nhau để đạt mục tiêu. Trạng thái cân bằng tương đương với việc tạo ra các hình ảnh thực.Mạng tạo sinh cung cấp các mẫu huấn luyện cho mạng phân biệt và mạng phân biệt cung cấp các phản hồi là mục tiêu tối ưu hóa cụ thể cho mạng tạo sinh. Mục đích tối ưu hóa của mạng tạo sinh là sử dụng hình ảnh được tạo ra để đánh lừa việc xác nhận của mạng phân biệt. Việc phân biệt này là xác định liệu mẫu đầu vào là được tạo hay dự liệu thực. Mô hình là một loại mô hình trí tuệ nhân tạo với một loạt các ứng dụng.

Việc tạo ra các GAN được mô tả trong chương này là một ví dụ điển hình về việc tạo ra .... Một mặt, nó có thể được sử dụng trong lĩnh vực hình ảnh để tạo ra hình ảnh chất lượng tốt hơn. Mặt khác, các ý tưởng được thể hiện rộng rãi và được áp dụng trong các nhiệm vụ khác.

Chương IX. Bậc thầy cờ vây: AlphaGo

Năm 2016, chương trình trí tuệ nhân tạo AlphaGo được sinh ra và đánh bại các tuyển thủ cờ vây chuyên nghiệp hàng đầu thế giới. Giới hạn cuối cùng của trí tuệ con người đã bị phá vỡ bởi chương trình trí tuệ nhân tạo.



Câu chuyện bắt đầu với phần mềm cờ vua máy tính. Năm 1997, siêu máy tính của IBM "Deep Blue" đã đánh bại nhà vô địch thế giới cờ vua Gary Kasparov và tạo lên sự khuấy động trên thế giới. Ngược lại, sự phát triển của các chương trình cờ vây lúc đó rất chậm, lý do vì có rất nhiều điểm hạ cờ, nhiều tình huống và độ phức tạp của nó cao hơn nhiều so với các trò chơi cờ khác như cờ vua. Vì vậy để một máy tính có thể giành chiến thắng một tuyển thủ cờ chuyên nghiệp là điều vô cùng khó khăn. Kể từ khi “Deep Blue” giành chiến thắng, sau hơn 10 năm phát triển, chương trình trí tuệ nhân tạo cờ vây chỉ đạt tới mức độ nghiệp dư và không thể đánh bại các tuyển thủ chuyên nghiệp.

AlphaGo là một chương trình trí tuệ nhân tạo cờ vây được phát triển bởi nhóm Deep Mind của Google vào năm 2014. Sự xuất hiện của nó là 1 dấu ấn mạnh mẽ trong lịch sử cờ vây trên thế giới. Vào tháng 10 năm 2015, AlphaGo đã đánh bại tuyển thủ Fan Hui tại giải cờ vây Châu Âu và trở thành chương trình máy tính đầu tiên đánh bại tuyển thủ chuyên nghiệp mà không bỏ cuộc. Tuy nhiên, nhiều nhà bình luận và chuyên gia trí tuệ nhân tạo vẫn tin rằng khả năng cờ vây của Fan Hui vẫn cách xa giải vô địch thế giới và AlphaGo không thể đánh bại các tuyển thủ cờ vây hàng đầu thế giới. Vào tháng 3 năm 2016, AlphaGo và nhà vô địch cờ vây trên thế giới Lee Sedol (cao thủ hạng 9-dan) đã thi đấu với nhau, cuối cùng AlphaGo đã chiến thắng với số điểm 4 - 1, và gây bất ngờ cho toàn thế giới.

Vào tháng 5 năm 2017, tại hội nghị tương lai cờ vây (Ô Chấn - Trung Quốc), phiên bản nâng cấp của AlphaGo đã chiến thắng tuyển thủ Kha Khiết, nhà vô địch cờ vây số 1 thế giới với số điểm 3 – 0. Thế giới đã công nhận sức mạnh cờ vây của chương trình trí tuệ nhân tạo AlphaGo. Nó đã chiến thắng những tuyển thủ giỏi nhất, chuyên nghiệp nhất của con người.

Reinforcement learning (Học tăng cường) đóng góp rất lớn cho sự thành công của AlphaGo, điều đó cũng khiến "Reinforcement learning trở thành điểm nóng trong nghiên cứu trí tuệ nhân tạo hiện nay". Chúng ta hãy tìm hiểu về AlphaGo để khám phá sự kỳ diệu của Reinforcement learning, thấy được cách để máy tính "hoạt động" như một con người và trở thành bậc thầy cờ vây mà không cần ai dạy.

