

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

NGUYỄN QUANG TRUNG

**HƯỚNG TIẾP CẬN DỰA TRÊN PHỒ TẦN SỐ
CHO BÀI TOÁN NHẬN THỨC TIẾNG NÓI**

LUẬN ÁN TIẾN SĨ CÔNG NGHỆ THÔNG TIN

Hà Nội - 2019

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN QUANG TRUNG

HƯỚNG TIẾP CẬN DỰA TRÊN PHỒ TÀN SỐ
CHO BÀI TOÁN NHẬN THỨC TIẾNG NÓI

Chuyên ngành: Khoa học máy tính

Mã số: 9480101.01

LUẬN ÁN TIẾN SĨ: CÔNG NGHỆ THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC:

1.PGS. TS. Bùi Thế Duy

Hà Nội - 2019

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu do tôi thực hiện dưới sự hướng dẫn của PGS., TS. Bùi Thế Duy tại bộ môn Khoa học máy tính, Khoa Công nghệ Thông tin, Trường Đại học Công nghệ, Đại học Quốc gia Hà nội. Các số liệu và kết quả trình bày trong luận án là trung thực, chưa được công bố bởi bất kỳ tác giả nào hay ở bất kỳ công trình nào khác.

Tác giả

Nguyễn Quang Trung

LỜI CẢM ƠN

Kết quả đạt được của Luận án không chỉ là những nỗ lực cá nhân, mà còn có sự hỗ trợ và giúp đỡ của tập thể người hướng dẫn, cơ sở đào tạo, cơ quan chủ quản, đồng nghiệp và gia đình.

Trước tiên, tôi xin bày tỏ sự biết ơn sâu sắc đến PGS.TS. Bùi Thế Duy. Được làm việc với thầy là một cơ hội lớn cho tôi học hỏi phương pháp nghiên cứu, tính kiên trì và phương pháp làm việc nghiêm túc, khoa học.

Tôi xin trân trọng cảm ơn Khoa Công nghệ thông tin, Phòng Đào tạo, Ban Giám hiệu trường đại học công nghệ, đại học Quốc gia Hà Nội đã tạo điều kiện thuận lợi cho tôi trong suốt quá trình thực hiện luận án.

Tôi xin cảm ơn Ban Giám đốc Học viện Thanh thiếu niên Việt Nam và các bạn bè, đồng nghiệp đã cỗ vũ, động viên và tạo các điều kiện thuận lợi nhất cho tôi trong quá trình học tập, nghiên cứu.

Tôi cũng bày tỏ lời cảm ơn sâu sắc tới sự hỗ trợ của đề tài “Nghiên cứu ứng dụng công nghệ đa phương tiện trong bảo tồn và phát huy di sản văn hóa phi vật thể”, mã số “ĐTDL-CN.34/16” cũng như sự giúp đỡ nhiệt tình của các thành viên tham gia đề tài.

Cuối cùng, tôi xin bày tỏ lòng biết ơn đối với gia đình tôi luôn bên cạnh ủng hộ, giúp đỡ, chia sẻ với tôi những lúc khó khăn.

Xin chân thành cảm ơn!

MỤC LỤC

LỜI CAM ĐOAN	1
LỜI CẢM ƠN	2
MỞ ĐẦU	14
1. Tính cấp thiết của đề tài.....	14
2. Mục tiêu, phạm vi nghiên cứu của luận án.....	15
3. Phương pháp và nội dung nghiên cứu	16
4. Kết quả đạt được của luận án.....	17
5. Cấu trúc luận án	18
Chương 1. TỔNG QUAN VỀ NHẬN THỨC TIẾNG NÓI	19
1.1. Giới thiệu	19
1.2. Quá trình nhận thức tiếng nói ở người.....	20
1.2.1. Tai ngoài thu nhận tín hiệu tiếng nói từ	20
1.2.2. Tai giữa.....	20
1.2.3. Tai trong và cơ chế truyền sóng âm trong ốc tai	20
1.3. Quá trình mô phỏng nhận thức tiếng nói trên máy tính.....	23
1.3.1. Lấy mẫu tín hiệu tiếng nói.....	24
1.3.2. Lượng tử hóa các mẫu	25
1.3.3. Mã hóa các mẫu lượng tử hóa	25
1.3.4. Biểu diễn tín hiệu tiếng nói.	25
1.3.5. Trích chọn đặc trưng tiếng nói	27
1.3.6. Phân lớp, phân cụm dữ liệu	27
1.4. Tổng quan tình hình nghiên cứu về nhận thức tiếng nói	28
1.5. Bài toán nhận thức tiếng nói trong khoa học máy tính.....	33
1.5.1. Bài toán nhận dạng người nói.....	33
1.5.2. Bài toán nhận dạng tiếng nói	34
1.5.3. Bài toán nhận thức tiếng nói.....	35

1.6.	Một số khó khăn trong nhận thức tiếng nói	36
1.6.1.	Tính tuyến tính.....	36
1.6.2.	Phân đoạn tiếng nói	36
1.6.3.	Vần đề phụ thuộc người nói	36
1.6.4.	Vần đề nhiễu.....	36
1.6.5.	Đơn vị nhận thức cơ bản.....	37
1.7.	Mô hình nhận thức tiếng nói dựa trên học quan hệ giữa tín hiệu tiếng nói với các tín hiệu khác	37
Chương 2.	MỘT SỐ HƯỚNG TIẾP CẬN HỌC MÁY CHO BÀI TOÁN NHẬN THỨC TIẾNG NÓI.....	39
2.1.	Giới thiệu	39
2.2.	Một số mô hình học máy cho bài toán nhận thức tiếng nói... 39	
2.2.1.	Mô hình Markov ẩn	39
2.2.2.	Mô hình ngôn ngữ	41
2.2.3.	Mô hình mạng nơ-ron.....	43
2.2.4.	Mạng học sâu.....	45
2.3.	Trích chọn đặc trưng tiếng nói cho các mô hình học máy..... 54	
2.3.1.	Đặc trưng MFCC	54
2.3.2.	Phương pháp mã dự đoán tuyến tính LPC	56
2.3.3.	Đặc trưng PLP	58
2.4.	Kết luận	60
Chương 3.	HƯỚNG TIẾP CẬN DỰA TRÊN PHỔ TẦN SỐ CHO BÀI TOÁN NHẬN THỨC TIẾNG NÓI TRONG MÔI LIÊN HỆ VỚI CÁC KHÁI NIỆM	61
3.1.	Giới thiệu	61
3.2.	Phổ tần số của tín hiệu tiếng nói	62
3.3.	Đặc trưng bất biến SIFT	64
3.4.	Phương pháp phân lớp NBNN.....	68

3.5.	Phương pháp phân lớp LNBNN	70
3.6.	Hướng tiếp cận trích chọn đặc trưng tiếng nói dựa trên phô tần số cho bài toán nhận thức tiếng nói.....	72
3.7.	Hướng tiếp cận mạng tích chập dựa trên phô tần số cho bài toán nhận thức tiếng nói	75
3.8.	Thực nghiệm và kết quả.....	75
3.8.1.	Dữ liệu thực nghiệm	76
3.8.2.	Thí nghiệm so sánh độ chính xác phân lớp của đặc trưng SIFT với đặc trưng MFCC khi sử dụng LNBNN	76
3.8.3.	Thí nghiệm với dữ liệu co dãn theo thời gian	79
3.8.4.	Thí nghiệm so sánh LNBNN và các phương pháp phân lớp khác	80
3.8.5.	Thí nghiệm khả năng học tăng cường của LNBNN.....	81
3.8.6.	Thí nghiệm với mạng tích chập trên tín hiệu tiếng nói	82
3.9.	Kết luận.....	84
Chương 4. MÔ HÌNH NHẬN THỨC TIẾNG NÓI THÔNG QUA HỌC MỐI QUAN HỆ GIỮA TÍN HIỆU TIẾNG NÓI VÀ HÌNH ẢNH		86
4.1.	Giới thiệu	86
4.2.	Các phương pháp học mối quan hệ.....	87
4.2.1.	Học mối quan hệ bằng mạng nhân tạo	87
4.2.2.	Học mối quan hệ bằng HMM.....	90
4.2.3.	Học mối quan hệ dựa trên luật	91
4.2.4.	Học mối quan hệ dựa trên thống kê.....	91
4.3.	Đề xuất mô hình nhận thức tiếng nói.....	93
4.3.1.	Cơ sở đề xuất mô hình.....	93
4.3.2.	Mô hình nhận thức tiếng nói dựa trên học quan hệ giữa tín hiệu âm thanh và tín hiệu hình ảnh	96

4.3.3. Mô hình nhận thức tiếng nói dựa trên ánh xạ giữa tín hiệu âm thanh và tín hiệu hình ảnh bằng mạng tích chập.....	99
4.4. Thực nghiệm và kết quả.....	100
4.4.1. Thực nghiệm mô hình nhận thức tiếng nói dựa trên học quan hệ giữa tín hiệu âm thanh và tín hiệu hình ảnh	100
4.4.2. Thực nghiệm mô hình nhận thức dựa trên mạng tích chập	
102	
4.5. Kết luận	106
Chương 5. MỘT SỐ CÁI TIẾN CHO BÀI TOÁN NHẬN THỨC TIẾNG NÓI DỮ LIỆU LỚN	108
5.1. Giới thiệu	108
5.2. Rút gọn đặc trưng.....	109
5.2.1. Giới thiệu về rút gọn đặc trưng	109
5.2.2. Rút gọn đặc trưng SIFT	110
5.2.3. Bảng băm đa chỉ số.....	113
5.2.4. Thực nghiệm và kết quả	115
5.3. Cài đặt phương pháp phân lớp LNBNN cho bài toán nhận thức tiếng nói dữ liệu lớn	116
5.3.1. Giới thiệu Framework Hadoop.....	116
5.3.2. Cài đặt thuật toán phân lớp LNBNN trên nền Hadoop ..	117
5.3.3. Thực nghiệm.....	121
5.4. Kết luận	124

DANH MỤC KÝ HIỆU VÀ TỪ VIẾT TẮT

TT	Viết tắt	Từ tiếng Anh	Nghĩa tiếng Việt
1.	ANN	Artificial Neural Network	Mạng trí tuệ nhân tạo
2.	BAM	Bi-directional Assosiation Memory	Mạng nhớ kết hợp hai chiều
3.	CNN	Convolution Neural Network	Mạng tích chập
4.	CFG	Context Free Grammar	Văn phạm phi ngữ cảnh
5.	CSLU	Center for Spoken Language Understanding	Trung tâm nghiên cứu tiếng nói
6.	DNN	Deep Neural Network	Mạng học sâu
7.	DoG	Different-of-Gaussian	Bộ lọc DoG
8.	DCT	Discrete Cosin Transform	Biến đổi Cosin rời rạc
9.	DFT	Discrete Fourier Transform	Biến đổi Fourier rời rạc
10.	DTW	Dynamic Time Warping	Phương pháp lập trình động
11.	FA	Factor Analysis	Phân tích nhân tố
12.	FFT	Fast Fourier Transform	Biến đổi Fuutier nhanh
13.	GMM	Gaussian Mixture Model	Mô hình Gaussian hỗn hợp
14.	HDFS	Hadoop Distributed File System	Hệ thống tệp phân tán
15.	HMM	Hidden Markov Model	Mô hình Markov ẩn
16.	HOG	Histogram of Oriented Gradients	Đặc trưng lược đồ độ dốc theo hướng
17.	ICA	Independent Component Analysis	Phân tích thành phần độc lập
18.	LBG	Linde–Buzo–Gray	Thuật toán LBG
19.	LDA	Linear Discriminant Analysis	Phân tích biệt thức tuyến tính
20.	LNBNN	Local Naïve Bayes Nearest Neighbor	Phương pháp phân lớp NBNN cục bộ
21.	LPC	Linear Predictive Coding	Mã dự báo tuyến tính
22.	MFCC	Mel-frequency cepstral coefficients	Hệ số Mel
23.	MPCA	Multiple Principal Component Analysis	Phân tích đa thành phần

24.	NBNN	Naïve Bayes Nearest Neighbor	Phương pháp phân lớp NBNN
25.	PCA	Principal Component Analysis	Phân tích thành phần chính
26.	PLP	Perceptual Linear Prediction	Mã nhận thức tuyến tính
27.	RNN	Recurrent Neural Network	Mạng hồi quy
28.	SIFT	Scale Invariant Feature Transform	Đặc trưng bất biến đối với phép biến đổi
29.	SOM	Self Organizing Map	Bản đồ tự tổ chức
30.	SURF	Speeded Up Robust Features	Đặc trưng ảnh nhanh
31.	SVM	Support Vector Machine	Máy véc tơ hỗ trợ
32.	VOT	Voice On Set time	Thời gian bắt đầu nguyên âm

DANH MỤC HÌNH ẢNH

Hình 1.1 Sơ đồ quá trình nhận thức tiếng nói.....	19
Hình 1. 2 Mô phỏng các bước trong nhận thức tiếng nói của máy tính	19
Hình 1. 3 Quá trình thu nhận âm thanh ở óc tai	21
Hình 1. 4 Cộng hưởng với các tần số âm khác nhau ở óc tai	22
Hình 1.5 Khu vực lưu trữ đặc trưng tiếng nói trên vỏ não	23
Hình 1. 6 Biểu diễn tín hiệu tiếng nói trên miền thời gian	26
Hình 1. 7 Biểu diễn tín hiệu tiếng nói trên miền tần số.....	27
Hình 1.8 Biểu diễn tín hiệu tiếng nói trên miền kết hợp	27
Hình 2. 1 Mô hình HMM-GMM có cấu trúc dạng Left-Right liên kết không đầy đủ	40
Hình 2. 2 Mạng Perceptron. (a) Perceptron 1 lớp, (b) Perceptron nhiều lớp.....	44
Hình 2. 3 Mô hình bộ tự mã hóa.....	47
Hình 2. 4 Mô hình mạng hồi quy.....	48
Hình 2. 5 Mô hình mạng tích chập CNN.....	49
Hình 2. 6 Tích chập một bộ lọc với dữ liệu đầu vào	50
Hình 2. 7 Ví dụ lấy mẫu với hàm max.....	51
Hình 2. 8 Mô hình mạng tích chập LeNet 5 [Lecun, 1998]	52
Hình 2. 9 Mô hình mạng tích chập AlexNet [Krizhevsky, 2012]	52
Hình 2. 10 Mô hình mạng ZF Net [Zeiler, 2014]	53
Hình 2. 11 Mô hình mạng tích chập VGGNET [Simonyan, 2014]	53
Hình 2. 12 Sơ đồ khôi các bước trích chọn đặc trưng MFCC	54
Hình 2. 13 Sơ đồ trích chọn đặc trưng LPC	57
Hình 2. 14 Sơ đồ khôi các bước trích chọn đặc trưng PLP	59
Hình 3. 1 Phổ của từ A trong tiếng Anh được nói bởi 4 người khác nhau	62
Hình 3. 2 Phổ của các chữ cái A-D trong tiếng Anh của cùng một người nói.....	63
Hình 3. 3 Phổ của âm tiết Haa trong tiếng Nhật được nói bởi 5 người khác nhau.....	63

Hình 3. 4 Phổ của 5 âm tiết tiếng Nhật do cùng một người nói	63
Hình 3. 5 Sơ đồ trích xuất phổ tần số của tín hiệu tiếng nói	64
Hình 3. 6 Mô tả điểm hấp dẫn SIFT [Lowe, 1999]	66
Hình 3. 7 Sơ đồ các bước trích chọn đặc trưng SIFT-SPEECH từ tín hiệu tiếng nói	67
Hình 3. 8 Một số điểm SIFT-SPEECH trích xuất từ phổ tần số của tín hiệu tiếng nói	67
Hình 3. 9 Mô hình phân lớp tiếng nói bằng LNBNN-SIFT-SPEECH.	72
Hình 3. 10 Mô hình CNN cho bài toán nhận dạng tiếng nói dựa trên phổ tần số.....	75
Hình 3. 11 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên dữ liệu số English Digits.....	77
Hình 3. 12 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên dữ liệu ISOLET.	78
Hình 3.13 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên 20 lớp đầu tiên của dữ liệu TMW	78
Hình 3.14 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên dữ liệu JVPD	78
Hình 3.15 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên dữ liệu số tiếng Việt.....	79
 Hình 4. 1 Mô hình mạng Hopfield [Raul, 1996]	88
Hình 4. 2 Mô hình mạng BAM [Kosko, 1987]	89
Hình 4. 3 Mô hình mạng tự tổ chức [Kohonen, 1982]	90
Hình 4. 4 Mô hình HMM [Baum, 1966]	91
Hình 4. 5 Ví dụ các luật theo văn phạm phi ngữ cảnh	92
Hình 4. 6 Sơ đồ các vùng vỏ não sơ cấp và vùng vỏ não liên kết.....	93
Hình 4. 7 Ví dụ minh họa tập dữ liệu thực nghiệm DIGITS	94
Hình 4. 8 Mô hình nhận thức tiếng nói cho người máy	95
Hình 4. 9 Mô hình học ánh xạ giữa tiếng nói và hình ảnh bằng mạng CNN.....	100
Hình 4. 10 Độ chính xác của mô hình trên bộ dữ liệu DIGITS	101
Hình 4. 11 Độ chính xác của mô hình trên bộ dữ liệu OBJECTS.....	101

Hình 4. 12 Hai mươi mẫu huấn luyện của 8 lớp trong bộ dữ liệu COIL	102
Hình 4. 13 Hai mươi mẫu huấn luyện của bộ dữ liệu FNT từ A đến Z	103
Hình 4. 14 Hai mươi mẫu huấn luyện chữ số viết tay trong MNIST.	103	
Hình 4. 15 Hai mươi mẫu hình ảnh do mô hình sinh ra của bộ dữ liệu COIL	104
Hình 4. 16 Hai mươi mẫu hình ảnh do mô hình sinh ra của bộ dữ liệu MNIST	104
Hình 4. 17 Hai mươi mẫu hình ảnh kết quả do mô hình sinh ra đối với bộ dữ liệu FNT	106
Hình 5. 1 a. Lược đồ giá trị các thành phần của điểm đặc trưng SIFT, b. Medians của các thành phần của SIFT trên dữ liệu ISOLET	110
Hình 5. 2 a. Lược đồ giá trị của các thành phần của SIFT trên cơ sở dữ liệu Digits, b. Medians của các thành phần của SIFT trên dữ liệu Digits	111
Hình 5. 3 Lược đồ giá trị các thành phần của đặc trưng SIFT trên dữ liệu PLACES, b. Median của SIFT trên dữ liệu PLACES	111
Hình 5. 4 a. Lược đồ giá trị các thành phần của SIFT trên dữ liệu JVPD, b. Trung vị của các thành phần của SIFT trên dữ liệu JVPD	112
Hình 5. 5 Lược đồ giá trị các thành phần của SIFT trên dữ liệu TMW, b. Medians của các thành phần của SIFT trên dữ liệu TMW	112
Hình 5. 6 Mô hình cụm máy tính thực nghiệm	122

DANH MỤC BẢNG

Bảng 3. 1 So sánh độ chính xác phân lớp của LNBNN với SIFT và MFCC	77
Bảng 3. 2 So sánh kết quả đối với dữ liệu bị co dãn một chiều	79
Bảng 3. 3 So sánh độ chính xác của các phương pháp phân lớp với đặc trưng MFCC	80
Bảng 3. 4 So sánh độ chính xác của các phương pháp phân lớp với đặc trưng SIFT	80
Bảng 3. 5 So sánh độ chính xác phân lớp khi bổ sung thêm dữ liệu huấn luyện cho tất cả các lớp	81
Bảng 3. 6 So sánh độ chính xác phân lớp khi bổ sung thêm lớp (tri thức) cho mô hình	82
Bảng 3. 7 So sánh độ chính xác phân lớp của CNN và LNBNN kết hợp với SIFT trên phổ tần số của tín hiệu tiếng nói.....	83
 Bảng 4. 1 Kết quả phân lớp trung bình hình ảnh do mô hình nhận thức tiếng nói sinh ra bằng mạng tích chập.....	105
 Bảng 5. 1 So sánh độ chính xác phân lớp trên các bộ dữ liệu	115
Bảng 5. 2 So sánh thời gian chạy trên các dữ liệu khác nhau (giây)..	115
Bảng 5. 3 So sánh độ phân lớp chính xác trên các dữ liệu thực nghiệm	123
Bảng 5. 4 So sánh thời gian truy vấn trung bình một đặc trưng trên các dữ liệu khác nhau (tính bằng giây).....	123

DANH MỤC THUẬT TOÁN

Thuật toán 3. 1 Thuật toán phân lớp NBNN	70
Thuật toán 3. 2 Thuật toán LNBNN	71
Thuật toán 3. 3 Thuật toán LNBNN-SIFT-SPEECH	73
Thuật toán 4. 1 Thuật toán học mồi quan hệ RELATION- Pha huấn luyện	98
Thuật toán 4. 2 Thuật toán học mồi quan hệ RELATION - Pha phân lớp	99
Thuật toán 5. 1 Thuật toán rút gọn đặc trưng SIFT_REDUCE.....	113
Thuật toán 5. 2 Thuật toán xây dựng bảng băm đa chỉ số MIH.....	114
Thuật toán 5. 3 Thuật toán tìm kiếm K hàng xóm gần nhất MIH_KNN	114
Thuật toán 5. 4 Thuật toán LNBNN-HADOOP-SETUP	119
Thuật toán 5. 5 Thuật toán LNBNN-HADOOP-MAP	119
Thuật toán 5. 6 thuật toán LNBNN-HADOOP-REDUCE	120
Thuật toán 5. 7 Thuật toán LNBNN-HADOOP-CLEANUP	121

MỞ ĐẦU

1. Tính cấp thiết của đề tài

Ngày nay, với sự bùng nổ của xã hội thông tin, con người không còn chỉ có nhu cầu giao tiếp với nhau nữa mà còn cần giao tiếp với những thiết bị điện tử. Hình thức giao tiếp người - máy thông qua ngôn ngữ tự nhiên sẽ đem lại nhiều ứng dụng, góp phần giải phóng sức lao động của con người. Chính vì vậy, việc làm cho máy tính có thể nhận thức được tiếng nói (hiểu tiếng nói) có tầm quan trọng đặc biệt liên quan đến quá trình phát triển của văn minh nhân loại. Nhận thức âm thanh nói chung hay nhận thức tiếng nói riêng đã được nghiên cứu từ đầu những năm 1950. Tuy nhiên, những nghiên cứu về nhận thức tiếng nói ở thời kỳ đầu chỉ tập trung vào một số bài toán cụ thể như bài toán tách nguồn tiếng nói, bài toán nhận dạng tiếng nói, bài toán nhận dạng hay xác thực người nói.

Gần đây, nghiên cứu về nhận thức tiếng nói đã đạt được nhiều thành tựu to lớn. Tuy nhiên, các nghiên cứu về nhận thức tiếng nói chỉ xây dựng các hệ thống có thể hiểu ở mức độ phân biệt được tiếng nói ở một khía cạnh nào đó như hệ thống có thể phân biệt được các nguồn tiếng nói khác nhau từ một nguồn tổng hợp các tín hiệu tiếng nói [Allen, 2004] , hay phân biệt tiếng nói từ nguồn có nhiều, hay bài toán phân biệt được nguyên âm với phụ âm [Hillenbrand, 1995] [Hillenbrand, 2001] [Krisztina, 2005] [Lengeris, 2014] , phân biệt được các âm tiết, nhận dạng được các từ độc lập [McClelland, 1986] [Bever, 1969] [Luce, 1998] , hay thậm chí là nhận dạng tiếng nói liên tục [Davis, 1980] [Fowler, 1995] . Nghĩa là, các nghiên cứu này chỉ tập trung mô phỏng hoạt động nhận thức tiếng nói xảy ra ở vùng vỏ não thính giác đặc biệt là vùng vỏ não thính giác sơ cấp nơi lưu trữ các đặc trưng về tần số của tiếng nói và vùng vỏ não thính giác thứ cấp nơi chứa các mẫu âm thanh có mối liên hệ với nhau. Rất ít nghiên cứu đặt bài toán nhận thức tiếng nói trong mối quan hệ với nhận thức của các hệ giác quan khác như thị giác, khứu giác, xúc giác.

Nói cách khác, các nghiên cứu về nhận thức tiếng nói đến nay chủ yếu là nghiên cứu mô phỏng quá trình nhận thức mối liên hệ giữa các tín hiệu âm thanh với nhau và liên kết giữa âm thanh với các từ, khái niệm định nghĩa trước. Hay nói cách khác, các nghiên cứu về nhận thức tiếng nói chủ yếu nghiên cứu

ánh xạ giữa tín hiệu âm thanh với các thành phần ngôn ngữ do tri thức con người cung cấp trước, chưa nghiên cứu nhận thức tiếng nói trong mối liên hệ giữa tín hiệu âm thanh với các tín hiệu khác đồng thời thu được bởi các giác quan không cần phải cung cấp các tri thức của con người.

Để giải quyết bài toán nhận thức tiếng nói ở khía cạnh ánh xạ giữa tín hiệu tiếng nói với các tri thức có sẵn hay còn gọi là bài toán nhận dạng tiếng nói, nhiều lý thuyết và mô hình đã được đề xuất. Các mô hình nhận thức tiếng nói kinh điển như mô hình vận động (Motor Theory) [Liberman, 1967] , Cohort [Marslen-Wilson, 1975] [Marslen-Wilson, 1987] , TRACE [McClelland, 1986] , mô hình tính toán nơ-ron [Kröger, 2009] , mô hình luồng kép [Hickok, 2000] [Hickok, 2007] .

Xuất phát từ thực tế và những lý do trên, việc lựa chọn đề tài “Hướng tiếp cận dựa trên phô tần số cho bài toán nhận thức tiếng nói” với mục tiêu nghiên cứu đề xuất mô hình mô phỏng quá trình nhận thức tiếng nói thông qua mô phỏng việc học liên kết giữa vùng vỏ não thính giác với các vùng vỏ não khác đặc biệt là liên kết giữa vùng vỏ não thính giác với vùng vỏ não thị giác.

Kết quả đề tài này có thể ứng dụng trong việc huấn luyện người máy, cải thiện cách thức huấn luyện người máy, làm quá trình huấn luyện người máy trở nên tự nhiên hơn thông qua việc trang bị cho người máy các bộ cảm biến mô phỏng các giác quan của con người.

2. Mục tiêu, phạm vi nghiên cứu của luận án

Mục tiêu chính của đề tài là xây dựng mô hình nhận thức tiếng nói dựa trên liên kết giữa tín hiệu thính giác với các thông tin, tín hiệu khác. Trong phạm vi đề tài này, chúng tôi tiến hành thực nghiệm xây dựng mô hình học mối quan hệ giữa tín hiệu thính giác với khái niệm cho trước và mô hình quan hệ giữa tín hiệu tiếng nói tín hiệu hình ảnh.

Xuất phát từ mục tiêu trên, phạm vi nghiên cứu của đề tài tập trung vào các vấn đề sau:

- Xử lý đoạn tín hiệu tiếng nói,
- Biểu diễn tín hiệu tiếng nói và trích chọn đặc trưng tiếng nói,
- Hiểu tiếng nói ở khía cạnh liên kết với từ, cụm từ định nghĩa sẵn,

- Hiểu tiếng nói ở khía cạnh liên kết với các tín hiệu khác, trong phạm vi của đề tài này, chúng tôi tiến hành thực nghiệm liên kết giữa tín hiệu tiếng nói với tín hiệu hình ảnh.

Nhiệm vụ của đề tài là:

- Cải thiện phương pháp học liên kết giữa tín hiệu tiếng nói với các từ được định nghĩa sẵn.

- Xây dựng mô hình học mối quan hệ giữa tín hiệu tiếng nói với các tín hiệu khác.

- Cải thiện tốc độ thông qua rút gọn dữ liệu đặc trưng, giảm kích thước bộ nhớ cần thiết cho mô hình.

- Cải thiện tốc độ thông qua thực hiện song song và phân tán hóa mô hình cho bài toán dữ liệu lớn.

3. Phương pháp và nội dung nghiên cứu

Phương pháp luận trong nghiên cứu của luận án là kết hợp giữa nghiên cứu lý thuyết và thực nghiệm.

Về lý thuyết, chúng tôi nghiên cứu về các lý thuyết nhận thức tiếng nói, các mô hình nhận thức tiếng nói, các mô hình tính toán cho bài toán nhận thức tiếng nói.

Về nghiên cứu thực nghiệm, chúng tôi xây dựng mô hình học máy mô phỏng bài toán nhận thức tiếng nói tiến hành thực nghiệm trên các bộ dữ liệu tiếng nói là các từ, cụm từ độc lập. Thực nghiệm mô hình mô phỏng liên kết giữa tín hiệu tiếng nói với tín hiệu hình ảnh.

Phương pháp tổng hợp tài liệu, các thông tin liên quan đến đề tài, lựa chọn các cách tiếp cận đã được áp dụng thành công ở các lĩnh vực khác hoặc trong các bài toán tương tự, tiến hành thử nghiệm với các bộ dữ liệu tiếng nói khác nhau, đánh giá kết quả, từ đó sẽ tiến hành nghiên cứu sâu hơn về giải pháp cải tiến phương pháp, hiệu chỉnh các tham số nhằm nâng cao chất lượng của mô hình để xuất đáp ứng bài toán thực tiễn.

4. Kết quả đạt được của luận án

- Đề xuất sử dụng đặc trưng SIFT-SPEECH được trích chọn từ phổ tần số của tín hiệu tiếng nói. Việc đề xuất sử dụng đặc trưng SIFT-SPEECH cho bài toán nhận thức tiếng nói là dựa trên cơ chế thu nhận đặc trưng tiếng nói của hệ thính giác ở con người.

- Đề xuất sử dụng phương pháp phân lớp LNBNN-SIFT-SPEECH cho bài toán nhận thức tiếng nói bằng cách kết hợp giữa phương pháp phân lớp LNBNN và phương pháp trích chọn đặc trưng SIFT-SPEECH trên phổ tần số của tiếng nói áp dụng cho bài toán nhận dạng tiếng nói đã thu được những kết quả tốt đối với các bộ dữ liệu thực nghiệm.

- Đề xuất mô hình mạng tích chập dựa trên phổ tần số của tiếng nói cho bài toán nhận thức tiếng nói trong môi liên hệ giữa tín hiệu tiếng nói với khái niệm được định nghĩa trước.

- Đề xuất xây dựng mô hình nhận thức tiếng nói mô phỏng việc nhận thức của con người ở vùng não liên kết, xây dựng mô hình học mối quan hệ giữa tín hiệu tiếng nói với tín hiệu hình ảnh.

- Đề xuất cải tiến hiệu năng của mô hình thông qua việc đề xuất phương pháp rút gọn dữ liệu bằng cách biểu diễn đặc trưng SIFT từ một véc tơ 128 chiều với mỗi chiều có kích thước một byte thành một véc tơ SIFT nhị phân 128 bít. Kết quả thực nghiệm cho thấy phương pháp rút gọn dữ liệu này vẫn giữ được độ chính xác của mô hình trong khi giảm kích thước lưu trữ 8 lần.

- Đề xuất cài đặt phương pháp phân lớp LNBNN-HADOOP trên nền Hadoop, một nền tảng cho bài toán xử lý dữ liệu lớn song song và phân tán. Nền tảng Hadoop, cho phép kết hợp nhiều máy tính có cấu hình thấp hơn để tạo thành một hệ thống xử lý song song, phân tán mạnh hơn, tận dụng được sức mạnh của các hệ thống máy tính hiện có.

Các kết quả nghiên cứu của luận án sẽ là những đóng góp mới về mặt lý thuyết cho lĩnh vực nhận thức tiếng nói, đồng thời có thể ứng dụng trong lĩnh vực giao tiếp người máy, chế tạo người máy. Đây cũng là bước tiền đề để phát triển mô hình nhận thức cho người máy hoàn thiện hơn, gần với quá trình nhận

thúc của con người thông qua việc trang bị các bộ cảm biến mô phỏng các cơ quan giác quan của con người, giúp nâng cao thông tin cho hệ thống người máy.

5. Cấu trúc luận án

Cấu trúc của luận án ngoài phần mở đầu có 5 chương nội dung, kết luận, danh mục tài liệu tham khảo và phụ lục.

Chương 1: Giới thiệu các khái niệm cơ bản về hệ thính giác của con người. Phần này chú trọng tới các đặc điểm có ảnh hưởng tới quá trình nhận thức của con người. Giới thiệu tổng quan về bài toán nhận thức tiếng nói, những bài toán và các hướng nghiên cứu cụ thể của bài toán nhận thức tiếng nói, các mức độ nhận thức cũng như các khó khăn trong bài toán này. Chương này cũng giới thiệu một cách khái quát các lý thuyết, mô hình cho bài toán nhận thức tiếng nói và các ứng dụng của bài toán nhận thức tiếng nói.

Chương 2: Giới thiệu các kiến thức cơ sở về nhận thức tiếng nói như các phương pháp học máy được sử dụng trong bài toán nhận thức tiếng nói, một số phương pháp trích chọn đặc trưng phổ biến được sử dụng trong các hệ thống nhận thức tiếng nói.

Chương 3: Đề xuất hai hướng tiếp cận mới cho bài toán nhận thức tiếng nói trong mối liên hệ với các khái niệm, thuật ngữ được định nghĩa trước bằng cách áp dụng phương pháp phân lớp LNBNN-SIFT-SPEECH và đề xuất mô hình tích hợp cho bài toán nhận thức tiếng nói này. Các mô hình được đánh giá thông qua thực nghiệm trên một số bộ dữ liệu cụ thể.

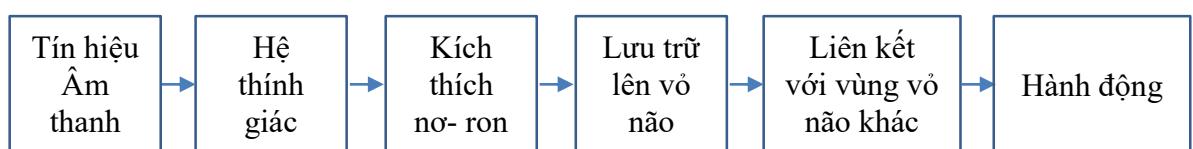
Chương 4: Đề xuất mô hình nhận thức tiếng nói dựa trên việc học mối quan hệ và mô hình học ánh xạ giữa một tín hiệu tiếng nói với một hình ảnh thu được của một sự vật, hiện tượng xảy ra cùng lúc với tín hiệu âm thanh được nghe thấy giống như quá trình học ngôn ngữ của con người.

Chương 5: Đề xuất phương pháp rút gọn đặc trưng bằng cách lượng tử hóa giá trị của các thành phần của đặc trưng SIFT về giá trị nhị phân sau đó mã hóa lại đặc trưng SIFT nhị phân thành một bộ mô tả mới, đồng thời đề xuất cài đặt phương pháp phân lớp LNBNN-HADOOP song song, phân tán trên nền tảng Hadoop cho bài toán nhận thức tiếng nói dữ liệu lớn.

Chương 1. TỔNG QUAN VỀ NHẬN THỨC TIẾNG NÓI

1.1. Giới thiệu

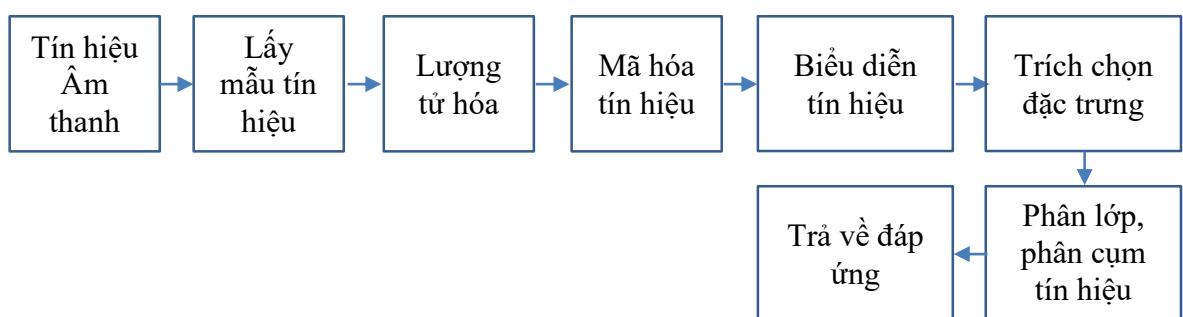
Nhận thức nói chung là việc tổ chức, xác định và diễn giải thông tin từ các giác quan để biểu diễn và hiểu môi trường xung quanh [Schacter, 2011]. Nhận thức liên quan đến các tín hiệu trong hệ thần kinh mà nó là kết quả từ sự kích thích vật lý hay hóa học của các cơ quan giác quan. Nhận thức tiếng nói là khả năng nhận biết cấu trúc ngôn ngữ trong tín hiệu âm thanh hay nhận thức tiếng nói là quá trình tín hiệu âm thanh của một ngôn ngữ được nghe, diễn dịch để hiểu ngôn ngữ.



Hình 1.1 Sơ đồ quá trình nhận thức tiếng nói

Từ sơ đồ quá trình nhận thức, tín hiệu âm thanh được thu nhận thông qua hệ thính giác, khi tín hiệu đủ mạnh sẽ làm kích thích các nơ-ron thần kinh làm kích hoạt một số nơ-ron trên vùng vỏ não. Đồng thời, cùng với các tín hiệu thu được từ hệ thính giác khác vỏ não sẽ tạo nên các liên kết giữa vùng vỏ não của vùng não thính giác với các vùng não khác để lưu trữ các thông tin bậc cao, thông tin ở mức trừu tượng về sự vật hiện tượng và có phản ứng phù hợp với tín hiệu thu được.

Trong khoa học máy tính, để máy tính có thể nhận thức được tiếng nói các nhà nghiên cứu đã cố gắng mô phỏng, giải thích cơ chế hoạt động nhận thức tiếng nói của con người. Chúng tôi cho rằng, quá trình mô phỏng nhận thức tiếng nói trong máy tính cơ bản có những bước sau:



Hình 1.2 Mô phỏng các bước trong nhận thức tiếng nói của máy tính

Trong phần 1.3 của chương này sẽ giải thích sơ lược các bước trong quá trình nhận thức tiếng nói ở người, và phần 1.4 sẽ giải thích các bước trong mô hình mô phỏng nhận thức tiếng nói trên máy tính.

1.2. Quá trình nhận thức tiếng nói ở người

Quá trình nhận thức tiếng nói được bắt đầu từ việc thu nhận tín hiệu âm thanh trải qua một số giai đoạn sau:

1.2.1. Tai ngoài thu nhận tín hiệu tiếng nói từ

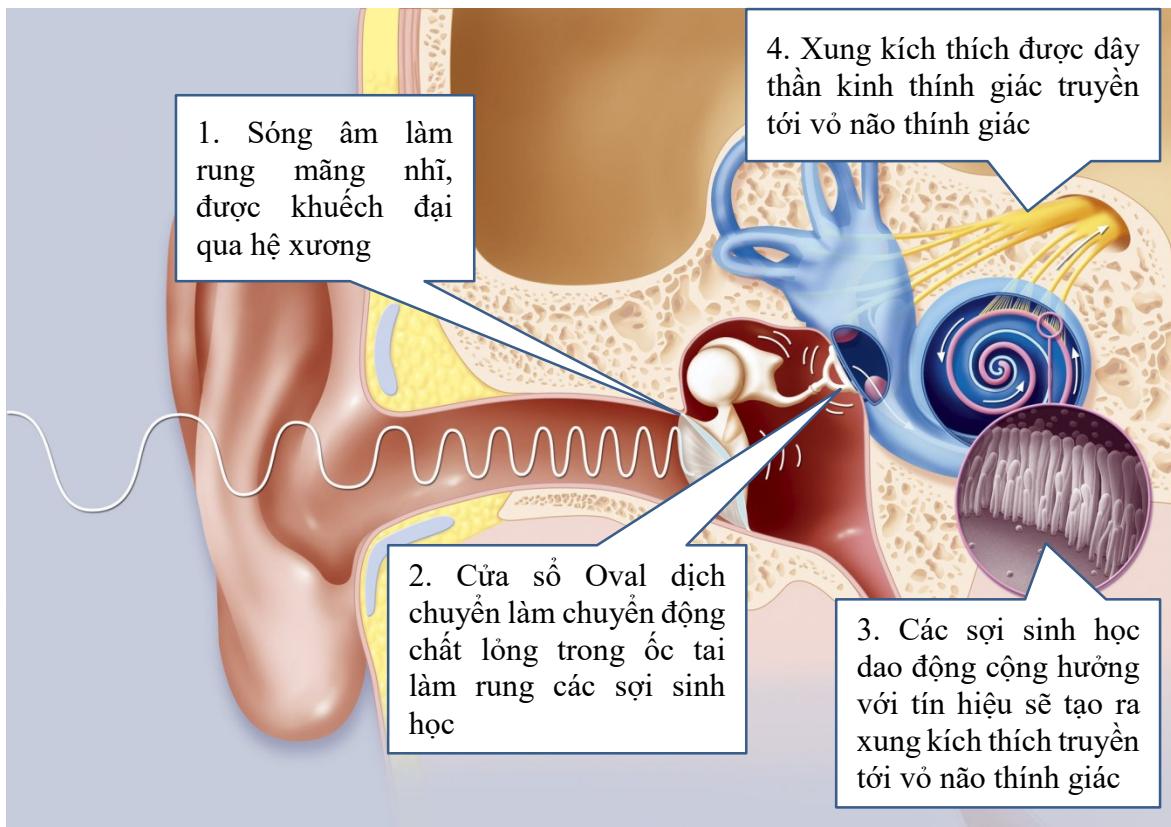
Tai ngoài được cấu tạo bởi vành tai và ống tai ngoài. Vành tai là một bộ phận có chức năng thu nhận âm thanh. Ống tai ngoài có tác dụng khuếch đại các âm thanh ở tần số âm thanh từ 2,5kHz đến 3,5kHz [Menezes, 2004]. Vành tai có tác dụng thu thập và tập trung tín hiệu âm thanh để truyền tải vào tai giữa và tai trong.

1.2.2. Tai giữa

Tai giữa được ngăn cách với tai ngoài bởi màng nhĩ. Màng nhĩ cực kỳ đàn hồi và là bộ phận chính tiếp nhận sóng âm để tạo ra các rung động tương ứng. Khi âm thanh đi vào trong ống tai, nó sẽ làm rung động màng nhĩ. Màng nhĩ có thể dễ dàng tiếp nhận sóng âm dù âm thanh được truyền đến từ bất cứ vị trí nào trên màng nhĩ. Khi tiếp nhận được tín hiệu âm thanh, màng nhĩ sẽ dao động và làm dịch chuyển hệ thống khuếch đại âm thanh thông qua cấu trúc liên kết của ba hệ xương là xương búa, xương đe và xương bàn đạp. Tín hiệu âm thanh sau khi được khuếch đại sẽ được truyền vào tai trong.

1.2.3. Tai trong và cơ chế truyền sóng âm trong óc tai

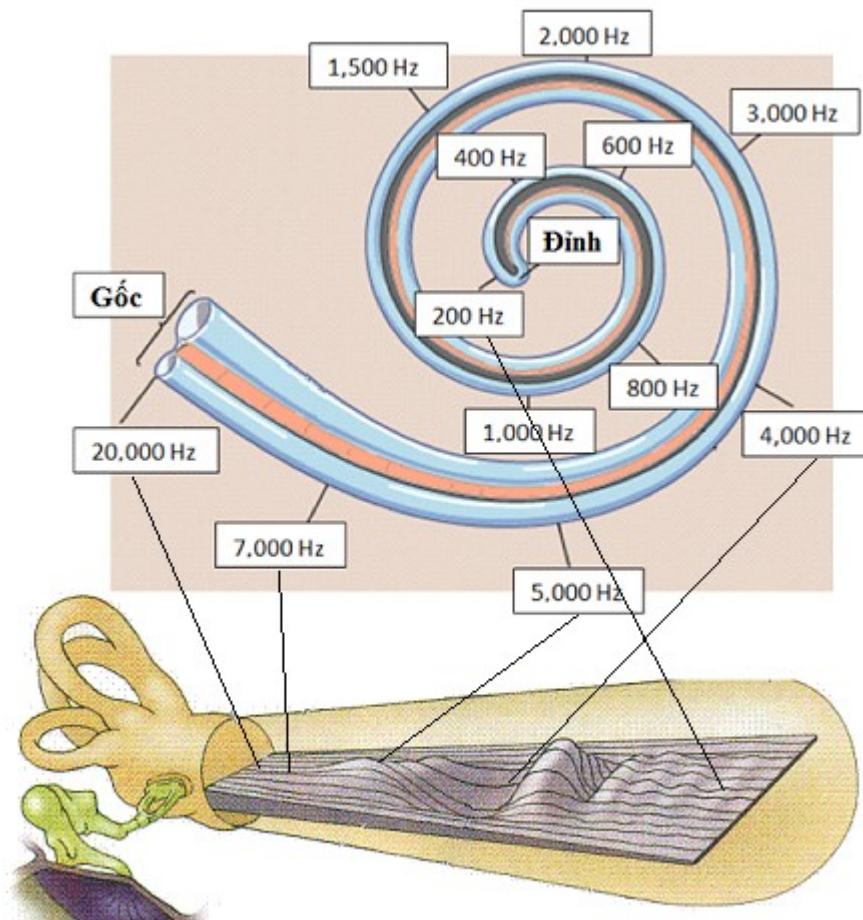
Tai trong gồm bộ phận tiền đình và óc tai. Óc tai là bộ phận phức tạp nhất của hệ thống thính giác. Óc tai có nhiệm vụ là sử dụng các dao động vật lý của sóng âm để chuyển hóa thành các tín hiệu mà bộ não hiểu được. Cấu tạo óc tai gồm ba ống đặt kề nhau ngăn cách bởi các màng mẫn cảm, các ống này co lại thành hình xoắn như trôn óc.



Hình 1. 3 Quá trình thu nhận âm thanh ở óc tai

Màng đáy, là một bề mặt cứng dàn trải toàn bộ chiều dài của ốc tai có chức năng tiếp nhận sóng âm thanh truyền từ bên ngoài đến đầu con lợn của ốc tai. Màng đáy được cấu tạo bởi khoảng 15.500¹ sợi sinh học dàn trải trên toàn bộ kích thước ốc tai. Các sợi này có cấu tạo khác nhau để cộng hưởng với các tần số khác nhau của sóng âm [Guenter, 1978] [Purves, 2001]. Khi một tần số sóng âm cộng hưởng với các sợi sinh học này ở một điểm nào đó, làm chúng dao động liên tục dẫn đến năng lượng của sóng âm sẽ được giải phóng. Các tín hiệu âm thanh với tần số cao sẽ làm dao động các sợi sinh học ở gần gốc trong khi các tín hiệu âm với tần số thấp sẽ làm dao động các sợi ở phần đỉnh của ốc tai.

¹ <http://www.cochlea.eu/en/hair-cells>

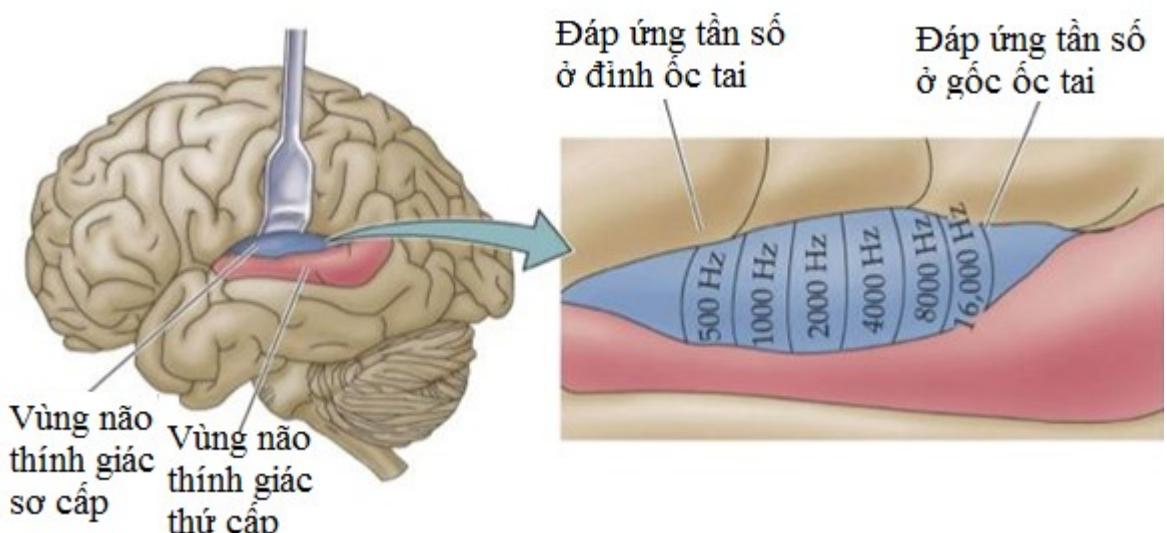


Hình 1. 4 Cộng hưởng với các tần số âm khác nhau ở óc tai

Các xung này lại tiếp tục được gửi đến vỏ não thính giác và được não tiếp nhận. Bộ não sẽ phân biệt âm thanh với các cao độ khác nhau qua các vị trí khác nhau mà những xung này được gởi đến từ các nang bào. Âm thanh có âm lượng càng lớn sẽ giải tỏa nhiều năng lượng hơn và làm di chuyển nhiều nang bào hơn. Bộ não phân biệt được các âm thanh là nhờ vào số lượng các nang bào cùng được kích hoạt trong một vị trí nào đó.

Khu vực vỏ não thính giác trước đây được chia thành các khu vực sơ cấp (A1), khu vực thứ cấp (A2) và vùng vành đai. Các quan điểm hiện đại [Pickles, 2012] [Purves, 2001] chia vỏ não thính giác thành các vùng là vùng lõi (A1), vùng vành đai và vùng parabelt. Vùng vành đai là khu vực ngay xung quanh lõi; vùng parabelt là tiếp giáp với phía bên của vành đai. Một số tác giả nghiên cứu về vai trò của não đối với hoạt động nhận thức chia vùng vỏ não thính giác thành vùng sơ cấp, vùng liên kết thính giác và vùng liên kết bậc cao hay còn gọi là vùng liên kết đa giác quan.

Chức năng của vỏ não thính giác sơ cấp là xử lý âm thanh. Vỏ não thính giác sơ cấp xử lý các thông tin như độ cao, âm lượng và vị trí của âm thanh, những đặc trưng này rất cần thiết cho việc hiểu ngôn ngữ. Các nơ-ron trong vỏ não thính giác được sắp xếp theo trật tự của tần số tương ứng với sự sắp xếp các sơ sinh học trong ốc tai, mỗi nơ-ron trong vỏ não thính giác phản ứng tốt nhất với một dải tần số cụ thể và được sắp xếp theo tần số từ cao xuống thấp từ gốc của đền đỉnh ốc tai. Vỏ não thính giác thứ cấp chịu trách nhiệm xử lý các tính chất âm thanh phức tạp hơn như các mẫu nhịp điệu trong khi vùng vành đai giúp tích hợp thính giác với các hệ thống giác quan khác.



Hình 1.5 Khu vực lưu trữ đặc trưng tiếng nói trên vỏ não

1.3. Quá trình mô phỏng nhận thức tiếng nói trên máy tính

Tín hiệu tiếng nói là tín hiệu tương tự, do đó khi biểu diễn tín hiệu tiếng nói trong môi trường tính toán tín hiệu số, việc biểu diễn và lưu trữ sao cho không bị mất thông tin là vấn đề rất quan trọng trong các hệ thống thông tin sử dụng tiếng nói. Biểu diễn tín hiệu tiếng nói dưới dạng số chịu ảnh hưởng quan trọng của lý thuyết lấy mẫu, do đó các trạng thái của tín hiệu có dải tần số giới hạn có thể được biểu diễn dưới dạng các mẫu lấy tuần hoàn theo một chu kì cố định được gọi là chu kì lấy mẫu. Phương pháp biểu diễn tín hiệu theo dạng sóng, được xem xét đến với việc bảo quản thông tin theo cách thông thường là giữ nguyên hình dạng sóng của tín hiệu tương ứng khi đã qua các bước lấy mẫu và lượng tử hóa tín hiệu. Phương pháp thứ hai được dùng để biểu diễn tiếng nói là phương pháp biểu diễn theo tham số. Phương pháp này xem xét đến trên khía

cạnh biểu diễn tín hiệu tiếng nói như là đầu ra của hệ thống tổng hợp tiếng nói. Để thu được các tham số biểu diễn tiếng nói, đầu tiên tín hiệu tiếng nói cũng được biểu diễn theo dạng sóng, nghĩa là tín hiệu tiếng nói được lấy mẫu và lượng tử hóa giống như phương pháp biểu diễn tín hiệu tiếng nói dạng sóng, sau đó sẽ tiến hành xử lý để thu được các tham số của tín hiệu tiếng nói của mô hình tổng hợp tiếng nói nêu trên. Các tham số của mô hình tổng hợp tiếng nói này thường được phân loại thành các tham số kích thích và các tham số của bộ máy phát âm tương ứng.

Để thu được biểu diễn của tín hiệu tiếng nói dưới dạng sóng người ta phải biểu diễn tín hiệu tiếng nói dưới dạng rời rạc. Quá trình rời rạc hoá tín hiệu tiếng nói bao gồm các bước sau: lấy mẫu tín hiệu tiếng nói, lượng tử hoá các mẫu, và mã hoá và nén tín hiệu.

1.3.1. *Lấy mẫu tín hiệu tiếng nói*

Lấy mẫu tín hiệu là quá trình chuyển đổi tín hiệu từ liên tục thành rời rạc bằng cách lấy từng mẫu (sample) của tín hiệu liên tục tại các thời điểm rời rạc. Vậy nếu tín hiệu $x(t)$ được đưa vào bộ lấy mẫu thì đầu ra là $x(nT) \equiv x(n)$ với T là chu kỳ lấy mẫu. Nghịch đảo của chu kỳ lấy mẫu sẽ được gọi là tần số lấy mẫu. Sau khi lấy mẫu, tín hiệu liên tục trở thành dãy các giá trị rời rạc và có thể lưu trữ trong bộ nhớ máy tính để xử lý. Khi lấy mẫu một tín hiệu tương tự với tần số lấy mẫu f_0 , cần đảm bảo rằng việc khôi phục lại tín hiệu đó từ tín hiệu rời rạc tương ứng phải thực hiện được. Shanon đã đưa ra một định lý để xác định tần số lấy mẫu đảm bảo khôi phục được tín hiệu gốc. Theo Shanon, điều kiện cần và đủ để khôi phục lại tín hiệu tương tự từ tín hiệu đã được rời rạc với tần số lấy mẫu f_0 là: $f_0 > F_{\max}$ với F_{\max} là thành phần tần số lớn nhất của tín hiệu tương tự.

Dải tần số của tín hiệu âm thanh mà con người có thể nghe được là từ 16Hz đến 20kHz, do đó theo định lý Shanon thì tần số lấy mẫu tối thiểu là 40kHz. Với tần số lấy mẫu lớn như thế thì khôi lượng bộ nhớ dành cho việc ghi âm sẽ rất lớn và làm tăng sự phức tạp trong tính toán. Vì vậy tùy mục đích ứng dụng của việc số hóa tiếng nói, tín hiệu tiếng nói có thể được lọc bỏ các thành phần tần số cao mà vẫn đảm bảo chất lượng, chẳng hạn như đối với tín hiệu tiếng nói cho điện thoại, người ta thấy rằng ngữ nghĩa của thông tin vẫn đảm

bảo khi phô được giới hạn ở 3400Hz, khi đó tần số lấy mẫu sẽ là 8000Hz. Do đó, trong xử lý tiếng nói, tần số lấy mẫu có thể dao động trong khoảng 6000-16000Hz tùy theo mục đích của bài toán.

1.3.2. Lượng tử hóa các mẫu

Lượng tử hóa các mẫu là quá trình chuyển đổi tín hiệu rời rạc có biên độ liên tục thành tín hiệu rời rạc có biên độ rời rạc. Mỗi mẫu tín hiệu được biểu diễn bằng một giá trị chọn từ trong tập hữu hạn các giá trị có thể có. Sự khác nhau giữa giá trị của mẫu chưa lượng tử hóa $x(n)$ và giá trị của mẫu đã lượng tử hóa $x_q(n)$ được gọi là sai số lượng tử hóa.

Về mặt toán học, lượng tử hóa chính là làm tròn giá trị của các mẫu rời rạc. Gọi giá trị lượng tử hóa là mức lượng tử hóa, khoảng cách giữa hai mức lượng tử hóa cạnh nhau là bước lượng tử hóa Δ , sai số lượng tử hóa trong trường hợp làm tròn nằm trong giới hạn là:

$$-\frac{\Delta}{2} \leq e_q(n) \leq \frac{\Delta}{2}$$

Nếu x_{\min} và x_{\max} là giá trị nhỏ nhất và lớn nhất của $x(n)$ và L là số mức lượng tử hóa thì:

$$\Delta = \frac{x_{\max} - x_{\min}}{L - 1}$$

Ta gọi $x_{\max} - x_{\min}$ là dải động của tín hiệu và Δ là độ phân giải.

1.3.3. Mã hóa các mẫu lượng tử hóa

Mã hóa các mẫu là quá trình gán cho mỗi mẫu lượng tử hóa một số nhị phân. Nếu ta có L mức lượng tử hóa, ta cần ít nhất L số nhị phân. Với từ mã dài b bit ta có 2^b số nhị phân khác nhau. Như vậy yêu cầu $b \geq \log_2 L$.