9.1. Mạng nơron AlphaGo

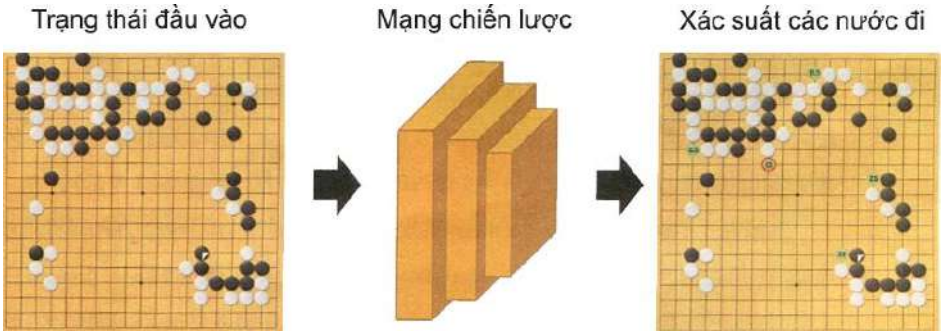
Như đã từng nói phía trên, AlphaGo là 1 chương trình trí tuệ nhân tạo. Cách hoạt động của nó là trong quá trình thi đấu với con người, AlphaGo sẽ cần quyết định nước cờ nên đặt cho tình huống hiện tại để giành chiến thắng. Quá trình đưa ra quyết định đó là trách nhiệm của mạng nơ-ron trong AlphaGo. Mạng Nơ-ron này sẽ mô tả bàn cờ dưới dạng input và xử lý nó thông qua một số lớp mạng khác nhau (through a number of different network layers) chứa hàng triệu kết nối như nơ-ron.

Một mạng nơ-ron được gọi là policy network sẽ tiếp nhận input và đưa ra output là vị trí được chọn để đặt quân cờ tiếp theo. Một mạng nơ-ron khác là mạng giá trị sẽ dự đoán tỷ lệ thắng cuộc trong trò chơi.Mạng nơ-ron AlphaGo đầu tiên được huấn luyện bởi kĩ thuật học tập có giám sát.

Học tập có giám sát

DNN được sử dụng để thực thi nhiệm vụ này. Thông tin vào của mạng lưới không chỉ là trạng thái hiện tại của bàn cờ, mà còn là các tham số của rất nhiều cấu trúc nhân tạo cũng được thêm vào, ví dụ như các vị trí trống trên bàn cờ.

Để thực hiện huấn luyện, đội ngũ AlphaGo đã tập hợp 160,000 người chơi online cờ vây trên diễn đàn cờ vây KGS, và dùng 30,000,000 mẫu từ đó để làm mẫu huấn luyện. Với mỗi mẫu bao gồm trạng thái hiện tại của bàn cờ gọi là s, và nước cờ tiếp theo mà người chơi đặt cờ gọi là a, ký hiệu (s, a). 30 triệu mẫu được sử dụng để huấn luyện trong mạng học tập có giám sát. Từ đó thu được một mạng các chiến lược học tập có giám sát và nó có thể mô phỏng lại phong cách của mỗi người chơi.



Hình 9-1: Sơ đồ học tập có giám sát

Mặc dù đã có thể bắt chước người chơi cờ, tuy nhiên phương pháp này có vấn đề rất lớn, bởi vì trình độ người chơi trên KGS là khác nhau, không phải mẫu nào cũng là lựa chọn tốt nhất. Hơn nữa những người đứng top chơi rất ít và không đủ mẫu để huấn luyện mạng lưới. Quả thực sau khi huấn luyện, khả năng chơi cờ của mạng lưới chiến thuật mới dừng ở mức độ nghiệp dư, và hoàn toàn không thể thi đấu với các tuyển thủ hàng đầu thế giới.

Phân tích lý do, ngoài việc trình độ người chơi chênh lệch, một lý do quan trọng khác là mẫu (s, a) chỉ quan tâm chơi như thế nào mà không cần biết kết quả thắng hay thua, bởi vậy nó sẽ cần học đâu là nước cờ tốt và đâu là nước cờ xấu. Vậy làm thế nào để cải thiện khả năng chơi cờ?

Bởi vậy, để cải thiện khả năng chơi cờ, AlphaGo sử dụng phương pháp Học tăng cường (Reinforcement learning). Thông qua phương pháp này, AlphaGo có thể cải thiện khả năng chơi cờ bằng cách "phân thân" và tự thi đấu với mình. Vậy Học tăng cường là gì?

Học tăng cường và các khái niệm cơ bản

Học tăng cường và học có giám sát đều là một loại thuật toán của máy học. Sự khác biệt giữa học tăng cường và học có giám sát đó là học tăng cường trả về sự đánh giá chứ không phải chỉ dẫn. Các thông tin ra của học có giám sát sẽ cung cấp cách xử lý cho hệ thống thông qua các tín hiệu giám sát. Học tăng cường sẽ đưa ra đánh giá cách xử lý nào tốt, cách xử lý nào xấu. Vì vậy hệ thống sẽ cần đưa ra được cách xử lý tối ưu nhất sau nhiều lần thử; Mỗi bước đi không chỉ ảnh hưởng đến bàn cờ hiện tại mà còn ảnh hưởng tới bước đi tiếp theo.

Học tập tăng cường cho phép máy tính có các kỹ năng tự hoàn thiện và học tập như còn người và trở thành một trí thông minh nhân tạo thực sự.

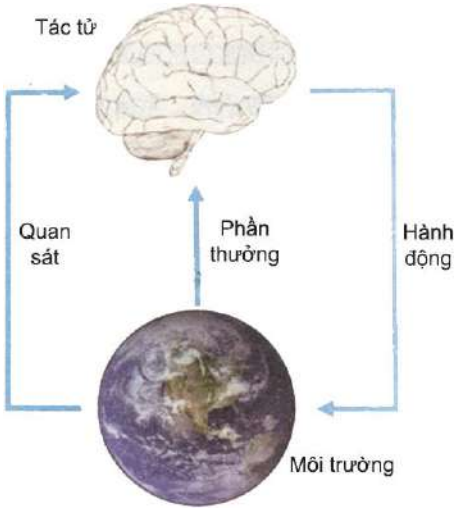
AlphaGo cũng sử dụng ý tưởng tự học này để cải thiện khả năng chơi cờ thông qua việc tự chơi với chính nó.

Tương tác giữa tác tử và môi trường

Sau khi hiểu các khái niệm cơ bản về học tập tăng cường, chúng ta hãy xem xét các yếu tố cơ bản của việc học tăng cường.

Trong học tập tăng cường, chúng ta gọi chủ thể chịu trách nhiệm đưa ra quyết định là **tác tử** (agent), ví dụ như AlphaGo, xe ô tô không người lái, người máy... Tác tử tồn tại trong môi trường, hành động của nó hoạt động trên **môi trường** (environment) và nhận phản hồi từ môi trường. Học tăng cường là nghiên cứu về sự tương tác giữa tác tử và môi trường (như trong *Hình 9-2*).

Tác tử lựa chọn **hành động** (action) làm **trạng thái** (state) của môi trường thay đổi. Sau khi trạng thái bị thay đổi, nó sẽ đưa ra sự **quan sát** (observation) môi trường và một **phần thưởng** (reward). Cùng lúc đó, tác tử có thể đưa ra một hành động mới dựa trên thông tin phản hồi đó và tiếp tục tương tác với môi trường. Nói một cách tổng quan, sau khi tác tử tác động vào môi trường thông qua hành động, thì sẽ phản hồi lại trạng thái mới của môi trường là tốt hay xấu.



Hình 9-2: Tương tác giữa tác tử và môi trường

Trong cờ vây, AlphaGo là một tác tử, và trạng thái của bàn cờ được coi là trạng thái môi trường. AlphaGo cần đối mặt với trạng thái hiện tại của trò chơi và chọn vị trí đặt quân cờ tiếp theo. Trạng thái mới của trò chơi sau khi AlphaGo đặt quân cờ là một sự quan sát mà môi trường trả về, và liệu AlphaGo có giành chiến thắng trong trò chơi hay không cũng là sự phản hồi từ môi trường.

Chiến lược và mục đích học tăng cường

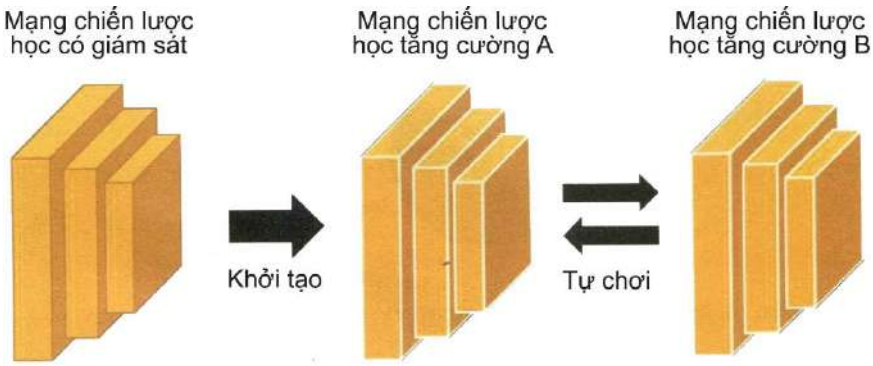
Có một mối liên hệ ánh xạ giữa tập hợp trạng thái (state) của môi trường và tập hợp hành động (action) của tác tử, đó là tác tử cần thực hiện hành động nào để có được trạng thái (state) mới của môi trường. Nói một cách tổng quan hơn, trong mỗi trạng thái, xác suất chuyển trạng thái (s), từ hành động (a) thành một trạng thái mới (s') thường là ngẫu nhiên. Chiến lược (Policy) - việc lựa chọn hành động của tác tử, là một ánh xạ từ tập trạng thái tới tập hành động. Ví dụ trong cờ vây, tập trạng thái của môi trường bao gồm tất cả các trạng thái có thể của trò chơi, tập hành động của tác tử là tất cả các bước đi mà AlphaGo có thể thực hiện, và chiến lược là cách xử lý của AlphaGo. Có thể nói là trong mỗi trạng thái khác nhau, AlphaGo sẽ lựa chọn ra chiến lược để chơi cờ.

Mục đích của Học tăng cường là lựa chọn ra một chiến lược tốt nhất. Bởi vậy sau khi tác tử đưa ra được một tập các hành động, nó sẽ nhận lại sự tích lũy phần thưởng từ các hành động đó. Hay nói cách khác, AlphaGo sẽ cần tìm ra chiến lược tốt nhất (phần thưởng nhiều nhất) thông qua học tăng cường. Dựa trên chiến lược này, AlphaGo có thể giành chiến thắng nhiều lần nhất qua nhiều lần chơi.