Tốc độ lấy mẫu càng cao và độ phân giải lượng tử hóa càng lớn (b lớn) thì kích thước dữ liệu số càng lớn.

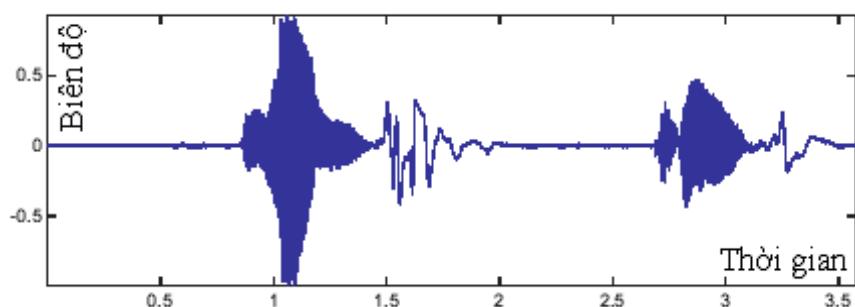
1.3.4. Biểu diễn tín hiệu tiếng nói.

Tín hiệu tiếng nói có thể được biểu diễn trên miền thời gian hoặc miền tần số, hoặc kết hợp thời gian và tần số. Tín hiệu tiếng nói xét trên miền thời gian có thể coi là tín hiệu ít biến đổi khi ta chỉ xét một khoảng thời gian đủ ngắn

(5-100ms), điều đó có nghĩa là tín hiệu tiếng nói có thể coi là ổn định trong khoảng thời gian ngắn. Tuy nhiên khi xét trong một khoảng thời gian dài hơn (0.5s) thì tín hiệu tiếng nói lại không ổn định, hay nó thay đổi theo các âm khác nhau được phát âm bởi người nói.

Để có thể thực hiện các phân tích trên tín hiệu tiếng nói nhằm tìm ra các đặc trưng riêng cho các đoạn tín hiệu ứng với các âm khác nhau, trước hết chúng ta cần có các phương pháp để biểu diễn tín hiệu tiếng nói. Sau đây là một số phương pháp thường được dùng.

1.3.4.1. Tín hiệu tiếng nói trên miền thời gian

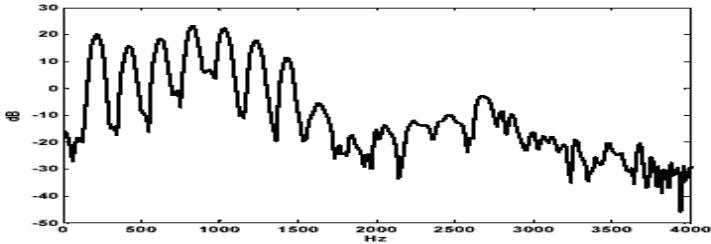


Hình 1. 6 Biểu diễn tín hiệu tiếng nói trên miền thời gian

Trên miền thời gian tín hiệu tiếng nói được biểu diễn bởi đồ thị biên độ tại các thời điểm t khác nhau, trong tự nhiên đó là một đồ thị liên tục, tuy nhiên tín hiệu tiếng nói được xử lý trong máy tính đã được số hóa nghĩa là rời rạc cả về mặt thời gian và tần số.

1.3.4.2. Tín hiệu tiếng nói trên miền tần số

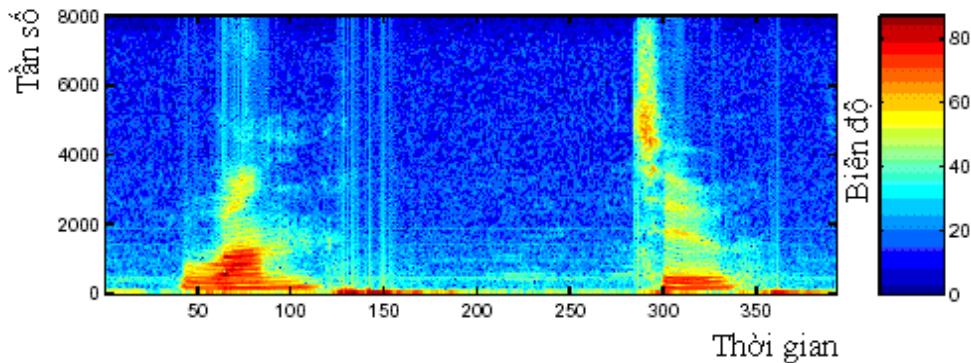
Tín hiệu tiếng nói không phải chỉ có một thành phần tần số mà gồm rất nhiều thành phần tần số khác nhau, tần số lớn nhất có thể lên tới hơn 10kHz [Stevens, 1998]. Mặt khác, mức độ tham gia của các thành phần tín hiệu này trong một tín hiệu tiếng nói cũng khác nhau. Dạng biểu diễn tín hiệu tiếng nói trên miền thời gian không chứa đủ thông tin để phân tích các thành phần tín hiệu ở các tần số khác nhau, vì vậy người ta cần đến dạng biểu diễn tín hiệu tiếng nói trong miền tần số, hay còn gọi là phổ tín hiệu.



Hình 1.7 Biểu diễn tín hiệu tiếng nói trên miền tần số

1.3.4.3. Tín hiệu tiếng nói trên miền thời gian và tần số kết hợp

Trong khi nghiên cứu tiếng nói, người ta luôn có gắng biểu diễn tín hiệu nhằm thu được nhiều thông tin nhất từ hình biểu diễn. Một trong những phương pháp biểu diễn được dùng nhiều nhất và đó là cách biểu diễn tín hiệu trên miền kết hợp thời gian và tần số gọi là phổ tần số. Thực chất của cách biểu diễn này là biểu diễn tín hiệu trên miền tần số nhưng được thực hiện với các đoạn tín hiệu ổn định (thời gian đủ ngắn) theo thời gian. Các giá trị biên độ được thể hiện bằng màu sắc.



Hình 1.8 Biểu diễn tín hiệu tiếng nói trên miền kết hợp

1.3.5. Trích chọn đặc trưng tiếng nói

Các mô hình học máy cho bài toán nhận thức tiếng nói thường cố gắng trích chọn đặc trưng tiếng nói ở một mức độ nhất định tùy theo hướng tiếp cận mô phỏng tương ứng với vùng não thính giác nào. Một số hệ thống tiếp cận theo hướng trích chọn các đặc trưng thính giác mức thấp đó là trích các đặc trưng về thành phần tần số, biên độ các thành phần tần số trong tín hiệu âm thanh. Một số hệ thống trích chọn các đặc trưng cao hơn là sự kết hợp của các đặc trưng mức thấp tạo thành các mẫu như âm vị, từ,..

1.3.6. Phân lớp, phân cụm dữ liệu

Các mô hình học máy cho bài toán nhận thức tiếng nói thường có 2 pha, pha huấn luyện hoặc thu thập mẫu và pha nhận dạng. Đối với bài toán dữ liệu có gán nhãn ở pha thứ hai sẽ phân lớp tín hiệu đầu vào thuộc một lớp nhãn dữ liệu nào đó đã được gán. Đối với bài toán dữ liệu không có nhãn, tín hiệu sẽ được phân cụm vào cùng với các tín hiệu sao cho chúng được coi là gần nhau nhất theo một khoảng cách nào đó.

1.4. Tổng quan tình hình nghiên cứu về nhận thức tiếng nói

Sự ra đời của các phương pháp tổng hợp tiếng nói và hệ thống phân tích âm thanh hiện đại trong giữa thế kỷ thứ 20 đã thúc đẩy sự phát triển của các nghiên cứu về nhận thức tiếng nói một cách mạnh mẽ. Nhiều thí nghiệm được tiến hành để đánh giá những ảnh hưởng khác nhau từ xử lý thông tin, sinh học/tâm lý học và ngữ âm tới nhận thức tiếng nói. Nhiều lý thuyết đã được phát triển để giải thích nguyên nhân nào cho phép biến đổi những tín hiệu có thể thay đổi thành đơn vị nhận thức bất biến. Nhưng những đơn vị nhận thức bất biến đó là gì?

Những nghiên cứu đầu tiên về nhận thức tiếng nói là nghiên cứu khả năng phân biệt một tín hiệu nhất định từ các âm thanh khác mà chúng xuất hiện đồng thời trong cùng môi trường. Khả năng này cho phép người nghe nhóm một số âm thanh thành một nhóm mà chúng được bắt nguồn từ cùng cơ quan phát âm, tách chúng ra khỏi các âm thanh khác. Bài toán này đặc biệt quan trọng và thực sự khó khăn khi tách các tín hiệu nhiều cũng là tiếng nói. Bài toán còn được gọi tên là hiệu ứng bữa tiệc (*cocktail-party effect*) hay đôi khi bài toán cũng được gọi sự nhận thức nhiều người nói (*multi-talker perception*). Thuật ngữ ‘cocktail-party effect’ được Cherry đưa ra và nghiên cứu đầu tiên vào năm 1953. Trong hàng loạt thí nghiệm của ông người nghe được nghe một thông điệp có nhiều bằng cả hai tai hoặc từng tai riêng biệt. Tiếp đó, năm 1957, Broadbent và Ladefoged nghiên cứu một bài toán hẹp hơn của bài toán này [Broadbent, 1957] đó là tập trung vào bài toán làm thế nào để nhận ra một người đang nói gì trong khi những người khác đang nói cùng một lúc, hay một bài toán cụ thể hơn là phân biệt hai nguyên âm chẳng hạn như /i/ và /e/ đồng thời, người nghe có thể nhóm các đỉnh cộng hưởng thích hợp lại với nhau sao cho người nghe có thể nhận biết được đó là hai nguyên âm riêng biệt chứ không

phải là một sự kết hợp của các đỉnh cộng hưởng. Ông đưa ra lập luận rằng theo lý thuyết thính giác thì tần số của các tín hiệu được xử lý và nhận biết ở màng đáy của hệ thính giác vì vậy không có sự khác nhau giữa việc nghe ở trong điều kiện một bên tai hay nghe bằng hai tai. Kết quả nghiên cứu của này minh họa tổng hợp của các đỉnh cộng hưởng riêng biệt chỉ thành một âm thanh đơn khi chúng có cùng tần số cơ bản F0 bất kể khi chúng được nghe bằng một tai hay cả hai tai. Hướng tiếp cận này chỉ tập trung trong việc trích chọn các đặc trưng của tiếng nói để nhận thức được người nói, hoặc nhận thức được các thành phần cơ bản của ngôn ngữ nói.

Hướng tiếp cận tích hợp nguồn hay khả năng tích hợp thông tin từ nhiều phương thức khác nhau cho bài toán nhận dạng tiếng nói cũng được nghiên cứu từ rất sớm. Đầu năm 1954, Sumby và Pollack đã chứng minh rằng sự kết hợp của thính giác và thị giác (*audio-visual*) làm tăng khả năng nhận dạng các âm tiết, đồng thời các tác giả cũng nhấn mạnh rằng đóng góp của thị giác là lớn nhất khi nhận dạng các từ trong môi trường có nhiễu cao [Sumby, 1954]. Tiếp đó, năm 1998, Massaro và đồng nghiệp đã đề xuất mô hình nhận thức tiếng nói bằng cách kết hợp thị giác với thính giác (*audio-visual*) và được nhiều tác giả nghiên cứu trong giai đoạn này như một hướng nghiên cứu chính [Massaro, 1998] [Rosenblum]. Trong hướng tiếp cận này, các tác giả đã đưa thêm thông tin từ thị giác nhằm mục đích nâng cao hiệu quả nhận thức tiếng nói

Hướng nghiên cứu vai trò của não đối với nhận thức tiếng nói đầu tiên được thực hiện bởi Kimura [Kimura, 1961a] [Kimura, 1961b]. Trong nghiên cứu này, Kimura cho các bệnh nhân nghe một nhóm sáu chữ số, ba chữ số cho mỗi bên tai, và bệnh nhân nói lại bất cứ điều gì họ có thể nhớ. Kimura kết luận rằng tiếng nói được xử lý hiệu quả hơn trong tai là bên đối diện với bán cầu ngôn ngữ chi phối, không phụ thuộc vào việc thuận tay của bệnh nhân và cho dù có những tổn thương ở bán cầu trái. Nghiên cứu này cho thấy sự phức tạp của các con đường nhận thức thính giác, vai trò thống trị não và mối quan hệ của nó với xử lý tiếng nói cũng như cách biểu diễn tiếng nói trong bán cầu não. Mặc dù khoa học đã có nhiều tiến bộ kể từ năm 1990 đến nay, nhưng bài toán nghiên cứu để hiểu rõ vai trò của não bộ đối với việc nhận thức tiếng nói vẫn còn nhiều thách thức.

Một trong số tác giả nghiên cứu về vai trò của bộ nhớ đối với nhận thức tiếng nói có thể kể đến là Miller. Ngay từ năm 1956, Miller đã nghiên cứu về bộ nhớ ngắn hạn (*short-term memory*) ở người trưởng thành đối với việc ghi nhớ và truy xuất thông tin [Miller G. , 1956] . Tiếp đến, năm 1973, Pisoni cũng có một số nghiên cứu về tầm quan trọng của bộ nhớ đối với phân lớp âm thanh [Pisoni, 1973] . Các nghiên cứu gần đây về bộ nhớ và học tập đã xem xét vai trò tiềm năng của mẫu nhớ cho các từ cụ thể. Các nghiên cứu về bộ nhớ được thực hiện từ năm 1998 [Goldinger, 1998] hầu hết các thí nghiệm không chỉ ra một cách rõ ràng về ảnh hưởng của các thông số âm học, và trong hầu hết các trường hợp các thông tin về âm học chỉ góp phần vào việc nhận dạng người nói hơn là xác định được ý nghĩa hoặc cấu trúc ngôn ngữ. Allen và Miller [Allen, 2004] đã chỉ ra rằng người nghe có thể nhận dạng được người nói từ sự khác nhau của khoảng thời gian trước khi bắt đầu nguyên âm (VOT). Smith [Smith, 2004] cho thấy thông tin chi tiết về âm vị có thể cải thiện kết quả nhận dạng được các từ trong tiếng nói liên tục.

Các nghiên cứu về nhận dạng tiếng nói đã được một số tác giả nghiên cứu, tổng hợp và xây dựng nên các lý thuyết và mô hình cho bài toán nhận thức tiếng nói. Điển hình như Liberman và các đồng nghiệp đề xuất lý thuyết vận động [Liberman, 1967] năm 1967. Lý thuyết này cho rằng việc nhận thức tiếng nói liên quan đến đặc điểm của cách phát ra các tín hiệu tiếng nói đó (*gestures*). Lý thuyết lượng tử hóa (*Quantal Theory*) được Stevens phác thảo năm 1972 [Stevens, 1972] , và hoàn thành vào năm 1989 [Stevens, 1989] . Mô hình TRACE là một trong những mô hình đầu tiên được phát triển để nhận thức tiếng nói [McClelland, 1986] , và là một trong những mô hình được biết đến nhiều nhất. Mô hình TRACE là một framework trong đó chức năng chính là lấy tất cả các nguồn thông tin khác nhau trong tiếng nói và tích hợp chúng để xác định các từ đơn. Halle & Stevens tổng hợp các kết quả nghiên cứu trước đó cho bài toán nhận dạng tiếng nói thành mô hình nhận dạng tiếng nói dựa trên phân tích bằng tổng hợp (*analysis-by-synthesis*) [Halle, 1962] . Mô hình này gồm hai giai đoạn, mỗi giai đoạn đều liên quan đến phân tích bằng tổng hợp. Mô hình nhận thức tiếng nói Cohort được đề xuất bởi Marslen-Wilson vào năm 1987 để nhận dạng từ vựng [Marslen-Wilson, 1987] . Lý thuyết mẫu đã được giới thiệu lần đầu tiên vào năm 1995 trong tâm lý học như là một mô hình nhận thức và phân

lớp, cũng năm đó Lacerda và Johnson áp dụng cho bài toán nhận dạng tiếng nói, và sau đó, năm 2001, Pierrehumbert (2001) cũng áp dụng lý thuyết mẫu cho bài toán nhận dạng tiếng nói. Lý thuyết này dựa trên liên kết giữa bộ nhớ và kinh nghiệm trước với các từ vựng. Mô hình tính toán nơ ron [Kröger, 2009] mô phỏng các con đường thần kinh ở những vùng khác nhau của não bộ có liên quan khi tiếng nói được phát ra và nhận thức. Sử dụng mô hình này, các vùng não chưa tri thức tiếng nói thu được bằng cách huấn luyện các mạng thần kinh để phát hiện tiếng nói trong vùng vỏ não và vỏ não tiêu não. Mô hình Dual Stream, đề xuất bởi Hickok và Poeppel, chứng minh sự hiện diện của hai thần kinh chức năng mạng riêng biệt trong xử lý tiếng nói và thông tin ngôn ngữ [Hickok, 2000] [Hickok, 2007]. Một mạng lưới thần kinh chủ yếu xử lý với các giác quan và thông tin âm vị liên quan đến các khái niệm và ngữ nghĩa. Mạng còn lại hoạt động với giác quan và thông tin âm vị liên quan đến hệ thống động cơ và hệ thống câu âm.

Trong khoa học máy tính, nhiều mô hình học máy cũng được nghiên cứu và áp dụng cho bài toán nhận thức tiếng nói. Các mô hình học máy được nhiều tác giả áp dụng cho bài toán nhận thức tiếng nói phổ biến như mô hình Markov ẩn (HMM) [Juang, 1991], mô hình GMM [Bagul, 2013], phương pháp SVM [Aida-zade, 2016], hay sử dụng mạng nơ-ron [Tsenov, 2010]. Gần đây, với sự phát triển của kỹ thuật máy tính, mạng học sâu bắt đầu được nhiều tác giả nghiên cứu và sử dụng cho bài toán nhận thức tiếng nói [Sak, 2014] [Soltau, 2014] và kết hợp giữa mạng học sâu với các phương pháp truyền thống nhằm nâng cao hơn nữa độ chính xác của bài toán như kết hợp giữa mạng hồi quy (RNN) với mô hình ngôn ngữ [Chen, 2017], mô hình Markov ẩn (HMM) kết hợp với mạng học sâu (DNN) [Dominique, 2017]. Nhìn chung, các mô hình học máy cho bài toán nhận thức tiếng nói cũng chủ yếu tập trung vào khía cạnh khai thác các phương pháp học máy đối với tín hiệu tiếng nói để phân biệt được các tín hiệu tiếng nói khác nhau thông qua mối liên hệ giữa tín hiệu tiếng nói với đơn vị ngôn ngữ cho trước. Chưa có mô hình nào nghiên cứu việc xây dựng mô hình liên kết tín hiệu tiếng nói với các tín hiệu khác, để sau khi huấn luyện, người nghe có thể gợi nhớ lại các thông tin đã được liên kết với tín hiệu tiếng nói mỗi khi được nghe tín hiệu tiếng nói đó.

Nghiên cứu nhận thức tiếng ở Việt Nam cũng được một số nhà nghiên cứu bắt đầu từ những năm 1990. Các nghiên cứu về nhận thức tiếng nói chủ yếu tập trung vào bài toán nhận dạng tiếng nói. Ngoài ra, cũng có một số nghiên cứu về bài toán nhận dạng người nói, hay bài toán xác thực người nói. Trong nghiên cứu nhận dạng tiếng nói, có 2 nhóm nghiên cứu chính với bộ từ vựng lớn đó là nhóm nghiên cứu thuộc Viện Công nghệ thông tin với phương pháp sử dụng là mạng trí tuệ nhân tạo (ANN) và sử dụng bộ công cụ CSLU [Vu Thang, 2005] [Huy, 2003] [Đức, 2004] [Thang, 2008]. Nhóm thứ hai là nhóm nghiên cứu thuộc trường đại học Khoa học tự nhiên thành phố Hồ Chí Minh [Tuan, 2009]. Nhóm này thường sử dụng phương pháp HMM với bộ công cụ HTK. Các nghiên cứu tập trung vào bài toán truy vấn thông tin bằng tiếng Việt, nhận dạng tiếng nói, hệ thống giao tiếp giữa người và máy tính, tìm kiếm bằng giọng nói, hay bài toán dịch tự động trực tiếp từ tiếng nói. Gần đây, có thêm nhóm nghiên cứu thuộc phòng thí nghiệm MICA về sự khả chuyền của các mô hình ngữ âm (acoustic model portability).

Bên cạnh các nhóm nghiên cứu lớn, cũng có một số nhà nghiên cứu khác với nhiều đề tài nhận thức tiếng nói tập trung trong bài toán nhận dạng tiếng Việt và trong điều khiển người máy và bài toán dịch ngôn ngữ tự động [Phúc, 2000] [Hoan, 1996] [Vu Ngoc, 2009] [Van Huy, 2015] [Hong Quang, 2008], bài toán nhận dạng người nói bằng tiếng Việt [Dũng, 2010].

Tóm lại, các nghiên cứu về nhận thức tiếng nói đến nay, chủ yếu tập trung vào việc nghiên cứu các phương pháp trích chọn đặc trưng của tiếng nói, liên kết các đặc trưng của tiếng nói với khái niệm ngôn ngữ như định danh, âm tiết, từ, ... và phát triển các phương pháp học máy để nâng cao khả năng phân biệt các tín hiệu tiếng nói với nhau, chưa xét đến góc độ nhận thức tiếng nói ở mức nhận thức được các đặc điểm, đặc trưng của sự vật, hiện tượng mà tín hiệu tiếng nói đề cập tới. Ví dụ, khi nghe được từ ‘quả chanh’ thì chúng ta có thể gợi nhớ lại được các đặc điểm về hình dáng, màu sắc, kích thước, mùi vị, của quả chanh. Đó là những thông tin thu được từ các giác quan khác đã được liên kết với tín hiệu tiếng nói của từ quả chanh mà chúng ta đã học được trước đây.

1.5. Bài toán nhận thức tiếng nói trong khoa học máy tính

Dựa vào đặc điểm hoạt động của mô hình nhận thức tiếng nói trong máy tính, chúng tôi chia bài toán nhận thức tiếng nói thành hai cấp độ: cấp độ thứ nhất là bài toán nhận dạng, và cấp độ thứ hai là bài toán nhận thức. Ở bài toán nhận dạng, các tín hiệu tiếng nói được liên kết với một khái niệm được cung cấp bởi tri thức sẵn có của con người. Như liên kết một tín hiệu tiếng nói với một âm tiết, một từ, hay liên kết với một tên định danh biết trước. Ở cấp độ nhận thức, tín hiệu tiếng nói không được cung cấp các tri thức có sẵn, mà là do tự học trong quá trình huấn luyện, hoạt động.

1.5.1. Bài toán nhận dạng người nói

Bài toán nhận dạng người nói là một bài toán con của bài toán nhận thức tiếng nói trong đó các tín hiệu tiếng nói được liên kết với một định danh gắn với người nói do con người cung cấp. Thông qua việc trích chọn các đặc trưng khác nhau do hệ thống phát âm khác nhau của người nói mà hệ thống phân biệt được tín hiệu tiếng nói là của người nào.

Nhận dạng người nói có nhiều ứng dụng như xác thực quyền truy nhập vào các hệ thống an ninh bằng giọng nói, giám sát người qua giọng nói hay tách tiếng nói của từng người từ môi trường có nhiều người nói, ứng dụng xác thực người nói trong các giao dịch điện tử hay trong lĩnh vực giám định pháp lý người nói.

Dựa vào chức năng của bài toán nhận dạng người nói người ta chia bài toán nhận dạng người nói thành hai bài toán: bài toán định danh người nói (*speaker identification*) và bài toán xác thực người nói (*speaker verification*).

Dựa theo phương pháp thì bài toán nhận dạng được chia thành hai bài toán: bài toán nhận dạng người nói phụ thuộc vào từ khóa (*text-dependent speaker recognition*) và bài toán nhận dạng người nói không phụ thuộc vào từ khóa (*text-independent speaker recognition*).

Có 3 phương pháp nhận dạng người nói đang được sử dụng phổ biến hiện nay đó là nhận dạng thủ công bằng cách so sánh phổ tần số của hai mẫu tiếng nói để quyết định xem liệu chúng có phải do cùng một người nói hay không và phương pháp tự động nhận dạng người nói được thực hiện tự động dựa trên việc mô hình hóa tín hiệu tiếng nói bằng cách trích chọn các đặc trưng

thông tin người nói và sử dụng các phương pháp học máy để học và phân lớp và nhận dạng người nói bằng cơ quan thính giác.

1.5.2. Bài toán nhận dạng tiếng nói

Bài toán nhận dạng tiếng nói cũng là một bài toán con của bài toán nhận thức tiếng nói trong đó các đoạn tín hiệu tiếng nói được liên kết với một âm tiết hoặc một từ trong một ngôn ngữ nào đó (tiếng Anh, tiếng Việt,...) do con người cung cấp. Thông qua việc trích chọn các đặc trưng cấu thành âm tiết, từ khác nhau để hệ thống phân biệt được các tín hiệu tiếng nói là tương ứng với âm tiết, hay từ nào.

Dựa vào các đặc điểm của hệ thống, hệ thống nhận dạng tiếng nói có thể có các cách phân loại sau:

- Nhận dạng tiếng nói rời rạc và nhận dạng tiếng nói liên tục: Trong các hệ thống nhận dạng các từ phát âm rời rạc yêu cầu người nói phải dừng một khoảng trước khi nói từ tiếp theo trong khi hệ thống nhận dạng các từ phát âm liên tục không đòi hỏi yêu cầu này.

- Nhận dạng tiếng nói độc lập người nói và nhận dạng tiếng nói phụ thuộc người nói: Đối với hệ thống nhận dạng phụ thuộc người nói đòi hỏi tiếng nói người nói phải có trong cơ sở dữ liệu của hệ thống, còn đối với hệ thống nhận dạng không phụ thuộc người nói thì người nói không nhất thiết phải có mẫu trong cơ sở dữ liệu của hệ thống trước khi nhận dạng.

- Nhận dạng tiếng nói với từ điển cỡ nhỏ, nhận dạng tiếng nói với từ điển cỡ vừa hay cỡ lớn: Hiệu năng của một hệ thống nhận dạng với từ điển cỡ nhỏ thường cao hơn hiệu năng của các hệ thống nhận dạng có từ điển cỡ vừa và cỡ lớn.

- Nhận dạng tiếng nói trong môi trường nhiều cao và nhận dạng tiếng nói trong môi trường nhiễu thấp: Hiệu năng của các hệ thống nhận dạng tiếng nói không bị nhiễu sẽ cao hơn hiệu năng của các hệ thống nhận dạng tiếng nói có nhiễu.

Các hệ thống nhận dạng tiếng nói tự động được chia làm ba hướng tiếp cận như sau: Hướng tiếp cận ngữ âm - âm học dựa trên lý thuyết âm học – ngữ âm. Lý thuyết này khẳng định sự tồn tại hữu hạn và duy nhất các đơn vị ngữ

âm cơ bản trong ngôn ngữ nói gọi là âm vị, được phân chia thành: nguyên âm - phụ âm, vô thanh-hữu thanh, âm vang - âm bẹt,... Các âm vị có thể xác định bởi tập các đặc trưng trong phô của tín hiệu tiếng nói theo thời gian; Hướng tiếp cận nhận dạng mẫu dựa vào lý thuyết xác suất - thống kê để nhận dạng dựa trên ý tưởng: so sánh đối tượng cần nhận dạng với các mẫu được thu thập trước đó để tìm mẫu giống đối tượng nhất; Hướng tiếp cận sử dụng mạng nơ-ron đặc biệt là mạng học sâu đang được sử dụng và tỏ ra rất thành công trong các bài toán nhận dạng nói chung và bài toán nhận thức tiếng nói riêng.

1.5.3. Bài toán nhận thức tiếng nói

Nhận thức tiếng nói là quá trình mà người nghe nghe các tín hiệu âm thanh của tiếng nói và phân biệt được sự vật, hiện tượng thông qua việc phản ánh được đối tượng bằng các giác quan của chủ thể nhận thức để từ đó có những phản ứng tương ứng phù hợp với tín hiệu tiếng nói được nghe. Ví dụ, khi chúng ta được nghe từ “quả chanh” chúng ta sẽ tưởng tượng là quả chanh có hình tròn, màu xanh, có mùi thơm nhẹ, có vị chua, thậm chí chúng ta sẽ có phản xạ tiết nước miếng,... nghĩa là chúng ta đã nhận thức được từ ‘quả chanh’. Để có được nhận thức về “quả chanh” chúng ta đã phải được nghe (thính giác), được nhìn (thị giác), được cầm (xúc giác), được ngửi (khứu giác), được ăn (vị giác) để có được các thông tin liên kết với từ ‘quả chanh’ đó.