Mạng chiến lược học tăng cường

Sau khi đã hiểu cơ bản về học tăng cường, vậy làm cách nào để huấn luyện học tăng cường? Và làm thế nào để huấn luyện mạng chiến lược mạnh mẽ hơn thông qua việc học tăng cường?

Đầu tiên, chúng ta tìm hiểu về huấn luyện học tăng cường, quá trình xử lý thực sự được thực hiện trong sự tương tác giữa tác tử và môi trường. Tác tử sẽ tiếp tục thực hiện các hành động làm thay đổi trạng thái của môi trường từ trạng thái hiện tại và trả về sự thay đổi trạng thái tương ứng và môi trường sẽ cho biết hành động hiện tại có mang lại lợi ích (phần thưởng) hay không. Sau khi nhận được sự phản hồi từ môi trường, tác tử sẽ điều chỉnh chiến lược nhanh chóng tùy vào phản hồi nhận được. Vì vậy hành động được đưa ra dưới sự chỉ dẫn của chiến lược mới sẽ nhận được phần thưởng cao hơn trong tương lai. Sau nhiều lần lặp lại, tác tử có thể liên tục điều chỉnh chiến lược theo sự phản hồi từ môi trường và từ từ tiếp cận chiến lược tốt nhất.



Hình 9-3: Huấn luyện mạng học tăng cường

AlphaGo sử dụng công nghệ học tăng cường có tên Strategy Gradient để huấn luyện mạng chiến lược "mạnh mẽ" hơn được gọi là Mạng chiến lược học tăng cường. Mạng này sử dụng mạng chiến lược học tập có giám sát để khởi tạo và sau đó liên tục tự chơi, sau khi đặt được mục tiêu giành chiến thắng, nó sẽ lặp lại và cập nhật mạng các tham số, từ đó cải thiện chiến lược và nâng xác suất chiến thắng. Mỗi lần, 2 bên thi đấu của trò chơi là phiên bản hiện tại của AlphaGo và chọn ngẫu nhiên một phiên bản AlphaGo trong các lần lặp trước đó. Sau khi kết thúc, kết quả sẽ dựa trên kết quả (outcome) của phiên bản AlphaGo hiện tại, nếu cuối cùng chiến thắng thì phản hồi tích cực và nhận khoản thưởng, còn không sẽ phản hồi tiêu cực và các thông số của mạng chiến lược sẽ thay đổi theo hướng tối đa hóa khoản thưởng thông qua kỹ thuật Strategy Gradient. Do đó, mục tiêu của việc huấn luyện mạng chiến lược học tăng cường không còn là mô phỏng lại phong cách của người chơi, mà là để giành được chiến thắng.

Sau khi huấn luyện và học tập tăng cường, khả năng chơi cờ của mạng chiến lược được tăng lên rất nhiều, nó đã đạt được tỷ lệ thắng lên đến 80%.

Trong quá trình cải thiện mạng chiến lược AlphaGo, chúng ta đã thấy sức mạnh của học tăng cường và trong phần tiếp theo, chúng ta sẽ tiếp tục tìm hiểu cách học tăng cường để giúp AlphaGo có khả năng dự đoán trước.

9.2. Tầm nhìn của AlphaGo

Mặc dù khả năng của mạng chiến lược đã tăng lên rất nhiều sau khi được huấn luyện học tăng cường, nhưng nó vẫn còn những thiếu sót rõ ràng, đó là phiên bản của AlphaGo chỉ đưa ra quyết định dựa trên trạng thái hiện tại của trò chơi, nó đã giống như một cao thủ nhưng chỉ chơi theo bản năng,

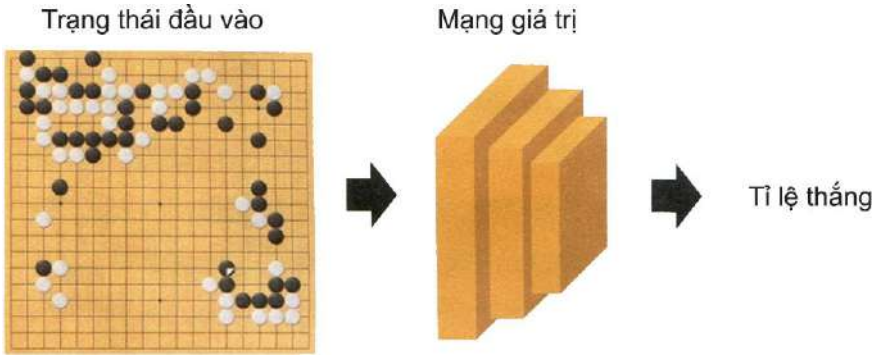
thiếu cái gọi là “nhìn xa trông rộng”. Cái gọi là “đi một bước, tính 3 bước”, các cao thủ cờ vây sẽ tập trung vào trọng tâm của trò chơi giúp họ tìm ra một kế hoạch cho trạng thái hiện tại. Sự thiếu sót đó của AlphaGo rõ ràng chưa đủ năng lực để thi đấu với các cao thủ cờ vây. Để AlphaGo có thể “nhìn xa trông rộng”, đội ngũ Deep Mind đã thêm một mạng giá trị (value network) vào AlphaGo, cùng lúc đó thuật toán cây tìm kiếm Monte Carlo cũng được áp dụng để phát triển tình hình hiện tại giúp AlphaGo tìm ra phương pháp tốt hơn.