Như vậy, ở cấp độ này, nhận thức tiếng nói là một quá trình học trực tiếp mối liên hệ giữa tín hiệu tiếng nói với các thông tin thu được từ các giác quan khác và thiết lập nên một mạng quan hệ hay ảnh xạ giữa tín hiệu tiếng nói với các tín hiệu khác trong vùng vỏ não liên kết đa giác quan, từ đó có thể hiểu được tiếng nói, có phản ứng phù hợp với tín hiệu tiếng nói được nghe sau này. Sau khi học xong, khi có một tín hiệu tiếng nói mới được nghe, não bộ sẽ gọi lại các thông tin liên kết với tín hiệu tiếng nói đó đồng thời sẽ điều khiển các hoạt động của cơ thể tương ứng với tín hiệu được nghe.

Trong luận án này, chúng tôi tiếp cận khái niệm nhận thức tiếng nói dưới góc độ xây dựng mạng liên kết của tín hiệu tiếng nói với các tín hiệu khác, cụ thể liên kết với khái niệm có sẵn trong bài toán nhận thức tiếng nói ở mức độ nhận dạng, và liên kết với các thông tin khác thu được từ các bộ cảm biến khác

nhau. Trong giới hạn của luận án này, chúng tôi chỉ tập trung vào xây dựng liên kết giữa tín hiệu tiếng nói với tín hiệu hình ảnh.

1.6. Một số khó khăn trong nhận thức tiếng nói

1.6.1. Tính tuyến tính

Trong một phát âm liên tục mỗi âm thường chịu ảnh hưởng rất lớn từ các âm trước và sau nó. Vì vậy các từ được phát âm rời rạc khi nhận dạng sẽ có độ chính xác cao hơn là các từ trong một phát âm liên tục. Do chất lượng nhận dạng cho một chuỗi phát âm liên tục còn phụ thuộc thêm vào việc phát hiện biên và khoảng trống giữa hai từ. Khi người nói phát âm với tốc độ cao thì khoảng trống và biên giữa các từ sẽ bị thu hẹp dẫn đến việc phân đoạn từng từ có thể bị nhầm lẫn hoặc trùm lên nhau làm ảnh hưởng đến độ chính xác cho việc nhận dạng từ đó.

1.6.2. Phân đoạn tiếng nói

Phân đoạn tiếng nói là quá trình xác định ranh giới giữa các từ, âm tiết, âm vị trong ngôn ngữ nói. Giống như hầu hết các vấn đề xử lý ngôn ngữ tự nhiên, để phân đoạn người ta phải đưa tiếng nói vào ngữ cảnh, ngữ pháp và ngữ nghĩa, và ngay cả như vậy kết quả phân đoạn tiếng nói thường cũng chỉ đạt được ở một mức độ tương đối nguyên nhân do hiện tượng khớp nối âm xảy ra giữa các âm vị, hay các từ lân cận nhau.

1.6.3. Vấn đề phụ thuộc người nói

Mỗi người nói sẽ có cấu trúc của bộ máy tạo âm khác nhau dẫn đến đặc tính của tiếng nói phát ra chịu ảnh hưởng rất nhiều vào người nói. Ngay cả đối với một người nói khi phát âm cùng một câu thì tiếng nói phát ra cũng có thể khác nhau do lưu lượng không khí thoát ra từ phổi, tình trạng cảm xúc, sức khỏe, độ tuổi khác nhau.

1.6.4. Vấn đề nhiễu

Trong thực tế tín hiệu tiếng nói thường bị ảnh hưởng bởi các tạp âm từ môi trường ngoài như phương tiện giao thông, tiếng động vật, hay tiếng nói của một hoặc nhiều người khác nói cùng thời điểm. Đối với con người việc phân biệt và tập trung vào một người đang nói để hiểu và phân biệt ngữ nghĩa là đơn giản tuy nhiên đối với máy tính các trường hợp như vậy sẽ gây ra những khó

khăn để nhận dạng do micro thu mọi loại tín hiệu âm trong băng tần mà nó làm việc. Hiện nay, ngay cả khi áp dụng các phương pháp tiền xử lý tối ưu trên tín hiệu thu được, đồng thời tách lọc tín hiệu của người nói thì chất lượng nhận thức cho các trường hợp này vẫn còn rất thấp

1.6.5. Đơn vị nhận thức cơ bản

Một vấn đề quan trọng trong nhận thức tiếng nói là lựa chọn đơn vị nhỏ nhất để phân tích. Nhiều nhà nghiên cứu sử dụng các đặc trưng, âm vị, âm tiết hoặc từ là các đơn vị nhận thức cơ bản trong khi một số nhà nghiên cứu khác đề xuất sử dụng các đơn vị nhận thức lớn hơn như cụm từ, mệnh đề, câu [Bever, 1969] [Miller G. , 1962] [Miller G. , 1962] [Johnson, 1997] . Vì vậy, đơn vị nhận thức cơ bản là gì hiện vẫn còn nhiều tranh cãi. Tùy vào mục đích của từng nghiên cứu, các nhà nghiên cứu vẫn đang sử dụng một trong những đơn vị như âm vị, từ và câu để làm đơn vị nhận thức cơ bản cho bài toán của mình.

1.7. Mô hình nhận thức tiếng nói dựa trên học quan hệ giữa tín hiệu tiếng nói với các tín hiệu khác

Từ những phân tích trên có thể thấy bài toán nhận thức là một lĩnh vực rất rộng, khái niệm nhận thức tiếng nói có thể hiểu là “*nhận thức tiếng nói là nhận thức được sự khác nhau giữa các tín hiệu tiếng nói*” để từ đó có hành động đáp ứng phù hợp. Sự khác nhau đó có thể là nhận thức được tiếng nói đó được nói bởi những người khác nhau, tiếng nói đó thuộc các lớp khác nhau như nguyên âm hay phụ âm, hữu thanh hay vô thanh, khi các lớp này là đại diện cho các đơn vị ngôn ngữ thì ta có bài toán nhận dạng tiếng nói.

Trong khuôn khổ của nghiên cứu này chúng tôi chỉ tập trung nghiên cứu tới khía cạnh nhận thức tiếng nói trong mối liên hệ với các khái niệm và trong mối liên hệ với các tín hiệu khác. Từ đó, đề xuất mô hình nhận thức tiếng nói dựa trên mô hình mô phỏng quá trình liên kết thông tin ở vùng vỏ não liên kết bậc cao nơi liên kết thông tin giữa các cơ quan cảm giác đặc biệt là liên kết thông tin giữa cơ quan thính giác và cơ quan thị giác. Đây là một hướng tiếp cận mới so với các tiếp cận trước đây cho bài toán nhận thức tiếng nói bởi vì các hướng tiếp cận trước đây chủ yếu tập trung mô phỏng quá trình nhận thức tiếng nói ở vùng nhớ sơ cấp và vùng nhớ liên kết của cơ quan thính giác, rất ít nghiên cứu đề cập tới vùng nhớ liên kết đa giác quan này.

Từ đó, bài toán nhận thức tiếng nói trong nghiên cứu này gồm những các nhiệm vụ cụ thể sau:

- Nghiên cứu trích chọn đặc trưng tiếng nói dựa trên đặc trưng phổ tần số của tiếng nói. Để đánh giá sự phù hợp của đặc trưng này, áp dụng đặc trưng cho bài toán nhận thức tiếng nói trong mối liên hệ với các khái niệm hay bài toán nhận dạng tiếng nói độc lập.
- Nghiên cứu và xây dựng mô hình nhận thức tiếng nói dựa trên việc học mối quan hệ giữa tín hiệu tiếng nói và tín hiệu hình ảnh thu được đồng thời từ hai hệ thống giác quan.
- Nghiên cứu các phương pháp rút gọn dữ liệu, đề xuất một phương pháp rút gọn đặc trưng để tăng hiệu quả của cho mô hình nhận thức tiếng nói.
- Nghiên cứu và đề xuất giải pháp cho bài toán nhận thức tiếng nói đối với dữ liệu lớn.

Chương 2. MỘT SỐ HƯỚNG TIẾP CẬN HỌC MÁY CHO BÀI TOÁN NHẬN THỨC TIẾNG NÓI

2.1. Giới thiệu

Nhận thức tiếng nói đã được nghiên cứu hơn 70 năm [Sumby, 1954], rất nhiều lý thuyết, mô hình đã được đưa ra nhằm giải thích cơ chế nhận thức tiếng nói ở con người. Các nghiên cứu tập trung vào hai vấn đề lớn đó là hướng tiếp cận cho bài toán nhận thức tiếng nói và nghiên cứu cách thức xử lý các đặc trưng tiếng nói trong não người.

Chương này giới thiệu một số kiến thức cơ sở của bài toán nhận thức tiếng nói, hướng tiếp cận trong khoa học máy tính và một số phương pháp trích chọn đặc trưng tiếng nói cho các mô hình nhận thức tiếng nói trên máy tính.

2.2. Một số mô hình học máy cho bài toán nhận thức tiếng nói

2.2.1. Mô hình Markov ẩn

Mô hình Markov ẩn (Hidden Markov Model - HMM) là mô hình học máy điển hình tiếp cận theo mô hình âm học [Klatt, 1979] cho bài toán nhận dạng tiếng nói. HMM là mô hình xác suất dựa trên lý thuyết về chuỗi Markov bao gồm các thành phần sau:

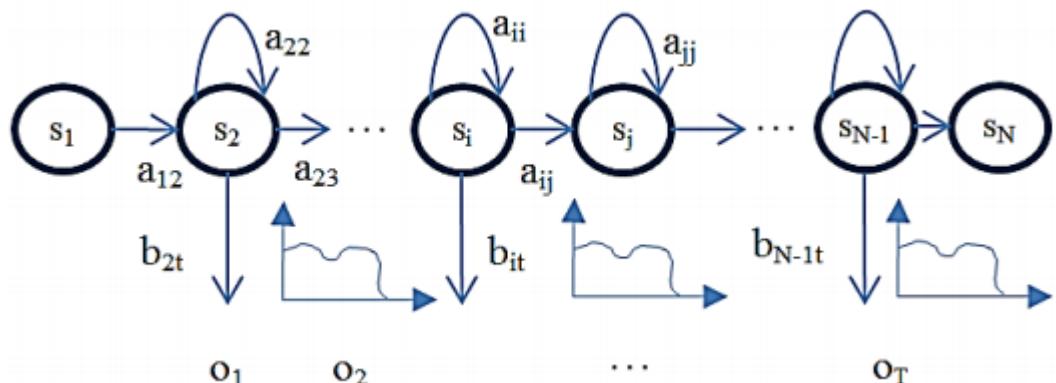
- * $O = \{o_1, o_2, \dots, o_T\}$ là tập các vector quan sát gồm T phần tử.
- * $S = \{s_1, s_2, \dots, s_N\}$ là tập hữu hạn các trạng thái s gồm N phần tử
- * $A = \{a_{11}, a_{12}, \dots, a_{MN}\}$ là ma trận hai chiều trong đó a_{ij} thể hiện xác suất để trạng thái s_i chuyển sang trạng thái s_j , với $a_{ij} \geq 0$ và $\sum_{j=1}^k a_{ij} = 1 \forall i$.
- * $B = \{b_{2t}, b_{it}, \dots, b_{(N-1)t}\}$ là tập các hàm phân phối xác suất của các trạng thái từ s_2 đến s_{N-1} , trong đó b_{it} thể hiện xác suất để quan sát o_t thu được từ trạng thái s_i tại thời điểm t. Trong nhận dạng tiếng nói hàm b_{it} thường được sử dụng là hàm Gaussian với nhiều thành phần trộn (mixture), khi đó mô hình được gọi là mô hình kết hợp Hidden Markov Model và Gaussian Mixtrue Model (HMM-GMM).
- * $\Pi = \{\pi_{ij}\}$ là tập xác suất trạng thái đầu, với $\pi_i = P(q_1 = s_i)$ với $i=1..N$ là xác suất để trạng thái s_i là trạng thái đầu q_1 .

Như vậy một cách tổng quát một mô hình Markov ẩn λ có thể được biểu diễn bởi $\lambda = (A, B, \Pi)$. Trong lĩnh vực nhận dạng thì mô hình Markov ẩn được áp dụng với hai giả thiết sau:

- Một là giả thiết về tính độc lập, tức không có mối liên hệ nào giữa hai quan sát lân cận nhau o_i và o_{i+1} , khi đó xác suất của một chuỗi các quan sát $O=\{o_t\}$ có thể được xác định thông qua xác suất của từng quan sát o_i như sau:

$$P(O) = \prod_{i=1}^T P(o_i) \quad (2.1)$$

- Hai là giả thiết Markov, xác suất chuyển thành trạng thái s_t chỉ phụ thuộc vào trạng thái trước nó s_{t-1} .



Hình 2.1 Mô hình HMM-GMM có cấu trúc dạng Left-Right liên kết không đầy đủ

Trong nhận dạng tiếng nói, mô hình HMM-GMM có thể được sử dụng để mô hình hóa cho các đơn vị tiếng nói như Âm vị (phoneme), Từ (word) hoặc Câu (sentence). Khi đó tập quan sát $O=\{o_t\}$ sẽ tương ứng với mỗi một phát âm (utterance) trong đó o_t là tập các vector đặc trưng (feature vector) của tín hiệu tiếng nói đầu vào thu được tại thời điểm t . Có nhiều cấu trúc HMM khác nhau, tuy nhiên trong thực tế, cấu trúc của HMM-GMM thường được sử dụng có 5 hoặc 7 trạng thái theo cấu trúc Left-Right được mô tả ở trên. Quá trình xây dựng một hệ thống nhận dạng tiếng nói sử dụng mô hình HMM-GMM thông thường có hai bước như sau:

- Huấn luyện (Training): Đối với từng ngôn ngữ, dữ liệu và mục đích cụ thể ta sẽ dùng HMM-GMM để mô hình cho các đơn vị nhận dạng là âm vị, Từ

hoặc Câu. Khi đó một hệ thống sẽ bao gồm một tập các mô hình HMM-GMM $\lambda = \{\lambda_i\}$. Đối với mỗi phát âm $O = \{o_t\}$ được mô hình bởi một chuỗi các trạng thái $Q = \{q_t\}$ với từ một hoặc nhiều mô hình λ_i . Quá trình huấn luyện là quá trình ước lượng các tham số sao cho xác suất $P(Q|O, \lambda)$ là lớn nhất, $P(Q|O, \lambda)$ được tính theo công thức (2.2), khi đó $P(Q|O, \lambda)$ được gọi là xác suất mô hình âm học (acoustic model).

$$P(Q|O, \lambda) = \sum_{q_t}^Q \pi_{tk} a_{t_{k-1} t_k} b_{tk}(o_t), k = 1..N \quad (2.2)$$

- Nhận dạng (decoding): Nhận dạng là quá trình xác định chuỗi trạng thái $\{q_i\} = Q, q_i \in S$ từ các mô hình HMM $\{\lambda_i\} = \lambda$ đã được huấn luyện tương ứng với một chuỗi đầu vào $\{o_t\} = O$ sao cho xác suất $P(O, Q|\lambda)$ là lớn nhất, với :

$$P(O, Q, \lambda) = \max\{P(q_1, q_2, \dots, q_t = i, o_1, o_2, \dots, o_t | \lambda)\} \quad (2.3)$$

2.2.2. Mô hình ngôn ngữ

Mô hình ngôn ngữ là một tập xác suất phân bố của các đơn vị (thường là từ) trên một tập văn bản cụ thể. Một cách tổng quát thông qua mô hình ngôn ngữ cho phép ta xác định xác suất của một cụm từ hoặc một câu trong một ngôn ngữ. Mô hình ngôn ngữ là một thành phần quan trọng trong hệ thống nhận dạng từ vựng lớn, khi mà tại một thời điểm mô hình âm học có thể xác định ra rất nhiều từ có cùng xác suất. Khi đó mô hình ngôn ngữ sẽ chỉ ra từ chính xác nhất thông qua xác suất của nó trong cả câu đầu ra. Mô hình ngôn ngữ không chỉ giúp bộ giải mã quyết định từ đầu ra đối với mỗi mẫu nhận dạng mà nó còn giúp chuẩn hóa về mặt ngữ pháp cho đầu ra của hệ thống nhận dạng. Mô hình ngôn ngữ có nhiều hướng tiếp cận, nhưng chủ yếu được xây dựng theo mô hình N-gram.

Mô hình n-gram dựa theo công thức Bayes để tính xác suất của một cụm từ gồm L từ “ $w_1 w_2 w_3 \dots w_L$ ” như sau:

$$\begin{aligned} P(w_1 w_2 \dots w_L) \\ = P(w_1) * P(w_2 | w_1) * \dots * P(w_L | w_1 w_2 \dots w_{L-1}) \end{aligned} \quad (2.4)$$

Để giảm độ phức tạp tính toán đối với các cụm từ kích thước lớn, thông thường phương pháp xấp xỉ Markov được áp dụng với giả thiết xác suất xuất hiện của một từ thứ L trong câu chỉ phụ thuộc vào n từ đứng trước nó. Theo giả thiết này công thức (2.8) được viết lại như công thức (2.9). Mô hình ngôn ngữ này gọi là mô hình ngôn ngữ n-gram.

$$P(w_1 w_2 \dots w_L) = P(w_1) * P(w_2 | w_1) * \dots * P(w_L | w_{L-n} w_{L-n+1} \dots w_{L-1}) \quad (2.5)$$

Công thức (2.10) được sử dụng để tính xác suất của từ w_i theo sau cụm từ w_{i-1} :

$$P(w_i | w_{i-1}) = \frac{C(w_{i-1} w_i)}{C(w_i)} \quad (2.6)$$

Trong đó w_{i-1} là một cụm từ có độ dài n bất kỳ nào đó, $C(w_{i-1} w_i)$ là số lần cụm từ $w_{i-1} w_i$, $C(w_i)$ là số lần cụm từ w_i xuất hiện.

Các vấn đề tồn tại của n-gram

- Phân bố không đồng đều: Trong thực tế mô hình n-gram thường được tính toán dựa trên một tập văn bản đầu vào xác định. Các giá trị $(w_{i-1} w_i)$, $C(w_i)$ được xác định hoàn toàn dựa vào tập văn bản này. Như vậy việc một từ w_i hoặc cụm từ $w_{i-1} w_i$ có thể sẽ không xuất hiện hoặc xuất hiện rất ít trong tập văn bản này là hoàn toàn có thể. Điều này dẫn đến giá trị của $C(w_{i-1} w_i)$ có thể bằng không. Tuy nhiên điều này là không đúng trong thực tế vì một văn bản xác định không thể chứa hết tất cả các cụm từ có thể trong một ngôn ngữ. Ngay cả khi một văn bản có thể chứa tất cả các cụm từ $w_{i-1} w_i$ và w_i thì mô hình n-gram lại đánh giá một cụm từ sai ngữ pháp tương đồng với một cụm từ đúng ngữ pháp và xuất hiện với tần suất lớn vì trong công thức (2.10) không phân biệt vị trí hay ngữ pháp của cụm từ $w_{i-1} w_i$.

- Kích thước: Nếu tập văn bản đầu vào có tập từ vựng và có kích thước rất lớn có thể dẫn đến số lượng các cụm $w_{i-1} w_i$ rất lớn, đây là lý do có thể làm gia tăng kích thước lưu trữ mô hình ngôn ngữ trên máy tính và làm giảm tốc độ tìm kiếm của quá trình giải mã.

Một số phương pháp làm tròn mô hình n-gram để khắc phục nhược điểm phân bố không đồng đều:

- **Phương pháp làm mịn Add-One:** Mục đích của phương pháp là chia sẻ xác suất từ các cụm từ xuất hiện nhiều lần sang các cụm từ không xuất hiện hoặc xuất hiện ít bằng cách cộng thêm 1 vào biểu thức tính $p(w_i|w_{i-n}w_{i-n+1}\dots w_{i-1})$ như sau:

$$P(w_i|w_{i-1}) = \frac{C(w_{i-1}w_i) + 1}{C(w_i) + V} \quad (2.7)$$

Trong đó V là kích thước bộ từ vựng.

Phương pháp truy hồi Back-off: Ý tưởng của back-off là nếu như $C(w_{i-n}w_{i-n+1}\dots w_{i-1}w_i) = 0$ thì nó sẽ được thay thế bởi số lần xuất hiện của cụm ngắn hơn $C(w_{i-n}w_{i-n+1}\dots w_{i-1})$. Một cách tổng quát xác suất của cụm từ “ $w_{i-1}w_i$ ” có thể được tính như sau:

$$\begin{aligned} P(w_i|w_{i-n}\dots w_{i-1}) \\ = \begin{cases} P(w_i|w_{i-n}\dots w_{i-1}) & \text{nếu } C(w_{i-n}\dots w_{i-1}, w_i) > 0 \\ \alpha * P(w_{i-n}\dots w_{i-1}) & \text{ngược lại} \end{cases} \end{aligned} \quad (2.8)$$

Trong đó α là một hệ số chọn trước.

Phương pháp nội suy Interpolation: Phương pháp này cũng tính giá trị $P(w_i|w_{i-1})$ dựa trên xác suất của các cụm từ ngắn hơn w_i có mặt. Công thức tổng quát như sau:

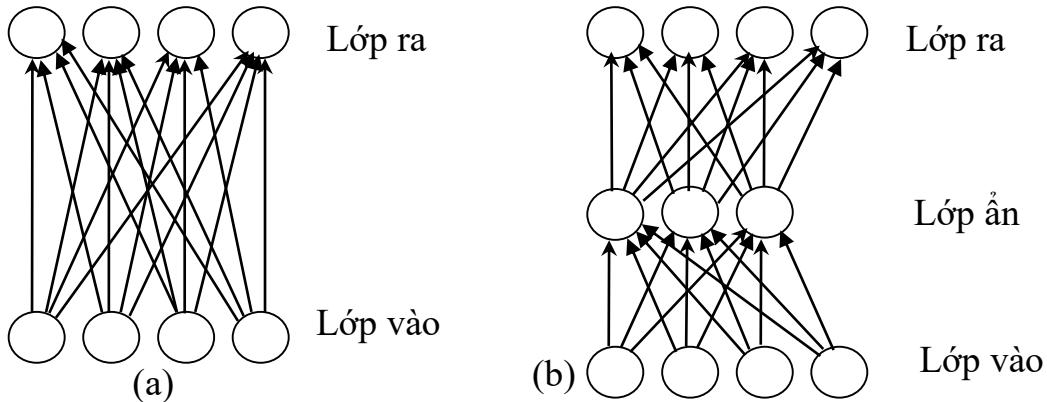
$$\begin{aligned} P(w_i|w_{i-n}\dots w_{i-1}) \\ = \varphi_1 P(w_i|w_{i-n}\dots w_{i-1}) \\ + \varphi_2 P(w_i|w_{i-n}\dots w_{i-2}) \dots + \varphi_n P(w_i) \end{aligned} \quad (2.9)$$

Trong đó tổng các hệ số $\sum_{i=1}^n \varphi_i = 1$.

2.2.3. Mô hình mạng nơ-ron

Mạng nơ-ron cấu trúc Perceptron nhiều lớp như hình 2.2 được sử dụng nhiều trong các hệ thống nhận dạng. Mạng nơ-ron MLP (MultiLayer Perceptron) là một cấu trúc mạng gồm có một lớp vào (input), một lớp ra (output) và một hoặc nhiều lớp ẩn (hidden). Véc-tơ đầu vào sẽ được đưa qua lớp vào (input) của mạng và sau đó các tính toán được thực hiện lan truyền tới

(feed-forward) từ lớp vào input sang các lớp ẩn và kết thúc ở lớp ra (output). Hàm kích hoạt kết hợp với các nốt ẩn hay các nốt ra có thể là hàm tuyến tính hay phi tuyến và có thể khác nhau giữa các nốt. Hình 2.2 mô tả các thành phần cơ bản của một nốt mạng. Hình 2.2 a mô tả cấu trúc của một mạng MLP có 2 lớp và hình 2.2 b mô tả cấu trúc của một mạng có 3 lớp (1 lớp đầu vào, 1 lớp ẩn và 1 lớp ra).



Hình 2.2 Mạng Perceptron. (a) Perceptron 1 lớp, (b) Perceptron nhiều lớp

Xét một mạng MLP có N lớp với kích thước của các lớp tương ứng là $S_1, \dots, S_i, \dots, S_N$. (Trong đó lớp đầu vào là S_1 và lớp đầu ra là S_N). Gọi giá trị kích hoạt của một nốt j trong lớp thứ i là $A_{i,j}$, trọng số của liên kết giữa nó với nốt thứ k trong lớp phía trước $i-1$ là $W_{i,j,k}$, và trọng số của nốt này trong lớp mạng hiện tại là $B_{i,j}$. Khi đó hàm lan truyền thẳng (feed-forward) để xác định giá trị ở lớp ra sẽ được thực hiện lần lượt trên từng lớp theo công thức sau:

$$A_{i,j} = f\left(\sum_{k=1}^{S_{i-1}} A_{i-1,k} * W_{i,j,k} + B_{i,j}\right) \quad (2.10)$$

Trong đó: $i=2, \dots, N$. $j=1, \dots, S_i$. $A_{0,k}=X_k$ là giá trị thứ k trong vector đầu vào. $A_{N,k} = \hat{Y}_{N,k}$ là giá trị lớp ra tại nốt thứ k .

Xét một tập mẫu đầu vào $\{X, Y\}$ trong đó $X=\{X_1, \dots, X_t, \dots, X_T\}$ là giá trị đầu vào, $Y=\{Y_1, \dots, Y_t, \dots, Y_T\}$ là giá trị mong muốn ở lớp ra tương ứng với X . Quá trình huấn luyện mạng là quá trình đi ước lượng các tham số W và B sao cho độ sai lệch giữa Y và \hat{Y} thoả mãn một điều kiện nào đó. Hàm xác định mối quan hệ giữa Y và \hat{Y} gọi là hàm mục tiêu. Hàm mục tiêu thường được sử dụng là hàm bình phương tối thiểu độ lệch giữa Y và \hat{Y} như công thức sau:

$$E = \frac{1}{2} * \sum_{t=1}^T \sum_{k=1}^{S_N} (Y_{t,k} - \hat{Y}_{t,k})^2 \quad (2.11)$$

Trong đó: S_N là kích thước lớp đầu ra, $Y_{t,k}$ là giá trị mong muốn tại nốt thứ k ở lớp đầu ra đối với vector đầu vào X_t , $\hat{Y}_{t,k}$ là giá trị của hàm lan truyền thẳng tại nốt thứ k ở lớp đầu ra đối với véc-tơ đầu vào X_t .

Như vậy mục tiêu của bước huấn luyện mạng là tối thiểu giá trị E trong công thức (2.6). Một trong các phương pháp huấn luyện phổ biến được sử dụng trong huấn luyện mạng MLP là phương pháp lan truyền ngược. Ý tưởng chính của phương pháp tối thiểu giá trị E bằng cách dùng chính E để xác định lại các giá trị trọng số trong công thức (2.6). Quá trình tính toán lại được thực hiện ngược lại từ lớp thứ N đến lớp thứ 2 của mạng theo công thức sau:

$$W_{i,j,k}^q = W_{i,j,k}^{q-1} - \alpha \frac{dE^{q-1}}{dW_{i,j,k}^{q-1}} \quad (2.12)$$

Trong đó: $W_{i,j,k}^q$ là giá trị trọng số của liên kết giữa hai nốt thứ j trong lớp i và nốt thứ k ở lớp $i-1$ tại vòng lặp thứ q , α là hệ số học của mạng (learning rate).

Có hai cách tiếp cận chính trong việc áp dụng mạng nơ-ron cho nhận dạng tiếng nói. Cách tiếp cận thứ nhất là sử dụng mạng nơ-ron như một mô hình âm học có chức năng phân lớp hay nhận dạng mẫu đầu vào. Cách tiếp cận này thường được sử dụng trong các hệ thống nhận dạng với từ vựng nhỏ như các hệ thống điều khiển hoặc tương tác người máy bằng tiếng nói. Khi đó với mỗi một vector đặc trưng đầu vào đưa qua mạng ta sẽ thu được ở đầu ra một quyết định tương ứng. Cách tiếp cận thứ hai là kết hợp mô hình HMM và GMM làm mô hình âm học trong các hệ thống nhận dạng từ vựng lớn. Trong cách tiếp cận này hàm xác suất phát tán được thay bằng hàm kích hoạt ở lớp đầu ra của mạng nơ-ron thay vì là hàm GMM như cách truyền thống.

2.2.4. Mạng học sâu

Mạng nơ-ron sâu (Deep Neural Network- DNN) thực chất là một mạng nơ-ron truyền tải có nhiều lớp ẩn, trong đó mỗi lớp ẩn có một số nơ-ron nhất

định, dữ liệu đầu vào của mỗi lớp là tất cả các kết quả đầu ra của lớp trước được nhân với một vectơ trọng số, tính kết quả và chuyển nó qua một hàm kích hoạt phi tuyến tính như sigmoid hoặc tanh như công thức 2.11. Trong mô hình học sâu có 3 loại mạng được sử dụng nhiều trong lĩnh vực thị giác máy và nhận dạng tiếng nói đó là bộ tự mã hóa, mạng hồi quy và mạng tích chập.

Bộ tự mã hóa

Bộ tự mã hóa (Auto-Encoders) là một mô hình cụ thể của mạng truyền tới nhiều lớp với đầu vào cũng là đầu ra. Thông thường, bộ tự mã hóa được sử dụng trong bài toán nén dữ liệu. Vì thế, bộ tự giải mã sẽ có các lớp ẩn với số nơ-ron ít hơn số nơ-ron ở lớp đầu vào. Bộ tự mã hóa sẽ nén dữ liệu đầu vào vào một bộ mã có số chiều ít hơn và sau đó tái tạo lại dữ liệu đầu ra từ bộ mã biểu diễn này. Bộ mã này được gọi là bản tóm tắt hoặc bản nén của dữ liệu đầu vào, và bộ mã này cũng được gọi là biểu diễn của dữ liệu trong không gian tiềm ẩn.

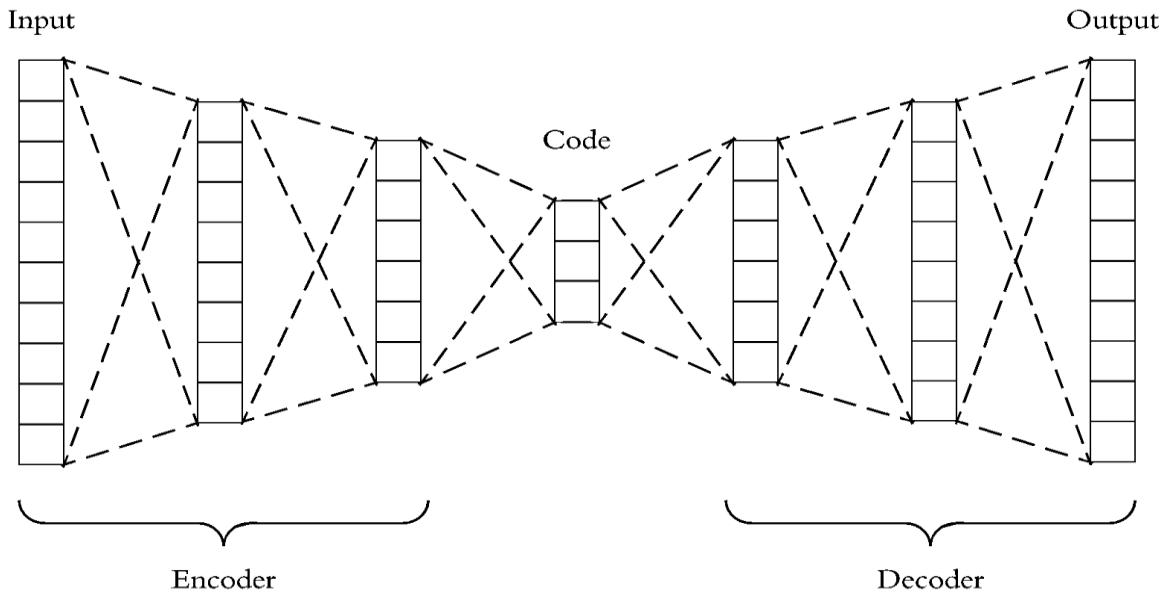
Thông thường, một bộ tự mã hóa bao gồm 3 thành phần đó là: bộ mã hóa, mã và bộ giải mã. Bộ mã hóa nén dữ liệu đầu vào và tạo ra bộ mã, bộ giải mã sau đó tái tạo lại dữ liệu đầu vào chỉ dựa trên bộ mã này.

Bộ tự giải mã thường được sử dụng như một phương pháp rút gọn chiều dữ liệu, hay phương pháp nén dữ liệu. Bộ tự mã hóa có một số đặc tính quan trọng sau:

- Tính cụ thể: Bộ tự mã hóa chỉ có thể mã hóa được dữ liệu có ý nghĩa tương tự như dữ liệu mà chúng đã được huấn luyện. Vì vậy, không thể sử dụng bộ tự mã hóa được huấn luyện bởi các chữ số viết tay để nén ảnh phong cảnh.

- Tính mất thông tin: Kết quả đầu ra của bộ tự mã hóa sẽ không chính xác giống như dữ liệu đầu vào, nó chỉ là một biểu diễn gần đúng của dữ liệu đầu vào.

- Tính không giám sát: Bộ tự mã hóa không cần phải gán nhãn dữ liệu khi huấn luyện. Vì vậy, bộ tự mã hóa được coi là một kỹ thuật học không giám sát.



Hình 2. 3 Mô hình bộ tự mã hóa²

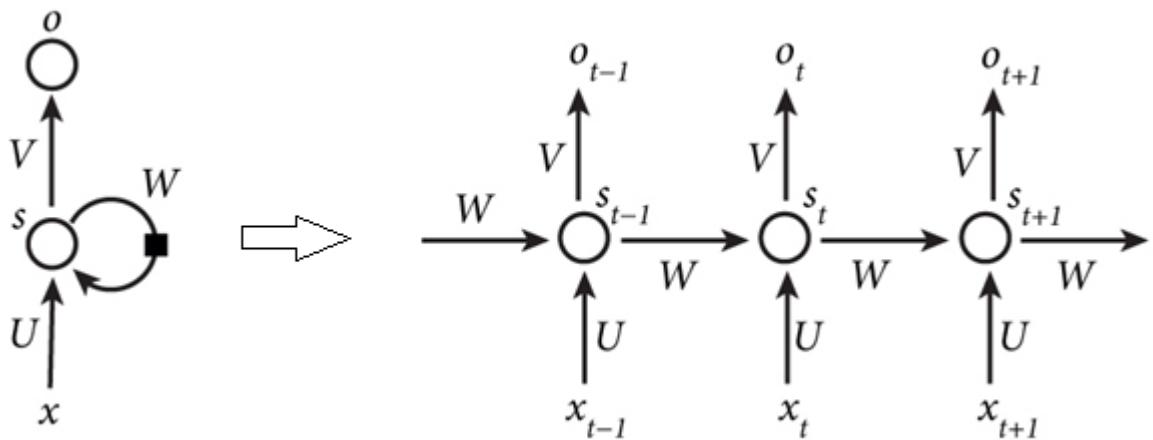
Mạng hồi quy

Mạng hồi quy (Recurrent Neural Networks - RNN) có ý tưởng chính là sử dụng chuỗi các thông tin, khác với các mạng nơ-ron truyền thống tất cả các đầu vào và cả đầu ra là độc lập với nhau, nghĩa là chúng không liên kết thành chuỗi với nhau. Các mô hình này không phù hợp với các bài toán mà thông tin có mối liên hệ thứ tự với nhau như trong xử lý ngôn ngữ tự nhiên, nhận dạng tiếng nói. Trong khi đó, mạng hồi quy thực hiện cùng một tác vụ cho tất cả các phần tử của một chuỗi với đầu ra phụ thuộc vào cả các phép tính trước đó. Nói cách khác, mạng hồi quy có khả năng nhớ các thông tin được tính toán trước đó.

Mạng hồi quy được sử dụng nhiều trong lĩnh vực xử lý ngôn ngữ tự nhiên. Trong lĩnh vực nhận thức tiếng nói, mạng hồi quy sâu đã được một số tác giả sử dụng thành công và cho kết quả cao như Alex sử dụng mạng hồi quy sâu trên bộ dữ liệu TIMIT đạt được sai số là 17.7% [Graves, 2013].

Về cơ bản một mạng hồi quy có dạng như sau:

² <https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798>



Hình 2. 4 Mô hình mạng hồi quy³

Trong đó x_t là đầu vào, s_t là trạng thái ẩn, o_t là đầu ra tại bước t .

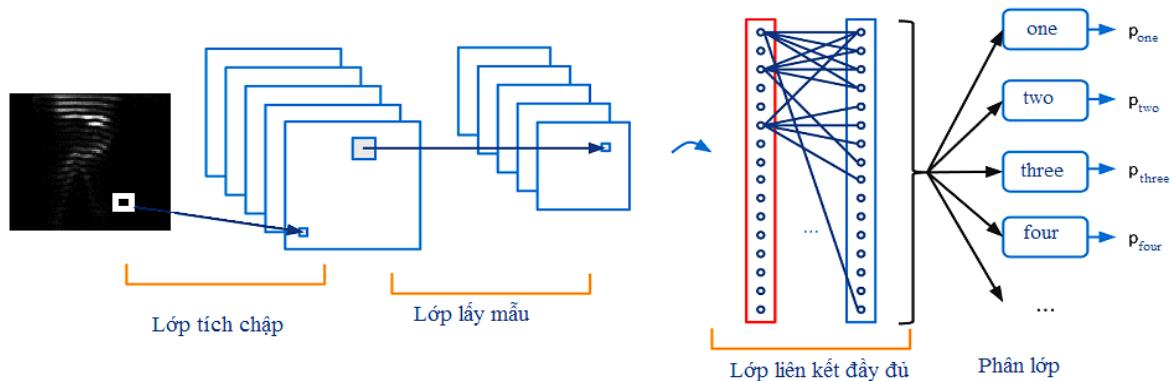
Mô hình mạng tích chập

Mô hình mạng nơ-ron truyền tín hiệu được áp dụng nhiều trong các bài toán nhận dạng. Tuy nhiên mạng nơ-ron truyền tín hiệu không thể hiện tốt đối với các dữ liệu như hình ảnh do sự liên kết quá dày đặc giữa các lớp trong mạng truyền tín hiệu. Dữ liệu hình ảnh thường có kích thước lớn, nếu một tấm ảnh đa cấp xám có kích thước 32×32 giá trị, sẽ cho ra vector đặc trưng có 1024 chiều. Điều này cũng có nghĩa là cần tới 1024 trọng số liên kết giữa lớp đầu vào với một nốt ở lớp ẩn kế tiếp. Số lượng trọng số sẽ càng tăng nhanh nếu số lượng nốt trong lớp ẩn tăng lên và số lượng lớp ẩn tăng lên. Điều này khiến cho việc thao tác với các ảnh có kích thước lớn hơn trở nên khó khăn. Mặt khác, việc liên kết một cách dày đặc các điểm ảnh vào một nốt trong mạng có sẽ tạo ra dư thừa vì sự phụ thuộc lẫn nhau giữa các điểm ảnh xa nhau là không nhiều mà chủ yếu là sự phụ thuộc giữa các điểm lân cận với nó. Dựa trên tư tưởng này Lecun [Lecun, 1998] đã đề xuất mô hình mạng nơ-ron tích chập (Convolutional Neural Network) cho bài toán nhận dạng ảnh. Trong mô hình này, thay vì toàn bộ điểm ảnh được nối với các nốt mạng thì chỉ có một phần cục bộ trong ảnh được nối đến các nốt mạng ở lớp sau. Thông qua các lớp, mô hình sẽ học được các đặc trưng để tiến hành phân lớp một cách hiệu quả. Thông thường, mô hình mạng nơ-ron tích chập bao gồm các lớp sau: lớp tích chập (Convolution layer), lớp lấy mẫu (Pooling layer) và lớp kết nối dày đặc (Fully connected). Sự sắp xếp về

³ <https://www.analyticsvidhya.com/blog/2017/12/introduction-to-recurrent-neural-networks/>

số lượng và thứ tự giữa các lớp này sẽ tạo ra những mô hình khác nhau phù hợp cho các bài toán khác nhau.

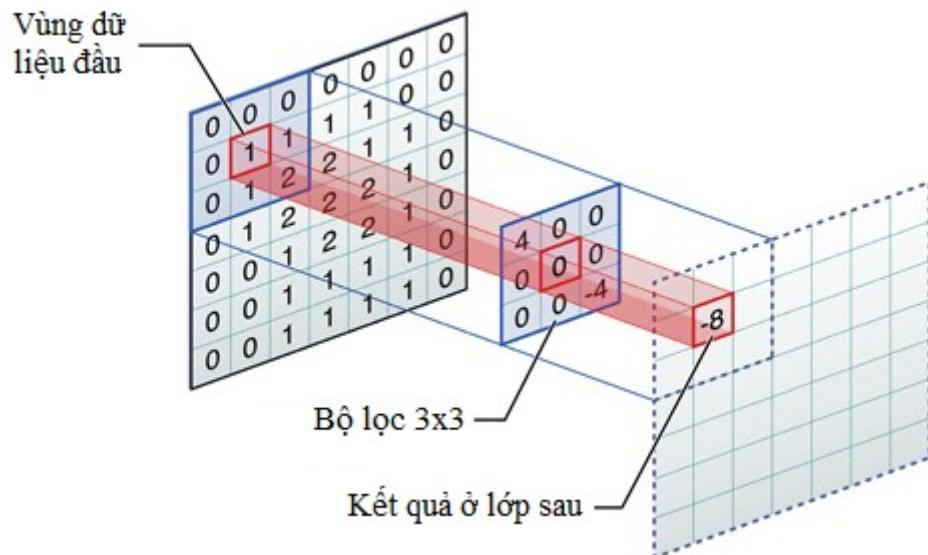
Mạng nơ-ron tích chập được áp dụng chủ yếu trong lĩnh vực thị giác máy, cụ thể trong các bài toán nhận dạng như nhận dạng vật thể trong ảnh [Ren, 2016] [Kim, 2016], nhận dạng chữ viết tay [Xu-Yao, 2017], nhận dạng vật thể 3D [Alexandre, 2016] [Xiaofan, 2016], nhận dạng khuôn mặt [Syaffeza, 2014] [Guo, 2017], ứng dụng trong y học [Li, 2016] [Wahab, April 2017] [Kleesiek, 2016]. Bên cạnh đó, mạng tích chập cũng được áp dụng và đạt được kết quả cao trong xử lý tiếng nói [Abdel-Hamid, 2014] cũng như trong xử lý ngôn ngữ tự nhiên [Yin, 2017] [Hang, 2018].



Hình 2. 5 Mô hình mạng tích chập CNN

Lớp tích chập

Lớp tích chập (Convolution layer) là tư tưởng cốt lõi của mạng nơ-ron tích chập. Thay vì kết nối toàn bộ điểm ảnh ở lớp đầu vào, lớp tích chập sử dụng các bộ lọc khác nhau có kích thước nhỏ (thường là 3×3 hoặc 5×5) tích chập với từng vùng trong ảnh để thu được một đặc trưng của các điểm ảnh trong vùng cục bộ đó. Bộ lọc sẽ lần lượt được trượt toàn bộ ảnh theo một bước nhất định.



Hình 2. 6 Tích chập một bộ lọc với dữ liệu đầu vào⁴

Hình 2.6 là ví dụ một bộ lọc 3×3 với dữ liệu đầu vào có kích thước 32×32 , ta sẽ có kết quả là một ma trận dữ liệu mới với mỗi giá trị là kết quả của phép tích chập của bộ lọc với một vùng dữ liệu cục bộ tương ứng trên dữ liệu gốc. Lớp này có bao nhiêu bộ lọc thì sẽ thu được bấy nhiêu ma trận kết quả tương ứng mà lớp này trả về và được truyền cho lớp tiếp theo. Ban đầu trọng số của các bộ lọc được khởi tạo ngẫu nhiên, các trọng số này sẽ được học sẽ được học khi huấn luyện mô hình.

Lớp phi tuyến Relu (Rectified linear unit)

Giả sử mạng tích chập có L lớp có lớp, trong đó lớp đầu vào (input) là lớp thứ 0. Khi đó mạng tích chập sẽ có L ma trận trọng số được ký hiệu là

$$W^l \in R^{d^{(l-1)} \times d^l} \text{ với } l = 1, 2, \dots$$

Trong đó W^l là các kết nối từ lớp thứ $(l-1)$ đến lớp thứ l , phần tử w_{ij}^l thể hiện kết nối của nơ-ron thứ i của lớp $(l-1)$ đến nơ-ron thứ j của lớp l . Các hệ số nhiễu (bias) thứ (l) được ký hiệu là $b^l \in R^{d^l}$. Để thực hiện phân lớp có kết quả tối ưu là quá trình đi tìm bộ tham số w và b . Mỗi nơ-ron không phải lớp đầu vào được tính bằng công thức:

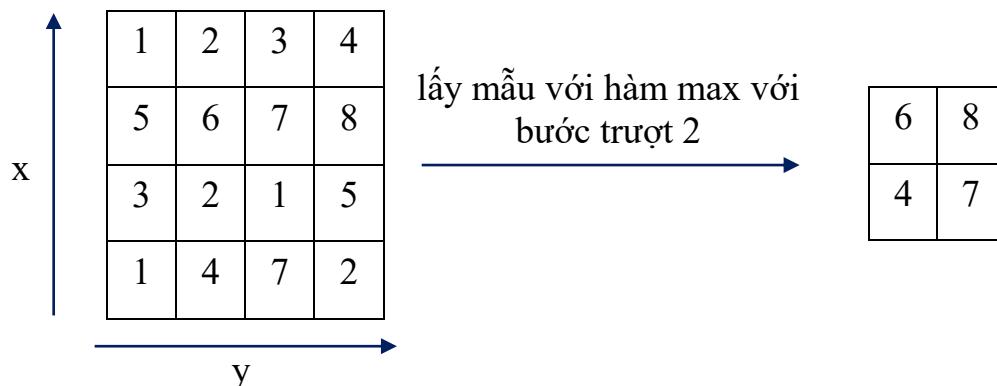
$$a_i^l = f((w_i^l)^T a^{l-1} + b_i^l)$$

⁴ <https://medium.freecodecamp.org/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050>

Trong đó $(w)^T$ là ma trận chuyển vị của ma trận w , f là một hàm kích hoạt phi tuyến được áp dụng cho một ma trận. Trong các mạng tích chập, người ta thường sử dụng hàm kích hoạt là hàm $f(x) = \max(0, x)$ chuyển toàn bộ giá trị âm trong kết quả lấy từ lớp tích chập thành giá trị 0 để tạo tính phi tuyến cho mô hình gọi là Relu. Ngoài ra còn có nhiều hàm kích hoạt khác như signmod, tanh. Tuy nhiên, hàm RELU được cho là dễ cài đặt tính toán nhanh và hiệu quả hơn [Krizhevsky, 2012].

Lớp lấy mẫu: Lớp lấy mẫu (Pooling layer) sử dụng một cửa sổ trượt quét qua toàn bộ dữ liệu, mỗi lần trượt theo một bước cho trước. Khi cửa sổ trượt trên dữ liệu, nó chỉ giữ lại một giá trị được xem là đại diện cho vùng dữ liệu đó. Các phương thức lấy mẫu phổ biến là lấy giá trị lớn nhất (max), lấy giá trị nhỏ nhất (min), lấy giá trị trung bình (average).

Lớp lấy mẫu có vai trò giảm kích thước dữ liệu nhưng vẫn giữ được những đặc trưng cần thiết cho việc nhận dạng từ đó làm giảm số lượng tham số cần học, làm tăng hiệu quả tính toán và tránh hiện tượng quá khớp trong học máy.



Hình 2.7 Ví dụ lấy mẫu với hàm max

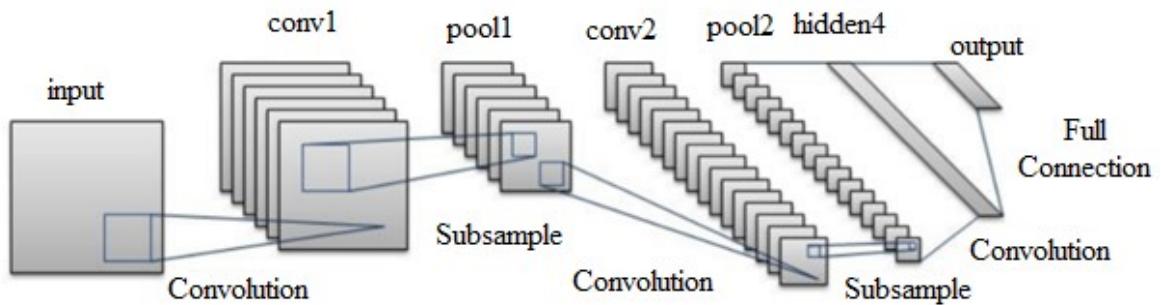
Hình 2.7 là một ví dụ của hàm lấy mẫu với kích thước dữ liệu ban đầu là 4×4 , sử dụng bộ lọc lấy mẫu kích thước 2×2 với phương pháp lấy mẫu là lớn nhất (max), với bước trượt 2 sẽ thu được dữ liệu có kích thước 2×2 , giảm một nửa so với dữ liệu ban đầu.

Lớp kết nối đầy đủ: Lớp kết nối đầy đủ (**Fully connected - FC**) tương tự với các lớp trong mạng nơ-ron truyền tới, các giá trị của các nốt ở lớp này được liên kết đầy đủ tới các nốt ở lớp tiếp theo. Trong mạng tích chập, sau khi

trích chọn được đặc trưng của dữ liệu thông qua các lớp tích chập và lấy mẫu, dữ liệu sẽ được kết nối đầy đủ tới lớp đầu ra để tiến hành phân lớp.

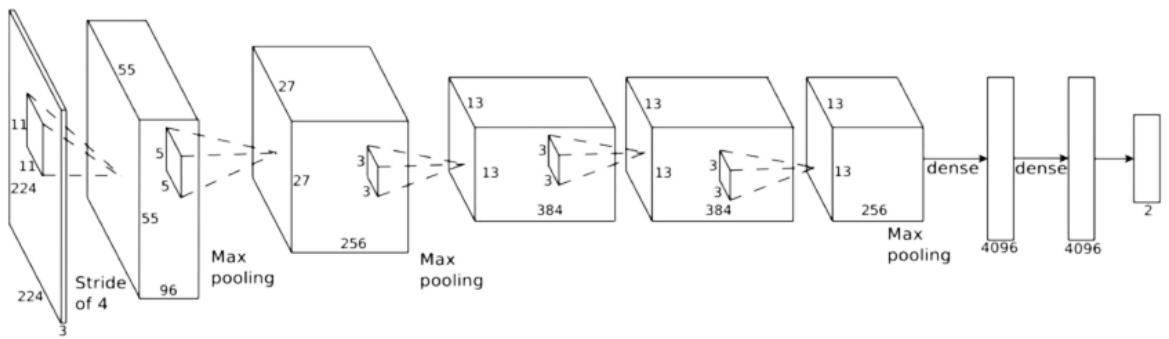
Một số cấu trúc CNN được sử dụng thành công hiện nay gồm:

- LeNet5: là mô hình mạng tích chập được đề xuất bởi Lecun vào năm 1998 được sử dụng để nhận dạng chữ số, ký tự trong văn bản với dữ liệu đầu vào là các ảnh đa cấp xám có kích thước 32x32px. Không tính lớp vào và lớp ra, mạng LeNet5 có 5 lớp với tổng số 60.000 tham số [Lecun, 1998].



Hình 2. 8 Mô hình mạng tích chập LeNet 5 [Lecun, 1998]

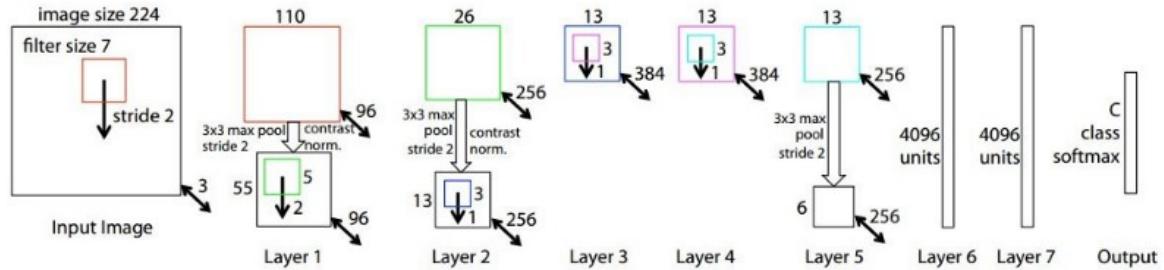
- AlexNet (2012): Được phát triển bởi Alex Krizhevsky và các đồng nghiệp. Lần đầu AlexNet được giới thiệu vào năm 2012 với cấu trúc khá tương tự như LeNet nhưng với số lượng neuron, filter và layer lớn hơn và số lượng tham số lớn hơn với 60 triệu tham số. Mạng AlexNet được coi là mạng tích chập đầu tiên phổ biến rộng rãi khả năng của mạng tích chập [Krizhevsky, 2012].



Hình 2. 9 Mô hình mạng tích chập AlexNet [Krizhevsky, 2012]

- ZF Net: Là mạng tích chập được đánh giá là tốt nhất năm 2013. Mạng ZF Net được phát triển bởi Matthew Zeiler và Rob Fergus. Mạng tích chập này được phát triển từ mạng AlexNet bằng việc tinh chỉnh các siêu tham số của mạng AlexNet như điều chỉnh kích thước các bộ lọc, bước trượt của bộ lọc

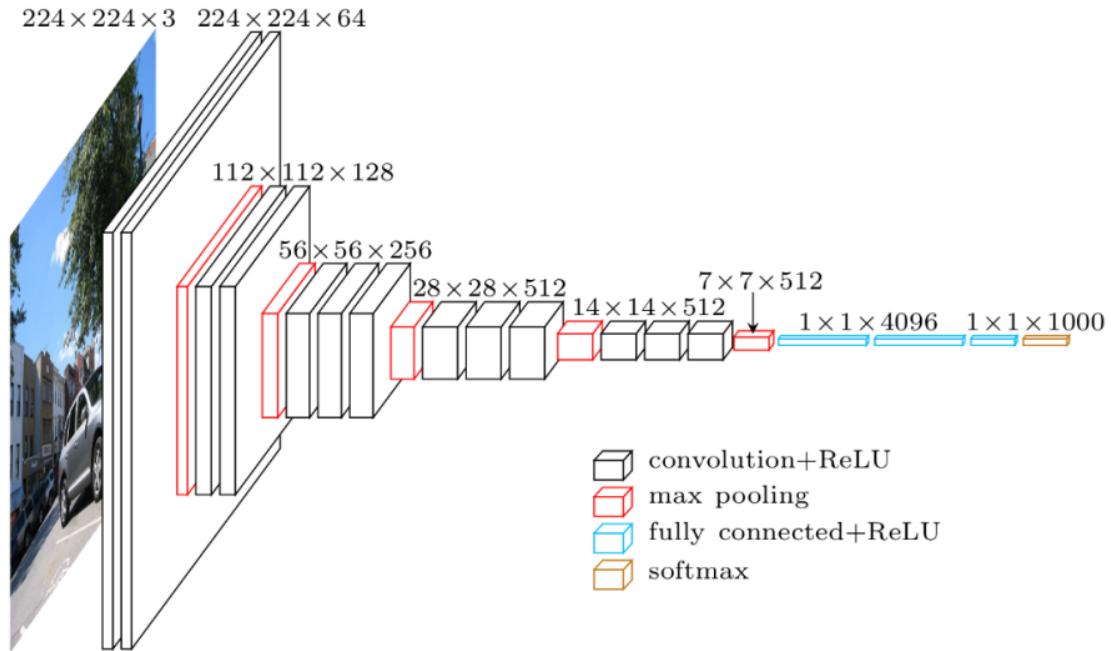
(filter size, stride,...). Với những lớp gần lớp đầu vào thì kích thước của các bộ lọc và bước trượt nhỏ hơn [Zeiler, 2014] .



Hình 2. 10 Mô hình mạng ZF Net [Zeiler, 2014]

– GoogLeNet (2014): Là mạng tích chập tốt nhất năm 2014 được phát triển bởi Szegedy từ Google. Mạng có 22 lớp tích chập, với một số cải tiến như giảm thiểu số lượng tham số trong mạng AlexNet từ 60 triệu xuống còn 4 triệu, sử dụng bộ lấy mẫu trung bình thay cho lớp kết nối đầy đủ [Christian, 2015] [Szegedy, 2016] .

– VGGNet (2014): Là mạng mạng tích chập tốt nhất năm 2015 được phát triển bởi Karen Simonyan và Andrew Zisserman. Mạng VGGNet có 16 lớp, sử dụng bộ lọc có kích thước 3×3 và bộ lấy mẫu có kích thước 2×2 cho tất cả các lớp của mạng [Simonyan, 2014] . Mạng VGGNET có tổng số 138 triệu tham số.



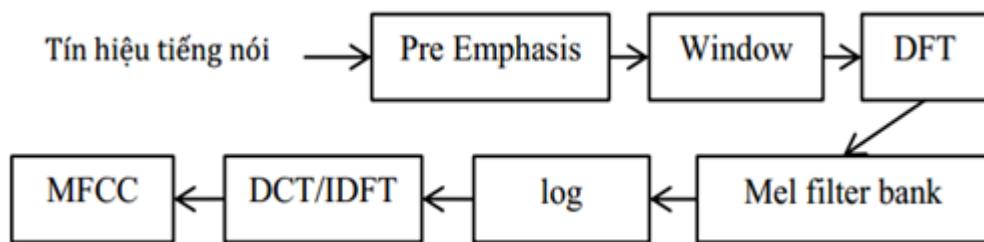
Hình 2. 11 Mô hình mạng tích chập VGGNET [Simonyan, 2014]

– ResNet (2015): Là mạng tích chập tốt nhất năm 2015 được phát triển bởi Kaiming He và các đồng nghiệp. Mạng ResNet bỏ qua lớp kết nối đầy đủ ở cuối mạng, và sử dụng cơ chế “skip connection” và “batch normalization”. Mạng ResNet vẫn được tiếp tục phát triển, với kết quả gần nhất được phát hành vào tháng 3 năm 2016 [Kaiming, 2016] [He, 2016]. Mạng ResNet có tổng số 152 lớp.