Mạng giá trị

Mạng giá trị trong AlphaGo được sử dụng để định lượng trạng thái hiện tại của bàn cờ, nó cho phép AlphaGo có thể nhanh chóng xác định được tỷ lệ chiến thắng trên trạng thái hiện tại mà không cần hoàn thiện trò chơi. Mạng giá trị nhận trạng thái hiện tại của bàn cờ là thông tin vào và dự đoán tỷ lệ chiến thắng của AlphaGo trong trạng thái đó.

Khi mạng giá trị được huấn luyện, đội ngũ AlphaGo thấy rằng trò chơi với con người không thể đưa ra được một hàm giá trị (value function) tốt. Bởi vậy AlphaGo một lần nữa sử dụng kỹ thuật học tăng cường, tức là phương pháp để máy tự chơi với máy tạo thành một cặp, từ đó tạo ra đủ số lượng mẫu để huấn luyện mạng giá trị. AlphaGo tự chơi với chính nó bằng cách sử dụng mạng đào tạo chiến lược học tăng cường, từ đó có được 30 triệu mẫu dưới dạng (s, z), s biểu diễn trạng thái và z là kết quả cuối cùng của trò chơi. Mỗi mẫu là một trò chơi khác nhau do đó loại bỏ sự tương quan giữa các mẫu với nhau.

Việc huấn luyện mạng giá trị bởi các mẫu được tạo ra từ quá trình tự chơi mang lại hiệu quả cao, đến thời điểm hiện tại, AlphaGo có thể dự đoán tỷ lệ chiến thắng của cả 2 bên mà không cần đợi trận đấu kết thúc, điều này khiến AlphaGo có khả năng loại trừ trong một khoảng thời gian giới hạn (a limited time) để tìm ra giải pháp tốt nhất.



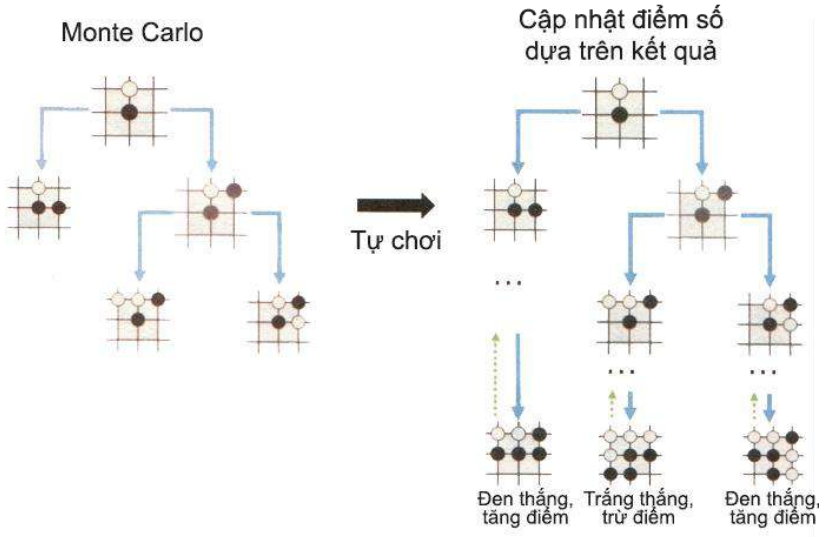
Hình 9-4: Sơ đồ mạng giá trị

Fast moving network

Để đẩy nhanh tốc độ của trò chơi, AlphaGo được thêm fast-moving network, có thể coi nó là một policy network nhỏ, mặc dù hiệu quả của nó không bằng Strategy network, nhưng tốc độ của nó gấp 1000 lần Strategy network. Ưu điểm của Fast moving network là nó có thể nhanh chóng mô phỏng các khả năng tương lai khi thực thi thuật toán tìm kiếm Cây tìm kiếm Monte Carlo, qua đó giúp máy tính có thể đánh giá trạng thái hiện tại một cách tốt hơn.

Cây tìm kiếm Monte Carlo

Măc dù đã có một Mạng giá trị để dự đoán tỷ lệ chiến thắng hiện tại, nhưng không đủ thời gian để chạy tất cả các trạng thái của AlphaGo, do đó thuật toán cây tìm kiếm Monte Carlo được sử dụng cho trạng thái hiện tại. Cây tìm kiếm Monte Carlo là một thuật toán tìm kiếm heuristic để thiết lập một cây tìm kiếm bằng cách loại trừ ngẫu nhiên. Chúng ta cũng có thể hiểu nó như thuật toán học tăng cường. Trong trò chơi, thuật toán Monte Carlo sẽ bắt đầu với trạng thái hiện tại của bàn cờ và thực hiện quá trình tự chơi và kết quả sẽ có 1 bên thắng và 1 bên thua, bên thắng sẽ được cộng điểm, ngược lại bên thua sẽ bị trừ điểm. Sau đó thuật toán sẽ quay lại và tiếp tục tự chơi, đồng thời tăng điểm cho bên thắng và giảm điểm của bên thua. Vì vậy, tỷ lệ chọn hướng đi để chiến thắng sẽ tăng lên khi gặp được tình huống tương tự sau này. Khi lặp lại các bước ở trên, máy tính sẽ kiểm tra một khả năng có thể xảy ra trong tương lai, giải pháp có điểm số cao hơn sẽ tiếp tục được cải thiện, qua đó giúp máy tính lựa chọn được giải pháp tốt hơn cho hiện tại để giành chiến thắng trong tương lai.



Hình 9-5: Sơ đồ cây tìm kiếm Monte Carlo

Hiện giờ AlphaGo đã làm chủ mạng chiến lược để đưa ra tỷ lệ chiến thắng của mỗi bước đi. Fast-moving network, value network có khả năng phán đoán giá trị cờ và thuật toán tìm kiếm Cây tìm kiếm Monte Carlo có thể phát triển khả năng chơi cờ. Vậy làm thế nào để tập hợp các module này để tạo thành một chương trình AlphaGo nổi tiếng?

Câu trả lời đó là kết nối các module khi cây tìm kiếm Monte Carlo tạo ra trò chơi.

AlphaGo cần đối diện với trạng thái hiện tại, nó cần sử dụng thuật toán Cây tìm kiếm Monte Carlo để đẩy nhanh tiến triển của trò chơi, nghĩa là có thể mô phỏng nhiều khía cạnh khác nhau của trò chơi. Tại thời điểm này, AlphaGo không còn sử dụng thuật toán ngẫu nhiên để chọn nước cờ tại mỗi bước của heuristic, thay vào đó sẽ lựa chọn vị trí dựa trên sự kỳ vọng (the expected) tại mỗi vị trí. Sự kỳ vọng này là sự kết hợp chức năng của các module. Bao gồm phân thân thành 2 từ trạng thái hiện tại thông qua fast-moving sub-network tới khi phân rõ thắng thua, và tính toán xác suất phân bố của từng tình huống bằng mạng chiến lược, ước tính giá trị của từng tình huống bằng mạng giá trị, và sự kỳ vọng sẽ liên tục được cập nhật trong quá tình trò chơi tiến triển.

Sau mỗi lần tự phân thân chơi, sự kỳ vọng trong mỗi bước của trò chơi sẽ được cập nhật. Do đó, sau nhiều lần thử, chúng ta sẽ có một ước lượng ổn định về sự kỳ vọng của các lựa chọn bước đi trong tình hình hiện tại, từ đó giúp chúng ta có thể chọn lựa một phương án tốt nhất.

Tại thời điểm này, AlphaGo được tích hợp hệ thống mạng chiến lược, mạng giá trị và fast-moving sub-network vào trong thuật toán cây tìm kiếm

Monte Carlo, sử dụng khả năng chơi cờ vượt trội và cuối cùng đánh bại các cao thủ cờ vây để trở lên nổi tiếng.

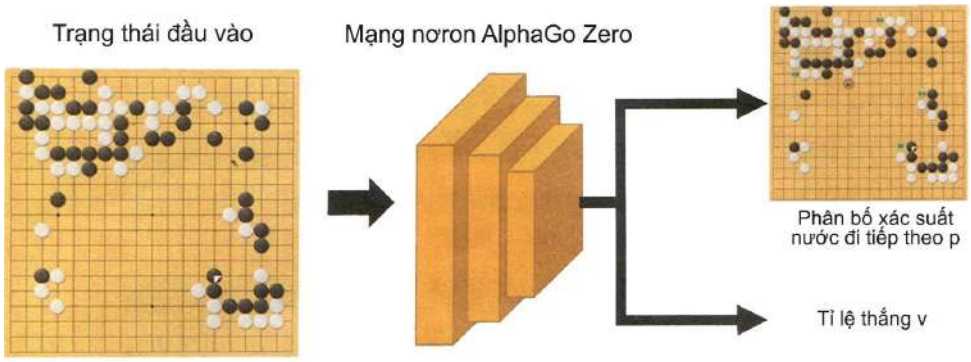
9.3. Thiên tài cờ vây: AlphaGo Zero

Sau khi AlphaGo nổi tiếng, đội ngũ DeepMind vẫn chưa thỏa mãn và tiếp tục tung ra phiên bản trí tuệ nhân tạo nâng cấp hơn, chương trình AlphaGo Zero. Phiên bản trước AlphaGo được đào tạo ban đầu là hàng nghìn mẫu trò chơi nghiệp dư và chuyên nghiệp của con người để học cách chơi cờ vây. AlphaGo Zero bỏ qua bước này và học cách chơi đơn giản bằng cách tự chơi với chính nó, bắt đầu từ chơi hoàn toàn ngẫu nhiên. Khi làm như vậy, nó nhanh chóng vượt qua mức độ chơi của con người và đánh bại phiên bản AlphaGo đã được công bố trước đó với chiến thắng 100-0.