2.3. Trích chọn đặc trưng tiếng nói cho các mô hình học máy

2.3.1. Đặc trưng MFCC

MFCC là một trong những loại đặc trưng được sử dụng phổ biến trong bài toán nhận thức tiếng nói. Ý tưởng chính của MFCC là tính toán các giá trị phổ của tín hiệu cho băng tần trên miền tần số mà tai người dễ cảm thụ nhất.



Hình 2. 12 Sơ đồ khái quát các bước trích chọn đặc trưng MFCC

Trong đó:

- **Pre Emphasis:** Tai người chỉ nhạy cảm với các tần số thấp nên một hàm tăng cường tín hiệu theo công thức (2.14) cho các tần số cao được áp dụng trước khi tín hiệu được đưa vào tính toán ở các bước sau:

$$s(n) = x(n) - \alpha * x(n - 1) \quad (2.13)$$

Trong đó $x(n)$ là tín hiệu vào, α là hệ số

Window: Tạo các khung tín hiệu gọi là cửa sổ. Tín hiệu tiếng nói là loại tín hiệu liên tục và biến đổi theo thời gian. Tuy nhiên trong một khoảng thời gian ngắn từ 10ms đến 30ms có thể được coi là ổn định. Đối với các hệ thống nhận dạng từ vựng lớn phát âm liên tục thì đơn vị nhận dạng thường là một âm vị và độ dài phát âm của một âm vị cũng thường nằm trong khoảng thời gian này. Vì thế thay vì ta đi tính toán đặc trưng trên toàn bộ một phát âm thì ta chỉ tính toán trên từng khung cửa sổ có độ dài từ 10ms đến 30ms. Để không bị mất thông tin giữa hai khung liên tiếp thì các cửa sổ thường được xếp chồng lên

nhau với khoảng cách từ 10ms đến 20ms. Hàm cửa sổ áp lên mỗi khung thường là hàm Hamming với công thức sau:

$$W(n) = \left\{ 0.54 - 0.46 \cos\left(\frac{2\pi n}{L}\right) \right\} \quad (2.14)$$

Khi đó giá trị của tín hiệu sau khi áp dụng hàm cửa sổ là:

$$y(n) = W(n)S(n)$$

Trong đó L là kích thước của cửa sổ, $0 \leq n \leq L$, $s(n)$ giá trị của tín hiệu ở miền thời gian tại thời điểm n .

DFT: Biến đổi Fourier rời rạc. Biến đổi DFT được áp dụng để trích chọn thông tin về phổ tần số của tín hiệu đầu vào. Phép biến đổi này được thực hiện trên mỗi một khung đã được lấy qua hàm cửa sổ. Tính toán DFT được mô tả ở công thức sau:

$$X(k) = \sum_{n=0}^{L-1} y[n]e^{-j2\frac{\pi}{L}kn} \quad (2.15)$$

Trong đó: L là kích thước của cửa sổ, $y[n]$ giá trị của tín hiệu đầu vào sau khi qua hàm cửa sổ.

Mel Filter bank: Lọc và biến đổi sang tần số Mel. Tần số tiếng nói thường dao động trong khoảng dưới 10 kHz, tuy nhiên tai người chỉ nhạy cảm hay nghe rõ nhất trong khoảng 1 kHz. Các hệ thống nhận dạng có gắng mô phỏng lại cách thức nghe của con người vì thế vấn đề đặt ra là cần biến đổi tín hiệu từ miền tần số sang miền tần số mà con người dễ nghe nhất. Miền tần số này gọi là Mel (được đặt tên bởi Steven and Volkmann, 1940). Công thức biến đổi được mô tả ở công thức (2.17).

$$mel(f) = 2595 \log_{10}(1 + \frac{f}{700}) \quad (2.16)$$

Các bộ lọc băng tần được thiết kế trên miền tần số Mel này

Logarithm (log) và biến đổi Cosine rời rạc (DCT): Hàm logarithm được áp dụng trên các giá trị DFT đo độ thính của tai người theo hàm logarithm,

vì vậy việc áp dụng hàm log để đưa đặc trưng tính toán được gần giống với tín hiệu mà tai người nghe. Đồng thời việc sử dụng hàm log giúp cho đặc trưng tính toán ít bị ảnh hưởng bởi sự biến đổi ngẫu nhiên ở tín hiệu đầu vào. Sau đó các giá trị logarithm này được áp dụng hàm biến đổi Fourier ngược (hoặc có thể dùng công thức biến đổi Cosine rời rạc) như công thức (2.18) để thu được các giá trị MFCC.

$$C[k] = \sum_{n=0}^{L-1} \log(|X[n]|) e^{j\frac{2\pi}{L}kn} \quad (2.17)$$

2.3.2. Phương pháp mã dự đoán tuyến tính LPC

Phương pháp trích chọn đặc trưng mã dự báo tuyến tính LPC được sử dụng để trích chọn các tham số đặc trưng của tín hiệu tiếng nói [Kinsner, 1988]. Bản chất của phương pháp này là một mẫu tiếng nói được biểu diễn xấp xỉ bởi một tổ hợp tuyến tính của các mẫu trước đó. Thông qua việc tối thiểu hóa tổng bình phương sai số giữa các mẫu hiện tại với các mẫu dự đoán để xác định được một tập duy nhất các hệ số dự báo. Các hệ số dự báo này là các trọng số được sử dụng trong tổ hợp tuyến tính. Với dãy tín hiệu tiếng nói $s(n)$ giá trị dự báo được xác định bởi công thức:

$$\tilde{s}(n) = \sum_{k=1}^P a_k s(n-k) \quad (2.18)$$

Trong đó a_k là các hệ số đặc trưng cho hệ thống.

Hàm sai số dự báo được tính theo công thức:

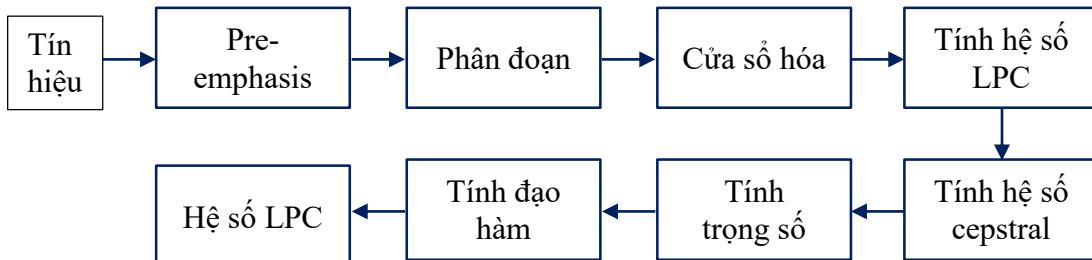
$$e(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^P a_k s(n-k) \quad (2.19)$$

Khi đó bài toán trở thành bài toán tìm tập giá trị $\{a_k\}$ phù hợp nhất để cực tiểu hóa hàm lỗi. Do tín hiệu tiếng nói thay đổi theo thời gian nên các hệ số dự báo phải được ước lượng từ các đoạn tín hiệu ngắn. Hàm lỗi dự báo trong một thời gian ngắn xác định bởi công thức sau:

$$E(n) = \sum_m e_m^2(n) = \sum_m (s_n(m) - \sum_{k=1}^P \alpha_k s_n(m-k))^2 \quad (2.20)$$

trong đó $s_n(m)$ là một đoạn tín hiệu tiếng nói lân cận mẫu thứ n.

Sơ đồ khôi bộ trích chọn các tham số đặc trưng LPC của tín hiệu tiếng nói gồm các bước thực hiện cụ thể như sau:



Hình 2. 13 Sơ đồ trích chọn đặc trưng LPC

Bước 1: Pre-emphasis, sử dụng bộ lọc thông cao có đáp ứng xung theo công thức 2.14

Bước 2: Phân đoạn thành các frame (frame này khác với các frame trong giai đoạn tìm điểm đầu điểm cuối), mỗi frame có N mẫu, độ chồng lấp M mẫu, thường $M = N/2$.

Bước 3: Cửa sổ hóa. Hàm cửa sổ thông dụng nhất là cửa sổ Hamming được định nghĩa như sau:

$$w(n) = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2n\pi}{M}\right) & \text{với } 0 \leq n \leq M \\ 0 & n \notin [0, M] \end{cases} \quad (2.21)$$

Bước 4: Xác định các hệ số dự báo tuyến tính dùng thuật toán Levinson-Durbin.

Bước 5: Chuyển các hệ số dự báo tuyến tính thành các hệ số cepstral.

$$c_m = \begin{cases} a_m + \frac{1}{m} \sum_{k=1}^{m-1} k c_k a_{m-k} & \text{với } 1 \leq m \leq P \\ \frac{1}{k} \sum_{k=1}^{m-1} k c_k a_{m-k} & n \notin [0, M] \end{cases} \quad (2.22)$$

Các hệ số cepstral này có độ tập trung cao hơn và đáng tin cậy hơn so với các hệ số dự báo tuyến tính. Thông thường chọn $Q = 3/2P$.

Bước 6: Chuyển sang cepstral có trọng số:

$$c'_m = w_m c_m \text{ với } 1 \leq m \leq Q \quad (2.23)$$

Hàm trọng số thích hợp là bộ lọc thông dải (trong miền cepstral)

$$w_m = \left[1 + \frac{Q}{2} \sin\left(\frac{\pi m}{Q}\right) \right] \text{ với } 1 \leq m \leq Q \quad (2.24)$$

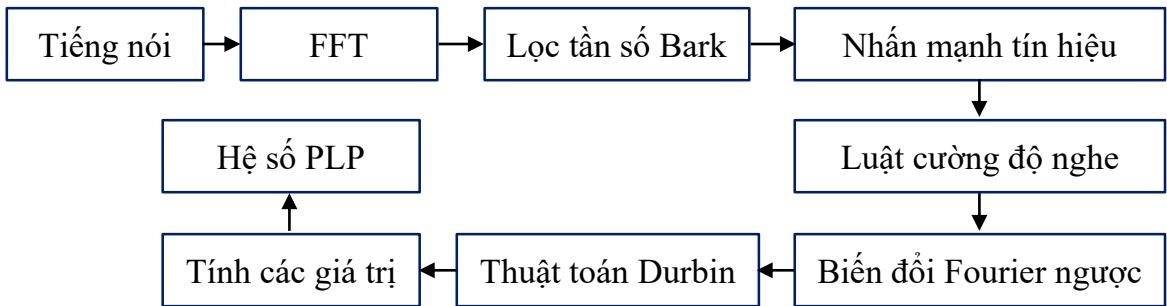
Bước 7: Tính đạo hàm cepstral

$$\frac{dc_m(t)}{dt} = \Delta c_m(t) \approx \mu \sum_{k=-K}^K k c_m(t+k) \quad (2.25)$$

với μ là hằng số chuẩn và $(2K+1)$ là số lượng frame cần tính. $K=3$ là giá trị thích hợp để tính đạo hàm cấp một. Vectơ đặc trưng của tín hiệu gồm Q hệ số cepstral và Q hệ số đạo hàm cepstral.

2.3.3. Đặc trưng PLP

Phương pháp trích chọn đặc trưng PLP dựa trên cơ sở phương pháp mã dự báo tuyến tính LPC. Đặc trưng này được tạo ra dựa trên đặc tính vật lý của tai người khi nghe [Hermansky, 1990].



Hình 2. 14 Sơ đồ khái quát các bước trích chọn đặc trưng PLP

Biến đổi Fourier nhanh (FFT): Tương tự như phương pháp MFCC, tín hiệu tiếng nói được chia thành các khung và được chuyển sang miền tần số bằng thuật toán FFT.

Lọc theo thang tần số Bark: Tín hiệu tiếng nói được lọc qua các bộ lọc phân bố theo thang tần số phi tuyến, trong trường hợp này là thang tần số Bark:

$$Bark(f) = 6 \ln \left\{ \frac{f}{1200} + [(\frac{f}{1200})^2 + 1]^{\frac{1}{2}} \right\} \quad (2.26)$$

Nhấn mạnh tín hiệu: dùng hàm cân bằng độ ồn (equal-loudness). Bước này tương tự bước nhấn mạnh (preemphais) của phương pháp MFCC. Hàm này mô phỏng đường cong cân bằng độ ồn (Equal-Loudness Curve)

$$E(\omega) = \frac{(\omega^2 + 56.8 * 10^6)\omega^4}{(\omega^2 + 6.3 * 10^6)(\omega^6 + 9.58 * 10^{26})} \quad (2.27)$$

Dùng luật cường độ nghe (Power Law of Hearing): Bước xử lý này giống như bước lấy giá trị logarit trong phương pháp MFCC. Hàm căn lập phương được dùng có dạng:

$$\Phi(f) = \Psi(f)^{0.33} \quad (2.28)$$

Biến đổi Fourier ngược (Inverse DFT): Các hệ số tự tương quan được biến đổi Fourier ngược là giá trị đầu vào cho LPC.

Thuật toán Durbin: Thuật toán Durbin được sử dụng để tính các hệ số dự báo tuyến tính như phương pháp LPC

Tính các giá trị delta: Phương pháp tính tương tự như phương pháp hệ số MFCC.

2.4. Kết luận

Chương này chúng tôi giới thiệu một số kiến thức cơ sở, các hướng tiếp cận học máy chủ yếu cho bài toán nhận thức tiếng nói như mô hình HMM, mô hình ngôn ngữ, mô hình mạng nơ-ron, đặc biệt là mạng học sâu. Trong việc mô phỏng quá trình nhận thức tiếng nói, hầu hết các mô hình học máy phải tiến hành thực hiện trích chọn đặc trưng tiếng nói. Chương này, cũng giới thiệu ba hướng tiếp cận chính cho việc trích chọn đặc trưng tiếng nói đó là MFCC, PLC và PLP.

Chương 3. HƯỚNG TIẾP CẬN DỰA TRÊN PHỔ TẦN SỐ CHO BÀI TOÁN NHẬN THỨC TIẾNG NÓI TRONG MÔI LIÊN HỆ VỚI CÁC KHÁI NIỆM

3.1. Giới thiệu

Các mô hình học máy cho bài toán nhận thức tiếng nói hiện nay hầu hết là sử dụng các đặc trưng tiếng nói dựa trên hai loại đặc trưng cơ bản là Mel-frequency cepstral coefficients (MFCC) [Davis, 1980] , PLC và Perceptual Linear Prediction (PLP) [Hermansky, 1990] . Ba loại đặc trưng này sử dụng các bộ lọc tần số dựa trên giả thuyết về tai người chỉ nhận thức được ở một số giải tần số nhất định [Majeed, 2015] . Điều này, dẫn tới làm mất đi một phần thông tin của tín hiệu tiếng nói.

Để trích được đặc trưng MFCC, PLC hay PLP từ tín hiệu tiếng nói, người ta phải chia tín hiệu tiếng nói thành các đoạn ngắn đều nhau để đảm bảo sự ổn định của tín hiệu trong việc trích chọn các phổ tần số của tín hiệu tiếng nói, trong khi tín hiệu tiếng nó của cùng một đơn vị tiếng nói lại có độ dài khác nhau tùy thuộc vào người nói, ngữ cảnh nói. Vì vậy, mỗi tín hiệu tiếng nói sẽ thu được một số lượng các véc tơ đặc trưng khác nhau. Mặt khác, hầu hết các mô hình học máy phổ biến cho bài toán nhận thức tiếng nói như HMM, SVM,... đòi hỏi dữ liệu phải có cùng kích thước giống nhau. Do đó, người ta phải thực hiện biến đổi [Francois, 2007] (như lấy mẫu lại, lượng tử hóa, phân cụm,...) tập các véc tơ đặc trưng ban đầu này thành một véc tơ đặc trưng khác sao cho chúng có cùng kích thước. Nghĩa là, mỗi tín hiệu tiếng nói sẽ được biểu diễn thành một véc tơ đặc trưng mới dựa trên các véc tơ đặc trưng thu được từ MFCC, hay PLP. Điều này, một lần nữa lại làm mất thông tin của tín hiệu tiếng nói. Hơn nữa, đặc trưng MFCC và PLP rất nhạy cảm với nhiễu và thiếu thông tin về pha [Majeed, 2015] .

Trong chương này, chúng tôi sẽ đề xuất trích chọn đặc trưng cho bài toán nhận thức tiếng nói dựa trên phổ tần số của tín hiệu tiếng nói. Hướng tiếp cận dựa trên phổ tần số của tín hiệu tiếng nói đã được một số tác giả đề xuất trong bài toán tìm kiếm âm thanh, trong đó tác giả đề xuất sử dụng mô tả khoảng cách của các cặp điểm cực trị trong ảnh phổ tần số làm đặc trưng của tín hiệu âm

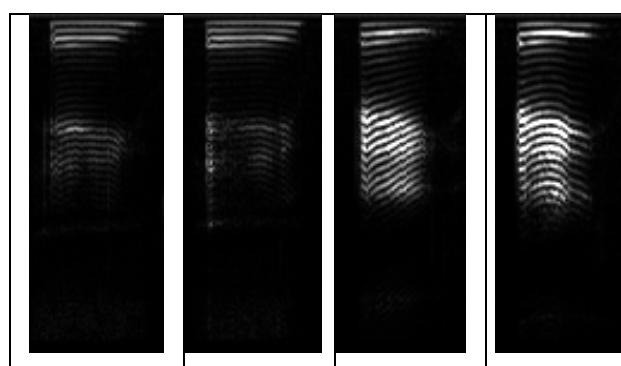
thanh⁵ [Zhang, 2015] [Reinhard, 2016]. Cụ thể, trong chương này, chúng tôi đề xuất hai hướng trích chọn đặc trưng tiếng nói từ phổ tần số của tín hiệu tiếng nói. Một là, đề xuất trích chọn đặc trưng SIFT_SPEECH, hai là đề xuất sử dụng mạng tích chập để tự động trích chọn đặc trưng trong phổ tần số của tiếng nói.

Để đánh giá hiệu quả của đặc trưng trích chọn từ phổ tần số của tiếng nói, chúng tôi tiến hành áp dụng cho bài toán nhận thức tiếng nói ở cấp độ liên kết với khái niệm đã biết, hay còn gọi là bài toán nhận dạng từ độc lập. Trong mô hình sử dụng trích chọn đặc trưng SIFT trực tiếp từ phổ tần số của tín hiệu tiếng nói, chúng tôi kết hợp phương pháp học máy LNBNN để phân lớp. Trong mô hình thứ hai sử dụng mạng tích chập dựa trên phổ tần số của tín hiệu tiếng nói chúng tôi sử dụng trực tiếp mạng tích chập với lớp SOFT_MAX để phân lớp tiếng nói.

Kết quả của chương sẽ chứng minh tính hiệu quả của đặc trưng trích chọn từ phổ tần số cho bài toán nhận thức tiếng nói.

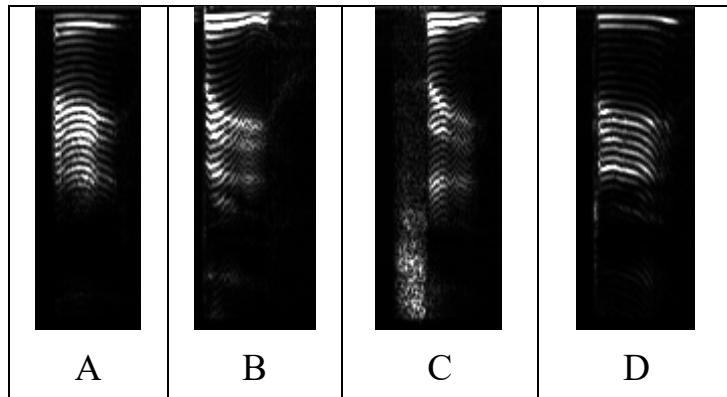
3.2. Phổ tần số của tín hiệu tiếng nói

Phổ của tiếng nói là một phương pháp biểu diễn tín hiệu trên miền kết hợp thời gian và tần số trong đó một chiều (trục tung) biểu diễn tần số, một chiều (trục hoành) biểu diễn thời gian và giá trị mỗi điểm ảnh là biên độ của các thành phần tần số có trong tín hiệu. Thực chất của cách biểu diễn này là biểu diễn tín hiệu trên miền tần số nhưng được thực hiện với các đoạn tín thời gian đủ ngắn để đảm bảo tín hiệu ổn định theo thời gian.

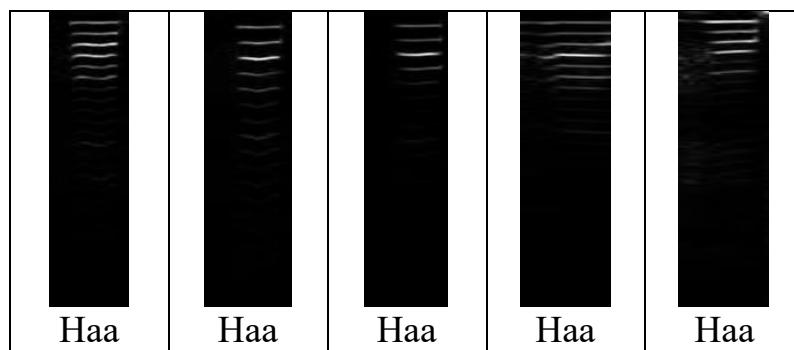


Hình 3. 1 Phổ của từ A trong tiếng Anh được nói bởi 4 người khác nhau

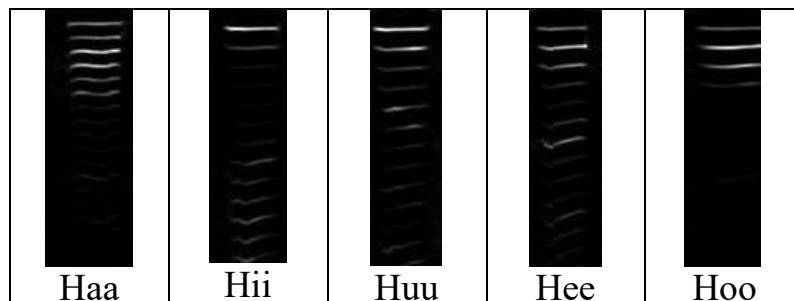
⁵ <http://www.ee.columbia.edu>



Hình 3. 2 Phổ của các chữ cái A-D trong tiếng Anh của cùng một người nói



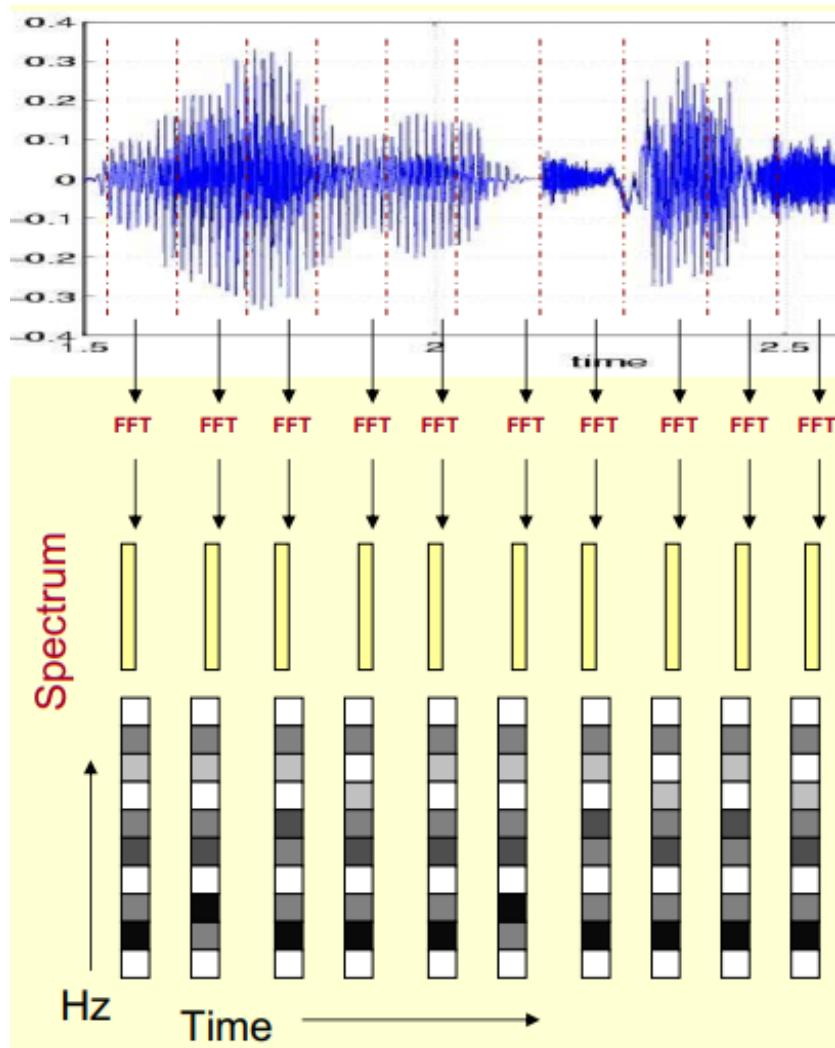
Hình 3. 3 Phổ của âm tiết Haa trong tiếng Nhật được nói bởi 5 người khác nhau



Hình 3. 4 Phổ của 5 âm tiết tiếng Nhật do cùng một người nói

Từ Hình 3.1, Hình 3.3 chỉ ra rằng cùng một tín hiệu tiếng nói được nói bởi những người nói khác nhau có xu hướng có nhiều điểm chung trong phổ tần số, Hình 3.2 và Hình 3.4 lại cho thấy phổ tần số của các tín hiệu tiếng nói khác nhau do cùng một người nói thì các điểm phổ tần số có xu hướng khác nhau.

Để có được phổ tần số, tín hiệu tiếng nói đầu tiên sẽ được phân thành các khung có thời gian ngắn nhằm đảm bảo tính ổn định của tín hiệu. Sau đó, các khung dữ liệu được tiền xử lý để tăng cường chất lượng. Tiếp theo, các khung dữ liệu được cho qua hàm cửa sổ, sau đó thực hiện phân tích FFT. Ghép các hệ số FFT theo thứ tự thời gian sẽ thu được phổ tần số của tín hiệu tiếng nói.



Hình 3. 5 Sơ đồ trích xuất phổ tần số của tín hiệu tiếng nói

3.3. Đặc trưng bất biến SIFT

SIFT là một đặc trưng được sử dụng trong lĩnh vực thị giác máy, dùng để nhận dạng và miêu tả những điểm đặc trưng cục bộ trong ảnh được giới thiệu bởi David Lowe năm 1999 [Lowe, 2004]. Đặc trưng SIFT bất biến với phép co dãn và phép xoay (Scale Invariant Feature Transform - SIFT) được sử dụng rất thành công trong bài toán nhận dạng đối tượng, nguyên nhân do SIFT được cho là có chung đặc điểm với đáp ứng của các nơ-ron thị giác sơ cấp [Lowe, 2004] [Lowe, 1999]. Tương tự như vùng vỏ não thị giác, vùng vỏ não thính giác sơ cấp được cho là có tổ chức theo mức độ biến đổi của tần số tương ứng với đáp ứng của các sợi sinh học trong óc tai [Pickles, 2012] [Purves, 2001] và não người nhận thức được âm thanh dựa vào thông tin về các tần số đạt cực trị và sự biến đổi xung quanh tần số đạt cực trị này. Điều này tương đồng với

điểm đặc trưng SIFT trong lĩnh vực thị giác máy. Đặc trưng SIFT đã được chứng minh là bất biến đối với phép co dãn, phép xoay và bất biến đối với hiện tượng méo hình [Karami, 2015] , nhưng chưa được chứng minh là bất biến với phép co dãn một chiều là một hiện tượng biến đổi phổ biến trong tiếng nói.

Ngày nay, phương pháp trích chọn đặc trưng này được ứng dụng rộng rãi trong nhận dạng đối tượng, mô hình hóa 3D [Leibe, 2004] . Đặc trưng SIFT có đặc điểm là bất biến đối với phép co dãn, với phép xoay và sự thay đổi của cường độ sáng. Phương pháp trích rút các đặc trưng bất biến SIFT từ một ảnh được thực hiện theo các bước sau:

Bước 1: Phát hiện các điểm cực trị trong không gian tỉ lệ

Bước đầu tiên này tiến hành tìm kiếm các điểm hấp dẫn trên tất cả các tỉ lệ và vị trí của ảnh. Bước này sử dụng hàm DoG (Different-of-Gaussian) để xác định tất cả các điểm hấp dẫn tiềm năng có tính bất biến với tỉ lệ và hướng của ảnh.