Thuật toán học tăng cường đóng vai trò rất quan trọng trong AlphaGo Zero. Có thể nói điều đó hoàn toàn chính xác bởi vì AlphaGo Zero trở thành giáo viên của mình và đã trở thành một thiên tài cờ vây.

Khái quát về AlphaGo Zero

Cấu trúc chương trình AlphaGo Zero rất ổn định và ngắn gọn, và nó là một ứng dụng mẫu của thuật toán học tăng cường. Khi bắt đầu đào tạo, AlphaGo Zero không có bất kỳ tin hiệu giám sát nào ngoài các policy để bắt đầu đào tạo, và chỉ sử dụng trạng thái hiện tại của bàn cờ làm thông tin vào cho mạng lưới, bao gồm các đặc trưng nhân tạo khác như AlphaGo. Thêm vào đó, AlphaGo Zero sử dụng duy nhất 1 mạng nơ-ron, nó đồng thời có thể dự đoán sự phân bố xác suất, và ước lượng tỷ lệ dành chiến thắng của trạng thái hiện tại, thay vì sử dụng Stragety network và Value Network riêng như trong phiên bản trước.



Hình 9.6: Sơ đồ mạng nơron AlphaGo Zero

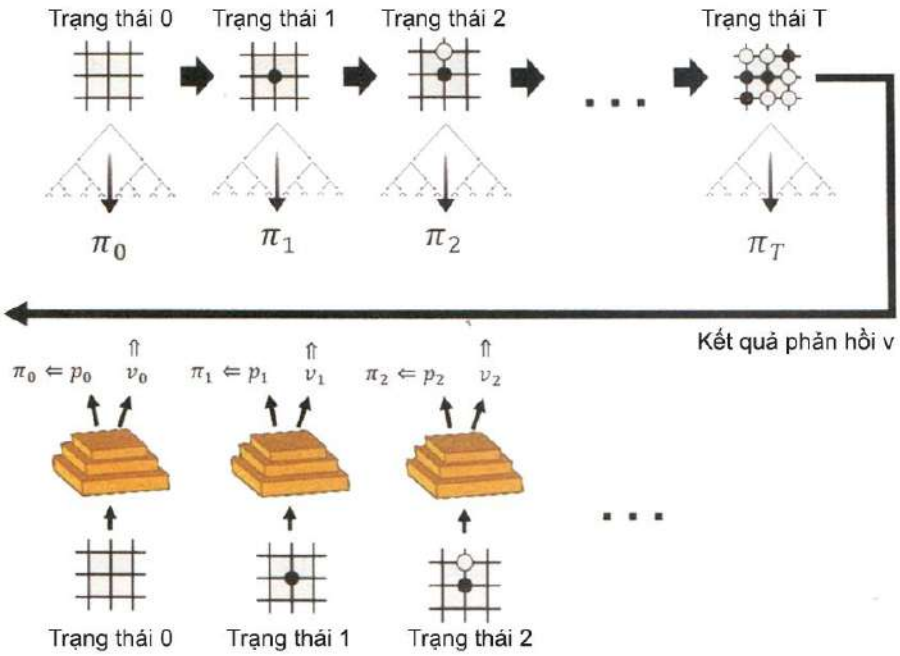
Từ quan điểm học tăng cường, AlphaGo Zero sử dụng một thuật toán lặp lại chiến lược học tăng cường để cập nhật các thông số của mạng nơ-ron. Nói một cách đơn giản, thuật toán lặp lại chiến lược học tăng cường để liên tục luân phiên đánh giá chiến lược và cải tiến chiến lược. Tiếp theo, chúng ta sẽ tìm hiểu cách AlphaGo Zero sử dụng thuật toán lặp lại chiến lược học tăng cường để hoàn thành tự cải thiện.

Huấn luyện AlphaGo Zero

Việc đào tạo AlphaGo Zero được thực hiện thông qua quá trình liên tục tự chơi. Trong mỗi trận đấu, AlphaGo Zero vẫn thực hiện thuật toán tìm kiếm cây Monte Carlo để loại trừ khi đối mặt với từng tình huống trong trò chơi. Giống như AlphaGo, AlphaGo Zero cũng dựa trên phản hồi sự kỳ vọng dự kiến của mỗi hành động trong tình huống hiện tại. Nhưng điểm khác biệt là AlphaGo Zero có thể cùng lúc dự đoán xác suất chiến thắng v và phân bố xác suất thả cờ p trong trạng thái hiện tại của bàn cờ thông qua chỉ một mạng nơ-ron, qua đó cập nhật sự kỳ vọng dự kiến của mỗi hành động tương ứng. Vì vậy, nó không cần mô phỏng các trò chơi từ trạng thái hiện tại thông qua Fast-moving network như AlphaGo nữa.