Bước 2: Định vị các điểm hấp dẫn

Khi đã lấy được tất cả những điểm hấp dẫn tiềm năng của ảnh, tiếp theo là lọc để thu được những điểm hấp dẫn chính xác hơn. SIFT sử dụng chuỗi khai triển mở rộng Taylor để lấy vị trí của các điểm cực trị chính xác hơn, sau đó xét xem nếu cường độ của điểm cực trị đó nhỏ hơn một giá trị ngưỡng cho trước thì sẽ loại bỏ điểm hấp dẫn tiềm năng đó.

Bên cạnh đó, DoG rất nhạy cảm với cạnh, để loại bỏ điểm hấp dẫn tieemg năng là các cạnh, SIFT sử dụng ma trận Hessian 2x2 để tính ra những đường cong chính. Khi các giá trị riêng lớn hơn một ngưỡng nào đó thì điểm hấp dẫn tiềm năng đó sẽ bị loại.

Bước 3: Xác định hướng cho các điểm hấp dẫn

Mỗi điểm hấp dẫn được gán cho một hướng phù hợp dựa trên các thuộc tính hình ảnh cục bộ đó là dựa vào hướng của điểm hấp dẫn này. Tại mỗi điểm hấp dẫn tính biểu đồ hướng trong vùng láng giềng của điểm hấp dẫn. Độ lớn của véc tơ định hướng và hướng của các điểm hấp dẫn được xác định theo công thức:

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2} \quad (3.1)$$

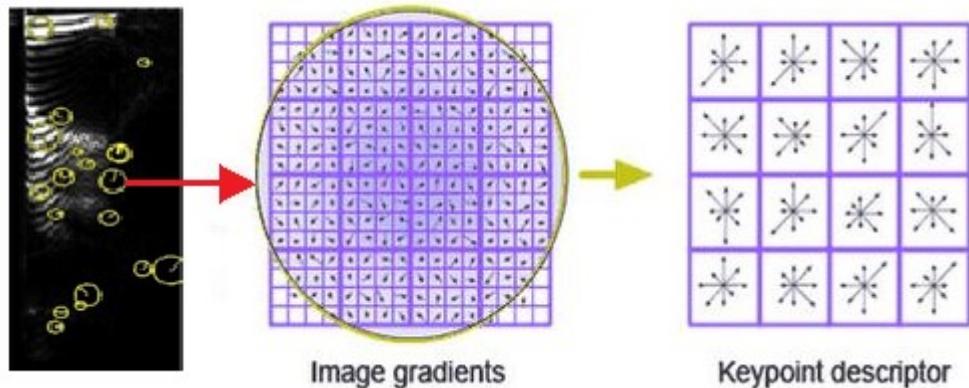
$$\theta(x, y) = \tan^{-1}((L(x, y + 1) - L(x, y - 1))/(L(x + 1, y) - L(x - 1, y))) \quad (3.2)$$

Trong đó $m(x,y)$ là độ lớn của vector định hướng, $\theta(x,y)$ là hướng của vector định hướng.

Một lược đồ hướng được tính từ định hướng gradient của các điểm lấy mẫu trong một khu vực xung quanh các điểm hấp dẫn. Đỉnh trong biểu đồ hướng tương ứng với hướng chủ đạo của gradient. Đỉnh cao nhất trong biểu đồ được phát hiện, và sau đó bất kỳ điểm nào khác có cao điểm là 80% so với đỉnh cao nhất cũng được sử dụng cũng tạo ra một điểm hấp dẫn với định hướng đó. Vì vậy, đối với các địa điểm có nhiều đỉnh cường độ tương tự sẽ có nhiều điểm hấp dẫn tạo ra tại cùng một vị trí và tỷ lệ, nhưng có hướng khác nhau.

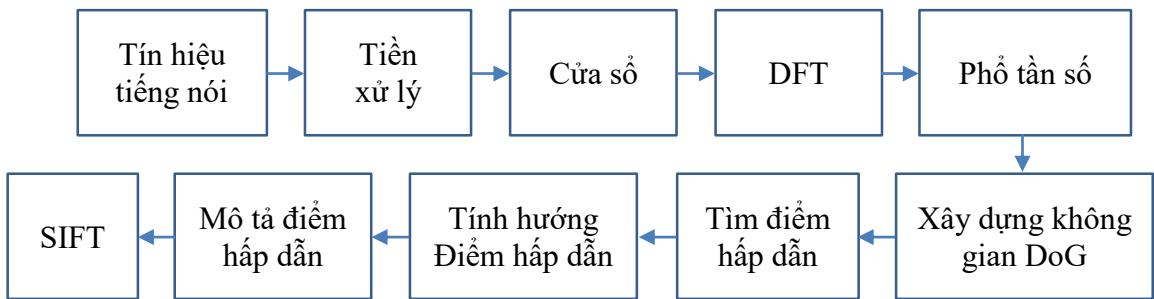
Bước 4: Mô tả các điểm hấp dẫn

Từ một lân cận 16×16 quanh điểm hấp dẫn được chia thành 16 vùng lân cận có kích thước 4×4 . Với mỗi vùng lân cận con, tính lược đồ histogram định hướng 8 bin. Vì vậy, có tổng cộng 128 giá trị bin. Nó được đại diện như là một véc tơ mô tả điểm hấp dẫn.

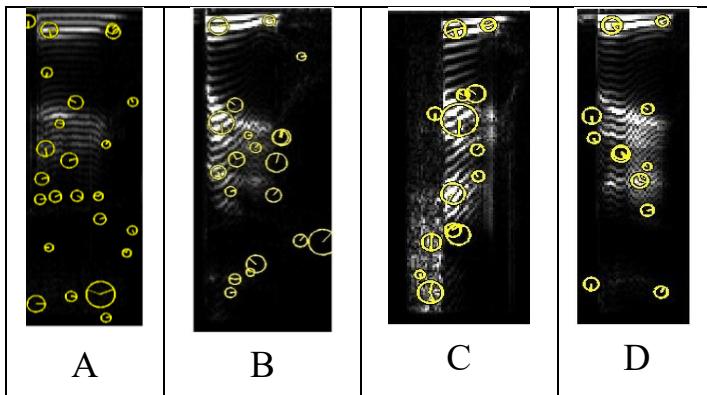


Hình 3. 6 Mô tả điểm hấp dẫn SIFT [Lowe, 1999]

Kết hợp với sơ đồ biểu diễn tín hiệu tiếng nói thành phô tần số ta thu được sơ đồ trích chọn đặc trưng SIFT-SPEECH từ phô tần số của tín hiệu tiếng nói (hình 3.7).



Hình 3. 7 Sơ đồ các bước trích chọn đặc trưng SIFT-SPEECH từ tín hiệu tiếng nói



Hình 3. 8 Một số điểm SIFT-SPEECH trích xuất từ phổ tần số của tín hiệu tiếng nói

Các điểm đặc trưng SIFT_SPEECH thu được từ ảnh phổ tần số của tín hiệu tiếng nói là các điểm cực trị trong phổ tần số, điều đó nghĩa là tại điểm đó biên độ của thành phần tần số đó là cực đại hoặc cực tiểu tương ứng với âm lượng của thành phần tần số đó là lớn hơn hoặc nhỏ hơn so với các thành phần tần số xung quanh nó. Não bộ sẽ nhận thức âm thanh với các cao độ khác nhau qua các vị trí khác nhau mà những xung tín hiệu được gởi đến từ các nang bào. Âm thanh có âm lượng càng lớn sẽ giải tỏa nhiều năng lượng hơn và làm di chuyển nhiều nang bào hơn. Não bộ nhận thức được các âm thanh là nhờ vào số lượng các nang bào cùng được kích hoạt trong một vị trí nào đó. Mặc dù tiếng nói bị phụ thuộc vào người nói, hoàn cảnh nói, nhưng tiếng nói vẫn tồn tại những đặc trưng bất biến do cách phát âm của cùng một từ giữa những người nói khác nhau phải giống nhau, vì vậy, tác giả cho rằng sẽ tồn tại những điểm bất biến của những đỉnh cộng hưởng tần số trong tín hiệu tiếng nói. Những đỉnh cộng hưởng này có thể bị tịnh tiến lên xuống do tần số cơ bản của người nói khác nhau, có thể bị tịnh tiến sang trái, phải do thời gian thu tín hiệu lệch nhau, nhưng xét trong một phạm vi cục bộ thì chúng là bất biến. Vì vậy, SIFT-SPEECH là một đặc trưng phù hợp cho bài toán nhận thức tiếng nói.

3.4. Phương pháp phân lớp NBNN

Phương pháp phân lớp Naïve Bayes Nearest Neighbor (NBNN) được đề xuất bởi Boiman cho bài toán phân lớp đối tượng trong lĩnh vực thị giác máy [Boiman O., Shechtman E., and Iran M., 2008] . NBNN là một phương pháp phân lớp phi tham số đồng thời không cần phải thực hiện huấn luyện trước khi phân lớp. Phương pháp này được thực nghiệm chứng tỏ có hiệu quả đối với bài toán phân lớp ảnh do không phải thực hiện lượng tử hóa các véc tơ đặc trưng của dữ liệu, đồng thời phương pháp này thực hiện so sánh mẫu truy vấn đến từng lớp dữ liệu thay cho việc so sánh với từng mẫu dữ liệu của các lớp. Phương pháp NBNN được mô tả như sau:

Bài toán: Cho một mẫu dữ liệu cần phân lớp Q được biểu diễn bởi một tập các véc tơ đặc trưng d_1, d_2, \dots, d_n . Tìm lớp C sao cho cực tiểu hóa tổng khoảng cách từ các véc tơ đặc trưng của Q tới véc tơ gần nhất tương ứng của tất cả các lớp.

Theo công thức Bayes ta có

$$p(C|Q) = \frac{p(Q|C)p(C)}{p(Q)} \quad (3.3)$$

$$posterior = \frac{\text{likelihood} \times \text{prior}}{\text{evidence}} \quad (3.4)$$

Cho một dữ liệu mới cần phân lớp, chúng ta cần xác định xem dữ liệu mới đó thuộc lớp nào. Như chúng ta đã biết rằng việc cực đại hóa xác xuất hậu nghiệm sẽ làm giảm sai số phân lớp trung bình

$$\hat{C} = \operatorname{argmax}_C p(C|Q) = \max_C p(Q|C) \quad (3.5)$$

Với giả thiết các thuộc tính của dữ liệu là độc lập khi đó ta có

$$\begin{aligned} p(Q|C) &= p(d_1, d_2, \dots, d_n | C) \\ &= \prod_{i=1}^n p(d_i | C) \end{aligned} \quad (3.6)$$

Thực hiện Logarit 2 vé của phương trình trên ta thu được

$$\hat{C} = \operatorname{argmax}_C \log(p(C|Q)) \quad (3.7)$$

$$= \operatorname{argmax}_C \log \left(\sum_{i=1}^n p(d_i|C) \right) \quad (3.8)$$

$$= \operatorname{argmax}_C \sum_{i=1}^n \log(p(d_i|C)) \quad (3.9)$$

Áp dụng công thức tính xác suất $p(d_i|C)$ bằng công thức ước lượng của số Parzen với nhân K ta thu được

$$\hat{p}_r(d_i|C) = \frac{1}{L} \sum_{j=1}^L K(d_i - d_j^C) \quad (3.10)$$

Trong đó L là tổng số véc tơ đặc trưng trong tập huấn luyện của lớp C, và d_j^C là véc tơ gần nhất thứ j của véc tơ d_i thuộc lớp C. Công thức này có thể xấp xỉ tiếp bằng cách chỉ giữ lại r phần tử gần nhất thay vì tính tổng khoảng cách tới tất cả các véc tơ đặc trưng thuộc lớp C trong tập huấn luyện, khi đó ta có công thức tính xác suất như sau:

$$\hat{p}_r(d_i|C) = \frac{1}{L} \sum_{j=1}^r K(d_i - d_j^C) \quad (3.11)$$

Chọn $r=1$ ta thu được phương pháp phân lớp NBNN, khi đó

$$\hat{p}_r(d_i|C) = \frac{1}{L} K(d_i - NN_C(d_i)) \quad (3.12)$$

Trong đó $NN_C(d_i)$ là véc tơ đặc trưng gần nhất của véc tơ d_i trong lớp C

Chọn K là hàm nhân Gaussian và thay vào công thức ta thu được

$$\hat{C} = \operatorname{argmax}_C \left[\sum_{i=1}^n \log \left(\frac{1}{L} e^{-\frac{1}{2\sigma^2} \|d_i - NN_C(d_i)\|^2} \right) \right] \quad (3.13)$$

$$\hat{C} = \operatorname{argmin}_C \left[\sum_{i=1}^n \|d_i - NN_C(d_i)\|^2 \right] \quad (3.14)$$

Từ đó ta có thuật toán phân lớp NBNN như sau:

Thuật toán NBNN (Q)
Đầu vào: $C = \{C_1, C_2, \dots, C_L\}$ là tập nhãn của dữ liệu huấn luyện $T = \{T_1, T_2, \dots, T_L\}$ là tập các đặc trưng của dữ liệu huấn luyện $Q = \{d_1, d_2, \dots, d_Q\}$ with $d_i \in R^m \forall i = 1 \dots Q$ là một truy vấn
Đầu ra: Class of Q
<ol style="list-style-type: none"> 1. for all $d_i \in Q$ do 2. for all classes C do 3. $\text{totals}[C] \leftarrow \text{totals}[C] + \ d_i - \text{NN}_C(d_i)\ ^2$ 4. end for 5. end for 6. return $\underset{C}{\operatorname{argmin}} \text{totals}[C]$

3.5. Phương pháp phân lớp LNBNN

Phương pháp Local Naïve Bayes Nearest neighbor (LNBNN) [Sancho, 2012] được Sancho đề xuất năm 2012 nhằm cải tiến thuật toán NBNN cho bài toán phân lớp ảnh. Đối với thuật toán NBNN, thuật toán phải tìm khoảng cách nhỏ nhất từ mỗi điểm đặc trưng trong tập truy vấn tới các lớp, như vậy với bài toán phân lớp có nhiều lớp và trong trường hợp điểm đặc trưng này quá xa so với hầu hết các lớp và chỉ gần một số lớp nhất định nào đó thì việc tính khoảng cách này là không cần thiết. Vì vậy Sancho đề xuất phương pháp cải tiến cho NBNN bằng cách thay vì phải tìm khoảng cách ngắn nhất từ mỗi điểm đặc trưng tới tất cả các lớp thì LNBNN chỉ tìm khoảng cách ngắn nhất đến các lớp có mặt trong K hàng xóm gần nhất của điểm đặc trưng đó. Như vậy, để thực hiện được thuật toán này, đầu tiên LNBNN thực hiện trọn tất cả điểm đặc trưng thu được từ tập huấn luyện tạo thành một cơ sở dữ liệu các điểm đặc trưng cho tất cả các lớp. Tiếp theo, LNBNN tìm tập hợp K điểm đặc trưng gần nhất của mỗi điểm đặc trưng trong tập truy vấn và cập nhật khoảng cách ngắn nhất tìm được đến các lớp có mặt trong K hàng xóm đó. Như vậy, nếu thực hiện tính tổng như NBNN thực hiện thì lớp nào càng xuất hiện nhiều trong K hàng xóm gần nhất của mỗi điểm đặc trưng của truy vấn thì tổng khoảng cách từ truy vấn

đến lớp đó càng tăng do đó không xác định được tổng khoảng cách nhỏ nhất. Vì vậy, thay vì cập nhật khoảng cách từ điểm đặc trưng đến lớp có mặt trong K hàng xóm gần nhất, LNBNN cập nhật hiệu khoảng cách nhỏ nhất tới lớp đó với khoảng cách tới hàng xóm thứ K+1 (hàng xóm thứ K+1 được coi như là biên giới, một khoảng cách đủ xa để có thể coi 2 phần tử là gần nhau). Do đó, tổng luôn được cập nhật một số âm. Khi đó, lớp nào càng xuất hiện nhiều thì tổng này càng âm, lớp nào càng ít xuất hiện thì tổng này càng gần 0 và lớp nào không xuất hiện trong K hàng xóm gần nhất của tất cả các điểm đặc trưng của truy vấn sẽ có tổng là 0. Như vậy, tổng nào có giá trị nhỏ nhất chính là nhãn lớp cần tìm.

Thuật toán 3. 2 Thuật toán LNBNN

Thuật toán LNBNN (Q, K)

Đầu vào:

$T = \{T_1, T_2, \dots, T_N\}$ là tập N mẫu huấn luyện

$T_i = \{d_{i_1}, d_{i_2}, \dots, d_{i_{N_i}}\}$ với $d_{i_j} \in R^m \forall j = 1..N_i$

$C = \{C_1, C_2, \dots, C_L\}$ là tập nhãn L nhãn

$Q = \{d_1, d_2, \dots, d_{N_Q}\}, d_i \in R^m \forall i = 1..N_Q$, truy vấn có N_Q điểm đặc trưng
Tham số K

Đầu ra: nhãn của Q

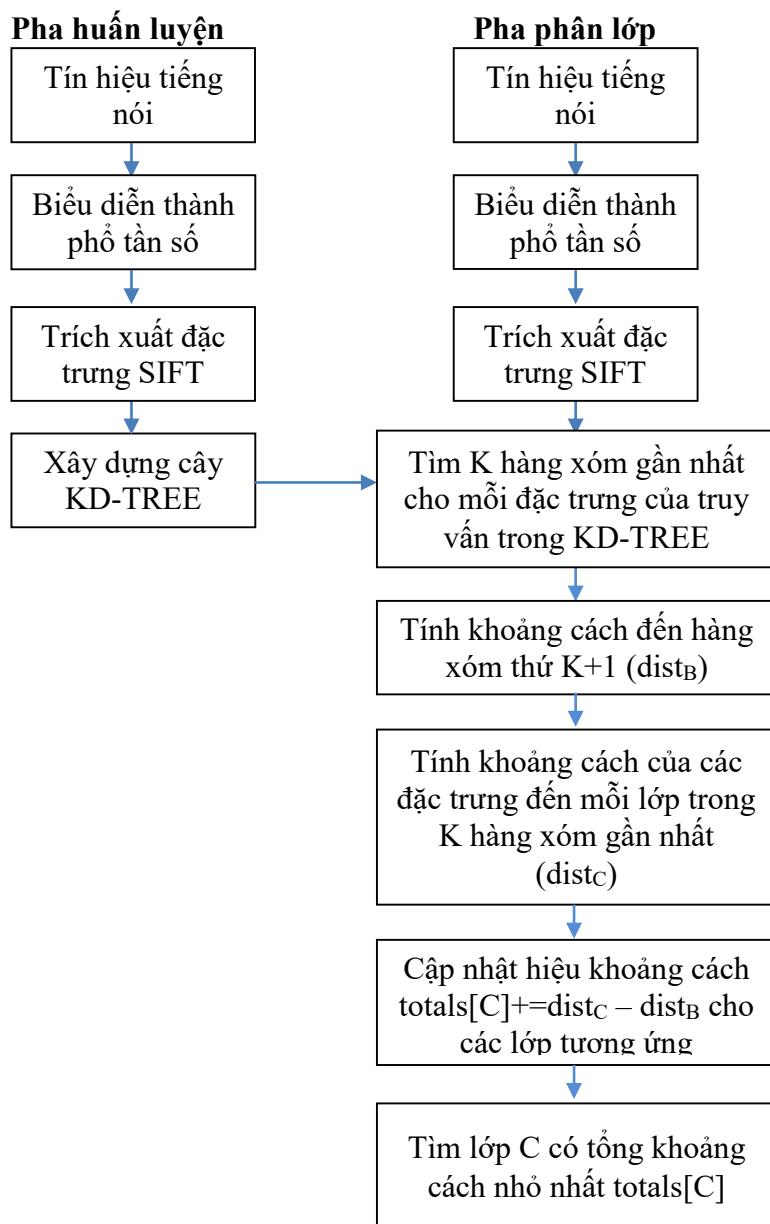
```

1. for all  $d_i \in Q$  do
2:   find  $\{p_1, p_2, \dots, p_{K+1}\}$  là  $K + 1$  hàng xóm gần nhất của  $d_i$ 
3:    $dist_B = \|d_i - p_{K+1}\|^2$ 
4:   for all classes C in the K nearest neighbors do
5:      $dist_C = \min_{\{p_j | class(p_j) = c\}} \|d_i - p_j\|^2$ 
6:      $totals[C] \leftarrow totals[C] + dist_C - dist_B$ 
7:   end for
8: end for
9: return  $\operatorname{argmin}_c totals[C]$ 

```

3.6. Hướng tiếp cận trích chọn đặc trưng tiếng nói dựa trên phổ tần số cho bài toán nhận thức tiếng nói

Trong nghiên cứu này, chúng tôi đề xuất mô hình nhận thức tiếng nói dựa trên trích chọn đặc trưng SIFT từ phổ tần số của tín hiệu tiếng nói kết hợp với phương pháp phân lớp LNBNN. Sơ đồ minh họa mô hình được miêu tả ở hình 3.9.



Hình 3. 9 Mô hình phân lớp tiếng nói bằng LNBNN-SIFT-SPEECH

Thuật toán phân lớp LNBNN kết hợp với đặc trưng SIFT trích chọn từ phổ tần số được mô tả ở thuật toán 3.3.

Bước 1. Biến đổi tín hiệu tiếng nói thành phổ tần số

Đầu tiên tín hiệu tiếng nói được tiến xử lý để loại bỏ nhiễu và nhấn mạnh các thành phần tần số mà tai người cảm nhận được tốt hơn thông qua các bộ lọc tần số. Tiếp theo, tín hiệu tiếng nói được phân thành các đoạn tín hiệu ngắn để đảm bảo tính ổn định của tín hiệu khi thực hiện phép biến đổi DFT.

Trong nghiên cứu này, chúng tôi chia tín hiệu tiếng nói thành các đoạn 10ms, các đoạn này chồng lên nhau 5 ms. Sau đó, tiến hành biến đổi DFT cho từng đoạn tín hiệu ngắn này để thu được phổ tần số cho từng đoạn tín hiệu tiếng nói. Ghép nối các véc tơ phổ của từng đoạn này theo thứ tự thời gian sẽ thu được một ma trận các thành phần tần số có trong tín hiệu tiếng nói theo toàn bộ thời gian của tín hiệu. Ma trận này chính là phổ tần số của tín hiệu tiếng nói.

Thuật toán 3. 3 Thuật toán LNBNN-SIFT-SPEECH

Thuật toán LNBNN-SIFT-SPEECH(Q, K)

Đầu vào:

$T = \{T_1, T_2, \dots, T_N\}$ là tập N mẫu huấn luyện

$C = \{C_1, C_2, \dots, C_L\}$ là tập L nhãn

Q: là mẫu truy vấn

Tham số K

Đầu ra: nhãn của Q

Bước 1. Biến đổi tín hiệu tiếng nói trong tập huấn luyện và truy vấn thành phổ tần số

Bước 2. Trích xuất đặc trưng SIFT từ phổ tần số

Bước 3. Xây dựng cây tìm kiếm KD-TREE

Bước 4. Tìm K+1 hàng xóm gần nhất cho mỗi điểm đặc trưng của truy vấn

Bước 5. Tính khoảng cách biên

Bước 6. Cập nhật khoảng cách nhỏ nhất đến mỗi lớp tìm thấy trong K hàng xóm gần nhất

Bước 7. Tìm lớp có tổng khoảng cách nhỏ nhất.

Bước 2. Trích xuất đặc trưng SIFT từ phổ tần số

Bước này sẽ tiến hành trích chọn đặc trưng theo các bước đã mô tả ở phần 3.1. Kết quả ta sẽ thu được một tập các điểm đặc trưng SIFT, trong đó mỗi điểm được biểu diễn bởi một véc tơ có 128 chiều là mô tả lân cận cục bộ của điểm

hấp dẫn. Khi đó, mỗi mẫu huấn luyện sẽ được biểu diễn bằng một tập hợp các điểm đặc trưng SIFT này.

Bước 3. Xây dựng cây tìm kiếm KD-TREE

Phương pháp LNBNN phân lớp dữ liệu dựa trên việc xấp xỉ xác suất hậu nghiệm bằng khoảng cách gần nhất đến mỗi lớp. Do đó, LNBNN sẽ phải thực hiện tìm kiếm K hàng xóm gần nhất của mỗi điểm đặc trưng SIFT của tín hiệu truy vấn. Việc tìm kiếm này sẽ tốn rất nhiều thời gian nếu dữ liệu huấn luyện lớn và số lượng điểm đặc trưng của truy vấn lớn. Vì vậy, để tăng tốc độ thực hiện tìm kiếm K hàng xóm gần nhất, LNBNN sử dụng cấu trúc dữ liệu KD-TREE để lưu trữ và thực hiện tìm kiếm hàng xóm gần nhất.

Bước 4. Tìm K+1 hàng xóm gần nhất cho mỗi điểm đặc trưng của truy vấn

Bước 5. Tính khoảng cách biên

Khoảng cách biên là khoảng cách từ điểm đặc trưng của truy vấn đến điểm đặc trưng là hàng xóm gần thứ $K + 1$ của điểm đặc trưng truy vấn này.

Bước 6. Cập nhật khoảng cách nhỏ nhất đến mỗi lớp tìm thấy trong K hàng xóm gần nhất

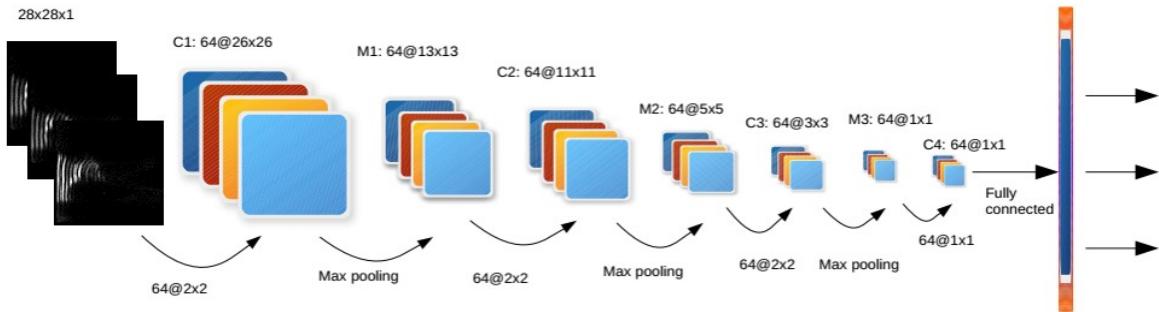
K hàng xóm gần nhất của mỗi điểm đặc trưng của truy vấn sẽ thuộc về một số lớp khác nhau. Tìm khoảng cách nhỏ nhất từ điểm đặc trưng truy vấn đến mỗi lớp thuộc K hàng xóm gần nhất. Tính hiệu giữa khoảng cách này với khoảng cách biên ở bước 5 và cập nhật vào tổng khoảng cách tương ứng với mỗi lớp. Nếu khoảng cách từ điểm đặc trưng của truy vấn tới lớp nào càng nhỏ thì hiệu số càng lớn, khi đó tổng khoảng cách sẽ được cộng thêm một số âm càng nhỏ, làm cho tổng khoảng cách này càng nhỏ.

Bước 7. Tìm lớp có tổng khoảng cách nhỏ nhất

Với mỗi điểm đặc trưng của truy vấn, sẽ có một số lớp được cập nhật thêm một số âm. Vì vậy, lớp nào càng được tìm thấy nhiều thì tổng khoảng cách càng âm. Kết quả lớp nào có tổng khoảng cách đến truy vấn là nhỏ nhất thì truy vấn thuộc về lớp đó.

3.7. Hướng tiếp cận mạng tích chập dựa trên phô tần số cho bài toán nhận thức tiếng nói

Mạng tích chập đã được sử dụng rất thành công trong lĩnh vực nhận dạng ảnh, trong phần này chúng tôi đề xuất sử dụng mô hình mạng tích chập cho bài toán nhận thức tiếng nói và thực nghiệm trên bài toán dạng tiếng nói rời rạc.



Hình 3. 10 Mô hình CNN cho bài toán nhận dạng tiếng nói dựa trên phô tần số

Kiến trúc mạng CNN được trình bày ở hình 3.10. Trong mô hình này, đầu tiên tín hiệu tiếng nói sẽ được chuyển đổi sang biểu diễn dưới dạng phô tần số theo sơ đồ 3.5. Dữ liệu phô tần số khi này có dạng một ma trận giống như dữ liệu ảnh, trong đó một chiều là theo thời gian và một chiều theo tần số. Phô tần số sau đó được biến đổi tỷ lệ để thu được một ma trận dữ liệu có kích thước 28x28 (trong mô hình thực nghiệm) để giảm số trọng số phải học của mô hình. Dữ liệu phô tần số sau khi biến đổi về cùng kích thước được sử dụng làm dữ liệu đầu vào cho mô hình CNN. Tiếp theo, là lớp tích chập với 64 bộ lọc. Kết quả thu được sau khi qua lớp tích chập thứ nhất được đưa vào lớp lấy mẫu Max Pooling, thu được 64 bộ dữ liệu có kích thước 13x13. Tiếp theo là lớp tích chập thứ hai với 64 bộ lọc, và lớp Max Pooling thứ 2 và thu được dữ liệu đầu ra của lớp này là 64x5x5. Lớp tích chập và lấy mẫu cuối cùng sẽ cho kết quả là 64x1x1. Kết quả này được kết nối đầy đủ với nhãn của mẫu dữ liệu đầu vào để thực hiện phân lớp thông qua hàm soft-max. Kết quả thực nghiệm của mô hình sẽ được trình bày ở phần 3.8.