Bằng cách suy luận trò chơi từ nhiều thuật toán tìm kiếm Cây tìm kiếm Monte Carlo, AlphaGo Zero cuối cùng có thể tìm kiếm và phân bố xác suất $\boldsymbol{\pi}$ của từng vị trí trong tình huống hiện tại, AlphaGo Zero thực hiện một số lượng lớn các tìm kiếm Cây tìm kiếm Monte Carlo, sau đó đếm số lần lựa chọn của từng hành động để đạt được phân bố xác suất $\boldsymbol{\pi}$ của từng vị trí trong tình huống hiện tại. Phân bố xác suất $\boldsymbol{\pi}$ được tìm ra bởi thuật toán Cây tìm kiếm Monte Carlo thường tốt hơn (hay hơn) phân bố xác suất p được tìm bởi mạng nơ-ron, tức là $\boldsymbol{\pi}$ có thể là giá trị đích của p, vì vậy thuật toán Cây tìm kiếm Monte Carlo thực sự là một quá trình cải tiến trong việc đào tạo các chiến lược của AlphaGo Zero.

AlphaGo Zero sử dụng chiến lược được cải tiến $\boldsymbol{\pi}$ dựa trên thuật toán tìm kiếm cây Monte Carlo trong quá trình tự chơi. Thống kê kết quả thắng thua sau khi trò chơi kết thúc, và sử dụng nó làm tiêu chuẩn cho policy evaluation trong thuật toán lặp policy, được sử dụng để quay lại và cập nhật các thông số của mạng nơ-ron.



Hình 9-7: Sơ đồ huấn luyện tự học của AlphaGo Zero

Khi quay lại và cập nhật các thông số của mạng nơ-ron, AlphaGo Zero sẽ làm cho phân bố xác suất p của mạng nơ-ron xấp xỉ với phân bố xác suất $\boldsymbol{\pi}$ của thuật toán cây tìm kiếm Monte Carlo, từ đó kết quả dự đoán của mạng nơ-ron sẽ gần như là kết quả của trò chơi.

Có thể nói AlphaGo Zero hoàn toàn sử dụng thuật toán học tăng cường làm cốt lõi để cải thiện mình bằng cách tự chơi với chính nó, và đã chứng minh rằng AlphaGo với cấu trúc phức tạp hơn và sử dụng các mẫu của con người để học tập có giám sát đã bị đánh bại bởi AlphaGo Zero mà không cần sử dụng các mẫu đào tạo từ con người. AlphaGo Zero cũng một lần nữa cho chúng ta thấy sức mạnh của thuật toán học tăng cường, cho chúng ta biết rằng trong trường hợp không có kiến thức trước của con người, máy cũng có thể đánh bại con người trong thử thách khó như cờ vây.

9.4. Tóm tắt chương

Trong chương này, chúng ta đã được giới thiệu các nguyên tắc cơ bản đằng sau chương trình trí tuệ nhân tạo AlphaGo. Và biết được rằng AlphaGo bao gồm bốn phần: Mạng chiến lược - Stragety Network, Mạng giá trị - Value Network, Fast-moving Network và thuật toán Cây tìm kiếm Monte Carlo. Với sự ra đời của nguyên tắc AlphaGo, chúng ta đã giới thiệu về thuật toán học tăng cường.

Học tập tăng cường là cốt lõi của toàn thể để đưa ra quyết định, nó cho phép máy tính nâng cao bản thân thông qua việc tự học hoàn toàn như con người với toàn bộ khả năng của trí tuệ nhân tạo. Chúng ta cũng đã được giới thiệu các yếu tố quan trọng của việc học tập tăng cường và quá trình đào tạo cơ bản cho việc học tăng cường. Bằng cách ứng dụng học tăng cường trong AlphaGo, chúng ta đã thấy rằng việc học tăng cường cho phép AlphaGo chơi với chính mình, do đó làm cho mạng chiến lược mạnh mẽ hơn, học tăng cường cũng giúp AlphaGo có khả năng dự đoán hiệu quả thông qua value network. Vì vậy, trò chơi có thể được chơi rất nhiều lần mà vẫn nắm bắt được sự phát triển của trò chơi.

Cuối cùng, chúng ta đã được tìm hiểu các nguyên tắc cơ bản của AlphaGo Zero phiên bản mạnh nhất của chương trình trí tuệ nhân tạo cờ vây, và sự khác biệt giữa nó và AlphaGo đó là AlphaGo Zero sử dụng một cấu trúc học tập ngắn gọn và chuyên sâu hơn, AlphaGo Zero tự học hoàn toàn, và trở thành một bậc thầy cờ vây thông qua quá trình tự chơi.

Mặc dù chúng ta đã thấy vai trò tuyệt vời của việc học tăng cường trong chương trình trí tuệ nhân tạo cờ vây, nhưng chúng ta cần phải hiểu rằng, chương này giới thiệu về học tăng cường chỉ là phần đỉnh của “tảng băng trôi”. Như chúng ta đã nói, học tăng cường là cốt lõi chung của trí tuệ nhân tạo và nó còn nhiều khả năng hơn mà chúng ta cần khám phá trong tương lai.