3.8. Thực nghiệm và kết quả

Trong nghiên cứu này, chúng tôi tiến hành 05 thực nghiệm trên 05 bộ dữ liệu. Thực nghiệm 1 thực hiện so sánh độ chính xác của phương pháp phân lớp LNBNN với đặc trưng SIFT và đặc trưng MFCC. Thực nghiệm 2 là so sánh độ chính xác của phương pháp phân lớp LNBNN với dữ liệu bị co dãn một chiều

(theo thời gian) của tín hiệu tiếng nói. Thực nghiệm 3 so sánh phương pháp phân lớp LNBNN với một số phương pháp phân lớp phổ biến hiện nay như Naïve Bayes, Bayesian Network, Support vector machine, Random Forest and Decision Tree Analysis J48. Thực nghiệm 4, đánh giá khả năng học thêm dữ liệu huấn luyện của mô hình. Thực nghiệm này gồm 2 thực nghiệm con là đánh giá khả năng học thêm dữ liệu huấn luyện với các lớp đã có và khả năng học thêm tri thức mới (học thêm dữ liệu huấn luyện đối với các lớp chưa được học). Thực nghiệm cuối là sử dụng mô hình tích chập cho bài toán nhận thức tiếng nói.

3.8.1. *Dữ liệu thực nghiệm*

Trong các thực nghiệm này, chúng tôi sử dụng 05 bộ dữ liệu tiếng nói đó là cơ sở dữ liệu tiếng nói các chữ cái tiếng Anh (ISOLET) [Fanty, 1994] , cơ sở dữ liệu tiếng nói các chữ số trong tiếng Anh DIGITS⁶, tên một số địa điểm trong tiếng Việt VN PLACES⁷, cơ sở dữ liệu tiếng nhật TMW (Tohoku University - Matsushita Isolated Word -TMW)⁸, và cơ sở dữ liệu 05 nguyên âm trong tiếng Nhật JVPD (Five Japanese Vowels of Males, Females, and Children Along with Relevant Physical Data - JVPD)⁹.

Cơ sở dữ liệu ISOLET gồm 676 mẫu phát âm 26 chữ cái trong tiếng Anh được nói bởi 26 người khác nhau. Cơ sở dữ liệu EN DIGITS (0-9, o) gồm 414 mẫu cho mỗi lớp. Cơ sở dữ liệu VN PLACES gồm 8 lớp (caphe, dung, karaoke, khachsan, khong, matxa, tramatm, trolai) là tên của một số địa điểm được phát âm bằng tiếng Việt trong đó mỗi lớp có 485 mẫu dữ liệu. Cơ sở dữ liệu TMW (Tohoku University - Matsushita Isolated Word) là thu âm 212 từ trong tiếng Nhật từ 60 người (30 nam và 30 nữ), mỗi lớp có 55 mẫu huấn luyện. Bộ dữ liệu JVPD là 5 nguyên âm trong tiếng Nhật thu âm từ các đối tượng Nam, Nữ cả trẻ em có tuổi từ 6 đến 56, tổng số mỗi nguyên âm có 384 mẫu huấn luyện.

3.8.2. *Thí nghiệm so sánh độ chính xác phân lớp của đặc trưng SIFT với đặc trưng MFCC khi sử dụng LNBNN*

⁶ <https://catalog.ldc.upenn.edu/LDC2008S07>

⁷ <http://www.alovoice.vn/ai/du-lieu-tieng-noi-tieng-viet/>

⁸ <http://research.nii.ac.jp/src/en/TMW.html>

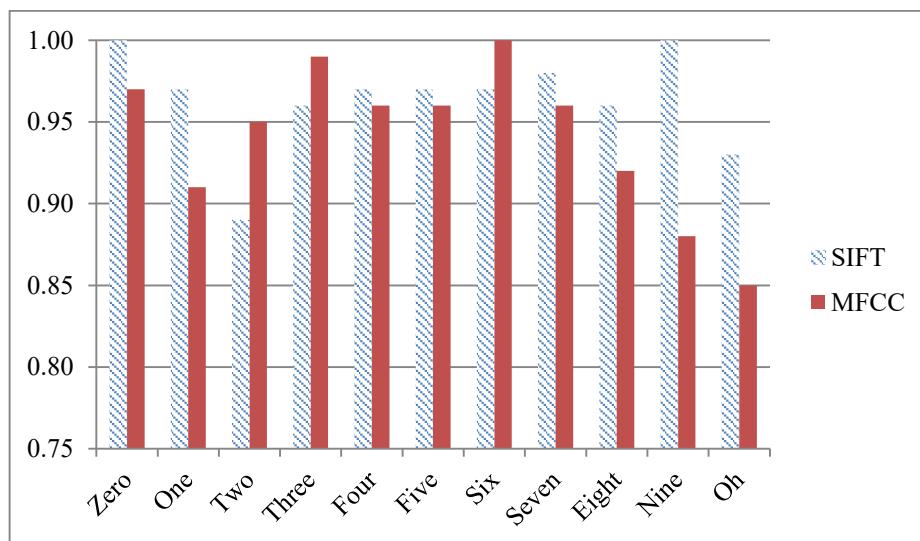
⁹ <http://research.nii.ac.jp/src/en/JVPD.html>

Trong thí nghiệm này LNBNN được sử dụng để phân lớp tiếng nói kết hợp với 2 phương pháp trích chọn đặc trưng là MFCC và SIFT. Thí nghiệm được tiến hành với 05 cơ sở dữ liệu, bảng 3.1 là kết quả thực hiện:

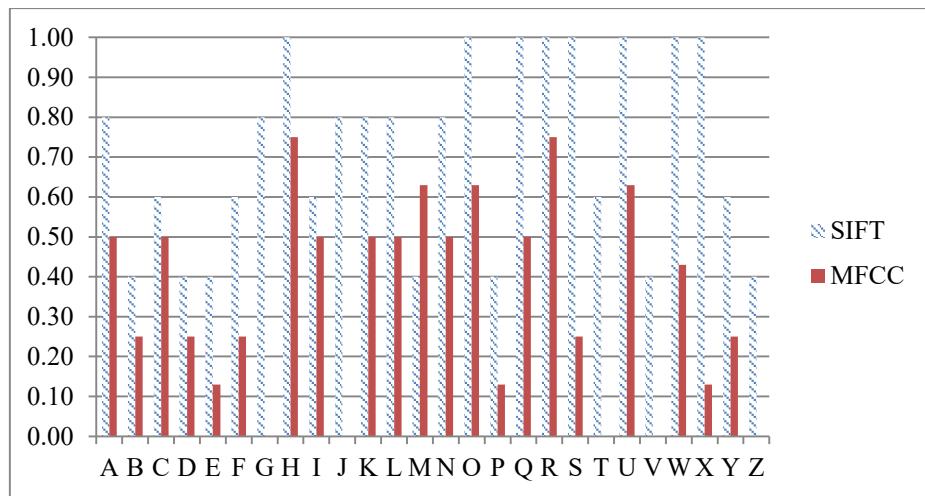
Bảng 3.1 So sánh độ chính xác phân lớp của LNBNN với SIFT và MFCC

Bộ dữ liệu	SIFT	MFCC
ISOLET	0.73	0.34
English Digits	0.96	0.94
Vietnamese Places	0.95	0.39
TMW	0.83	0.39
JVPD	0.97	0.53

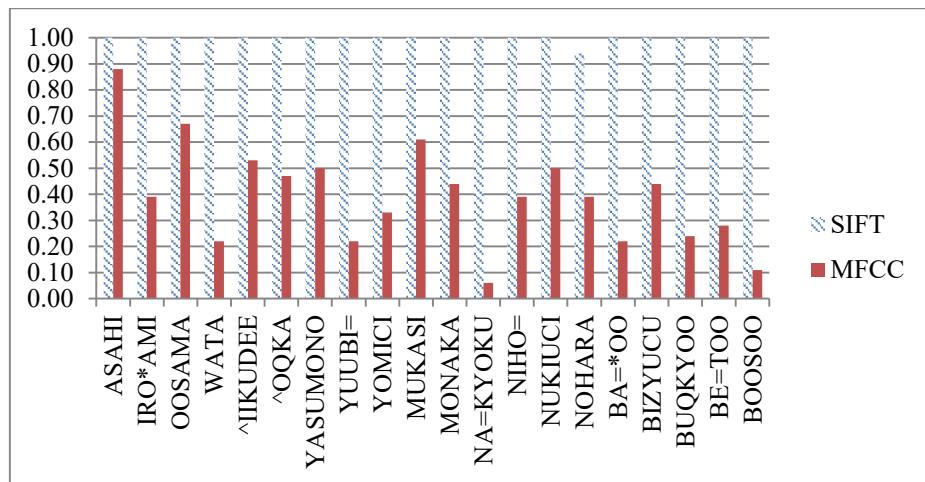
Bảng 3.1 cho thấy độ chính xác của LNBNN khi sử dụng kết hợp với phương pháp trích chọn đặc trưng SIFT trên phổ tần số có kết quả cao hơn so với LNBNN sử dụng kết hợp với MFCC phương pháp trích chọn đặc trưng tiếng nói được sử dụng phổ biến trong các phương pháp hiện nay. Dữ liệu đạt kết quả phân lớp cao nhất đối với bộ dữ liệu JVPD đạt 97% khi sử dụng SIFT và chỉ đạt 39% với MFCC. Điều này chứng tỏ trích chọn đặc trưng SIFT từ phổ tiếng nói phù hợp hơn so với đặc trưng MFCC cho bài toán nhận thức tiếng nói khi sử dụng phương pháp phân lớp LNBNN.



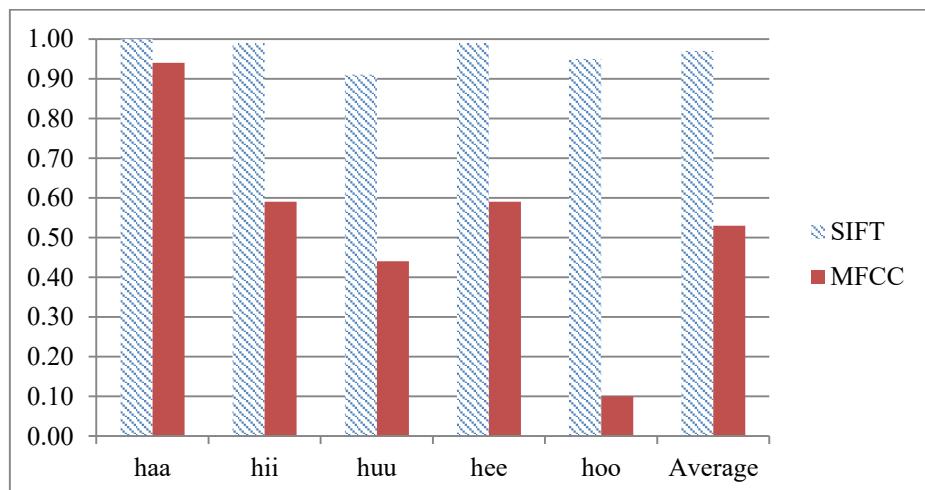
Hình 3.11 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên dữ liệu số English Digits



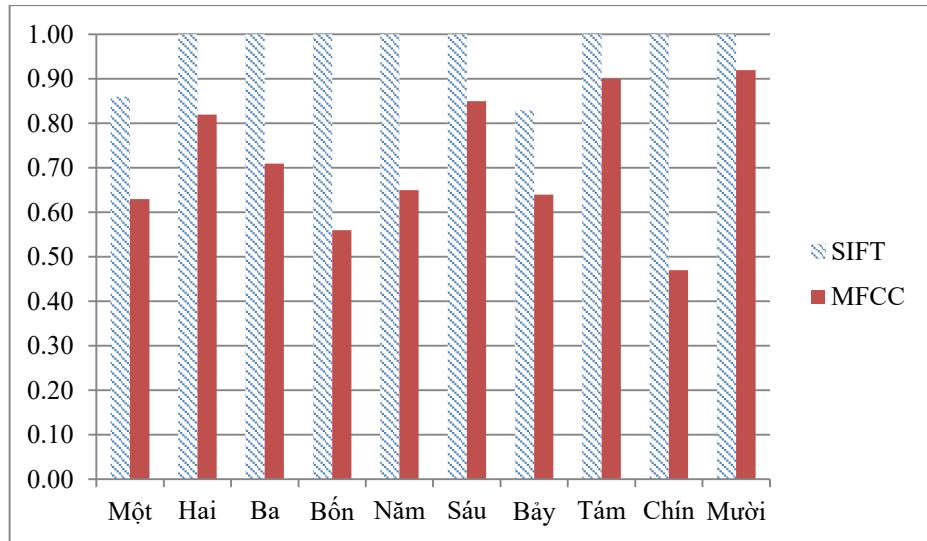
Hình 3.12 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên dữ liệu ISOLET.



Hình 3.13 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên 20 lớp đầu tiên của dữ liệu TMW



Hình 3.14 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên dữ liệu JVPD



Hình 3.15 So sánh độ chính xác của LNBNN kết hợp với MFCC và SIFT trên dữ liệu số tiếng Việt

3.8.3. Thí nghiệm với dữ liệu co dãn theo thời gian

Từ kết quả thí nghiệm ở phần 3.8.2 cho thấy đặc trưng SIFT được trích chọn từ phổ tần số của tín hiệu tiếng nói phù hợp cho bài toán nhận thức tiếng nói. Đặc trưng này đã được chứng minh là bất biến đối với phép co dãn hai chiều, phép tịnh tiến, phép xoay và bất biến cả đối với dữ liệu bị méo. Tuy nhiên, dữ liệu tiếng nói thường bị biến đổi theo một trong hai cách đó là phép tịnh tiến do thời gian thu tín hiệu không giống nhau, và phép co dãn một chiều do tốc độ nói của người nói không giống nhau. Trong phần này, chúng tôi sẽ thực nghiệm với dữ liệu tiếng nói bị co dãn một chiều với tỉ lệ 10%, 20% và 30%. Kết quả thực nghiệm trên các bộ dữ liệu được trình bày tại bảng 3.2.

Bảng 3.2 So sánh kết quả đối với dữ liệu bị co dãn một chiều

Database	Origin	Scale 10%	Scale 20%	Scale 30%
ISOLET	0.734	0.731	0.729	0.724
English Digits	0.962	0.962	0.959	0.958
Vietnamese Places	0.953	0.951	0.948	0.941
TMW	0.830	0.830	0.825	0.808
JVPD	0.973	0.972	0.967	0.963

Từ kết quả ở bảng 3.2 cho thấy với dữ liệu tiếng nói bị co dãn một chiều ở mức độ nhỏ thì độ chính xác hầu như không đổi.

3.8.4. Thí nghiệm so sánh LNBNN và các phương pháp phân lớp khác

Thí nghiệm thứ 2 so sánh phương pháp phân lớp LNBNN với một số phương pháp phân lớp phổ biến hiện nay như Naïve Bayes, Bayesian Network, Support vector machine (SVM), Random Forest and Decision Tree Analysis J48 (Tree.J48).

Đối với phương pháp phân lớp LNBNN đặc trưng được sử dụng là SIFT và MFCC, trong khi các phương pháp phân lớp khác đòi hỏi dữ liệu huấn luyện và dữ liệu kiểm tra phải được biểu diễn bằng những véc tơ có cùng số chiều. Vì vậy, chúng tôi tiến hành lượng tử hóa các véc tơ đặc trưng thu được từ dữ liệu bằng cách sử dụng phương pháp LBG. Các điểm đặc trưng SIFT sau khi được trích chọn từ phổ tần số của tiếng nói được lượng tử hóa thành 16 véc tơ 128 chiều, sau đó được chuyển thành một véc tơ có số chiều là 16x128 làm đại diện cho mỗi mẫu dữ liệu.

Bảng 3. 3 So sánh độ chính xác của các phương pháp phân lớp với đặc trưng MFCC

Method	ISOLET	English Digits	Vietnamese Places	TMW	JVPD
LNBNN	34.0	94.1	38.5	39.0	87.1
Naïve Bayes	64.2	98.6	67.6	44.6	44.5
Bayes Net	57.0	99.5	70.2	21.3	21.3
SVM	61.6	99.5	78.0	40.7	96.5
RandomForest	64.4	98.4	71.8	56.7	97.2
TreeJ48	38.1	90.2	53.8	15.2	82.7

Đối với đặc trưng MFCC, chúng tôi trích 18 véc tơ hệ số MFCC từ mẫu tiếng nói sau đó cũng thực hiện lượng tử hóa bằng phương pháp LBG thành 16 véc tơ, cuối cùng chuyển thành một véc tơ có số chiều là 16x18 chiều.

Bảng 3. 4 So sánh độ chính xác của các phương pháp phân lớp với đặc trưng SIFT

Method	ISOLET	English Digits	Vietnamese Places	TMW	JVPD
LNBNN	72.8	96.2	95.0	83.0	96.9
Naïve Bayes	32.8	50.4	58.5	34.1	55.8
Bayes Net	20.6	57.2	70.5	33.1	60.8
SVM	3.8	11.3	12.5	8.5	35.2
RandomForest	37.7	70.7	78.5	69.0	62.4
Tree J48	18.3	47.3	60.3	17.4	46.8

Bảng 3.3 cho thấy LNBNN khi kết hợp với đặc trưng MFCC có kết quả phân lớp thấp hơn so với các phương pháp phân lớp khác trong khi LNBNN kết hợp với đặc trưng SIFT (bảng 3.4) cho độ chính xác phân lớp cao nhất so với các phương pháp phân lớp khác.

3.8.5. Thí nghiệm khả năng học tăng cường của LNBNN

Một trong những ưu điểm của phương pháp phân lớp LNBNN đó là LNBNN cho phép học thêm dữ liệu sau khi huấn luyện mà không phải thực hiện lại quá trình huấn luyện đối với toàn bộ dữ liệu. Thí nghiệm này sẽ minh họa và đánh giá khả năng học thêm dữ liệu huấn luyện mới cũng như tri thức mới đối với phương pháp phân lớp LNBNN. Để minh họa khả năng học thêm dữ liệu mới chúng tôi tiến hành thực nghiệm như sau: Đầu tiên chúng tôi chia dữ liệu huấn luyện thành 5 phần, mỗi phần chiếm 20% của tập dữ liệu huấn luyện. Tiếp theo, chúng tôi tiến hành đưa từng phần vào huấn luyện sau đó tiến hành đánh giá mô hình và tiếp tục bổ sung thêm 20% dữ liệu huấn luyện tiếp theo để huấn luyện và đánh giá mô hình cho đến hết dữ liệu. Để thực nghiệm khả năng học thêm tri thức mới của mô hình, chúng tôi chia tập huấn luyện thành khoảng 5 phần theo số lớp và huấn luyện từng phần, đầu tiên huấn luyện mô hình với khoảng 20% số lớp, 40% số lớp cho tới khi 100% số lớp được huấn luyện. Tại mỗi bước huấn luyện thực hiện đánh giá kết quả phân lớp. Trong cả hai thí nghiệm này, mô hình sử dụng LNBNN kết hợp với đặc trưng SIFT trên phô tần số của tín hiệu tiếng nói. Bảng 3.5 so sánh độ chính xác phân lớp khi bổ sung thêm dữ liệu huấn luyện đối với tất cả các lớp trong khi Bảng 3.6 là so sánh độ chính xác của mô hình khi bổ sung thêm các lớp (tri thức mới) cho mô hình.

Bảng 3. 5 So sánh độ chính xác phân lớp khi bổ sung thêm dữ liệu huấn luyện cho tất cả các lớp

Database	20% training samples	40% training samples	60% training samples	80% training samples	100% training samples
ISOLET	0.46	0.56	0.60	0.68	0.73
English Digits	0.90	0.92	0.94	0.95	0.96
VN Places	0.91	0.92	0.93	0.94	0.95
TMW	0.73	0.77	0.80	0.82	0.83
JVPD	0.94	0.96	0.96	0.95	0.97

Bảng 3. 6 So sánh độ chính xác phân lớp khi bổ sung thêm lớp (tri thức) cho mô hình

Database	20% classes	40% classes	60% classes	80% classes	100% classes
ISOLET	0.55	0.64	0.60	0.60	0.73
English Digits	1.00	0.98	0.98	0.97	0.96
VN Places	1.00	0.97	0.95	0.94	0.95
TMW	1.00	0.98	0.96	0.90	0.83
JVPD	1.00	1.00	0.97	0.97	0.97

Trong bảng 3.5 cho thấy với số lớp đã cố định, khi bổ sung thêm mẫu huấn luyện cho mỗi lớp thì kết quả phân lớp càng tăng. Trong khi bảng 3.6 cho thấy khi bổ sung thêm lớp, tức là bổ sung thêm tri thức mới thì việc kết quả phân lớp giảm. Điều này là phù hợp với nguyên lý của học máy thống kê cũng như trong hoạt động nhận thức thực tiễn của con người. Từ thực nghiệm này, đã cho thấy mô hình phân lớp LNBNN phù hợp cho bài toán có tính chất thay đổi dữ liệu, hay nói cách khác LNBNN cho phép học tăng cường mà không phải huấn luyện lại toàn bộ mô hình.

3.8.6. Thí nghiệm với mạng tích chập trên tín hiệu tiếng nói

Như đã trình bày ở phần trên, mạng tích chập đã được ứng dụng rất thành công trong lĩnh vực thi giác máy. Một số tác giả đã sử dụng mạng tích chập kết hợp với mô hình HMM trực tiếp từ dữ liệu tiếng nói với kết quả giảm được sai số xuống còn 6%-10% đối với bộ dữ liệu TIMIT [Abdel-Hamid, 2014]. Dữ liệu trực tiếp từ dạng sóng ban đầu sẽ không thể hiện được các đặc trưng về tần số của tín hiệu một cách đầy đủ. Trong thực nghiệm này, thay vì sử dụng trực tiếp tín hiệu tiếng nói, chúng tôi biểu diễn tín hiệu tiếng nói ở dạng phổ tần số làm đầu vào cho mô hình.

Trong thí nghiệm này, chúng tôi chia dữ liệu thành hai bộ: tập huấn luyện và xác nhận lần lượt chiếm 80% và 20%. Mô hình đào tạo huấn luyện với 500 vòng. Chúng tôi tiến hành thay đổi kích thước dữ liệu đầu vào, số lớp và số nốt của từng lớp, các tham số của mô hình, rồi tiến hành so sánh các kết quả để tìm bộ tham số phù hợp nhất của mô hình với tập dữ liệu thực nghiệm. Do giới hạn về bộ nhớ của bộ xử lý đồ họa, nên số lớp tăng lên thì phải giảm số nốt trên các lớp và giảm kích thước của dữ liệu đầu vào. Kết quả tốt nhất khi mô hình là gồm các lớp như sau: C1: 64@26x26, M1: 64@13x13, C2: 64@11x11, M2:

64@5x5, C3: 64@3x3, M3 64@1x1, C4 54@1x1, số học 1.0, rho = 0.95, epsilon là 1e-6, với kích thước đầu vào là 28x28.

Bảng 3. 7 So sánh độ chính xác phân lớp của CNN và LNBNN kết hợp với SIFT trên phổ tần số của tín hiệu tiếng nói

Databases	CNN (Phổ tần số)	LNBNN (SIFT từ phổ tần số)
ISOLET (26 classes)	0.81	0.73
English Digits (10 classes)	0.98	0.96
Vietnamese Places (8 classes)	0.83	0.95
TMW (212 classes)	0.93	0.83
JVPD (5 classes)	0.95	0.97

Bảng 3.7 mô tả quả thực nghiệm của mô hình CNN đã trình bày trong mục 3.7 trên 05 bộ dữ liệu đồng thời so sánh kết quả với mô hình phân lớp LNBNN kết hợp với đặc trưng SIFT được trích chọn từ phổ tần số của tín hiệu tiếng nói. Mô hình được thực hiện trên máy tính có bộ xử lý Pentium core i5, 32 GB Ram với 02 GPU GTX 1050 2GB Ram chạy hệ điều hành Ubuntu 16.04.

Từ kết quả ở bảng 3.7 cho thấy mạng tích chập cho kết quả tốt hơn ở các bộ dữ liệu có nhiều lớp hơn so với LNBNN. Nhìn chung, mạng tích chập dựa trên phổ tần số của tín hiệu tiếng nói là một phương pháp phù hợp cho bài toán nhận thức tiếng nói nói chung, và bài toán nhận dạng tiếng nói độc lập nói riêng. Tuy nhiên, kích thước của của mạng tích chập là bao nhiêu lớp và mỗi lớp có bao nhiêu nốt thì tối ưu còn là một bài toán khó. Việc tăng kích thước của mạng đến một ngưỡng nào đó sẽ làm giảm tốc độ và giảm độ chính xác của mô hình đồng thời khó đáp ứng được về yêu cầu hạ tầng thuật. Kích thước và độ sâu của mạng nơ-ron khiến mạng nơ-ron trở nên không linh hoạt và khí có thêm lớp dữ liệu (tri thức mới), hoặc khi muốn thay đổi kiến trúc mạng thì mô hình phải huấn luyện lại từ đầu.

Do đó, trong thực tế, mô hình mạng tích chập có thể được sử dụng giải quyết các bài toán cụ thể khi mà nhu cầu mở rộng không quá lớn và không đòi hỏi độ linh hoạt cao.

3.9. Kết luận

Trong chương này, chúng tôi đã đề xuất phương pháp trích chọn đặc trưng tiếng nói ở mức độ thính giác dựa trên phổ tần số của tín hiệu tiếng nói đồng thời kết hợp với phương pháp phân lớp LNBNN, phương pháp phân lớp phi tham số có ưu điểm là cho phép mô hình có thể học thêm mẫu dữ liệu huấn luyện, học thêm tri thức mà không phải huấn luyện lại. Chương này, cũng đề xuất sử dụng mô hình mạng tích chập dựa trên phổ tần số của tín hiệu tiếng nói cho bài toán nhận dạng tiếng nói, một bài toán chính trong bài toán nhận thức tiếng nói.

Các kết quả thực nghiệm cho thấy việc trích chọn đặc trưng SIFT kết hợp với LNBNN cho kết quả cao hơn so với việc sử dụng đặc trưng MFCC kết hợp với các phương pháp phân lớp khác thậm chí là kết hợp đặc trưng MFCC với LNBNN. Nhưng so sánh phương pháp phân lớp LNBNN kết hợp với SIFT với mô hình tích chập trên phổ tần số của tín hiệu tiếng nói thì mô hình tích chập cho kết quả tốt hơn.

Mô hình LNBNN có ưu điểm là cho phép học tăng cường mà không phải huấn luyện lại, còn mô hình CNN phải huấn luyện lại khi bổ sung thêm dữ liệu huấn luyện. Mô hình LNBNN có nhược điểm là phải lưu toàn bộ tập đặc trưng của dữ liệu huấn luyện từ đó dẫn đến khó khăn trong việc lưu trữ đặc trưng, giảm tốc độ tính toán của mô hình đồng thời việc sử dụng các đặc trưng SIFT một cách độc lập theo mô hình Bag-of-feature này không mô tả được đặc trưng mang tính liên kết của tín hiệu tiếng nói. Trong khi đó, mô hình CNN lại đòi hỏi hệ thống máy tính phải có khả năng tính toán cao, thông thường phải có bộ xử lý chuyên dụng như bộ xử lý đồ họa (GPU) mới thực hiện được.

Từ đó cho thấy, cả mô hình LNBNN kết hợp với đặc trưng SIFT, và mô hình tích chập dựa trên phổ tần số của tín hiệu tiếng nói đều có khả năng áp dụng vào bài toán nhận thức tiếng nói. Tùy từng điều kiện kỹ thuật khác nhau, chúng ta có thể sử dụng mô hình LNBNN kết hợp với SIFT hoặc mô hình CNN.

Kết quả nghiên cứu nêu trên được công bố tại kỷ yếu có phản biện của Hội nghị quốc tế 2015 2nd National Foundation for Science and Technology Development Conference on Information and Computer Science, NIC'S 2015

(công trình khoa học số 1) và tạp chí khoa học có phản biện Vietnam Journal of Computer Science, Volume 3, Number 4, November 2016, SpringerOpen (công trình khoa học số 2).

Chương 4. MÔ HÌNH NHẬN THỨC TIẾNG NÓI THÔNG QUA HỌC MỐI QUAN HỆ GIỮA TÍN HIỆU TIẾNG NÓI VÀ HÌNH ẢNH

4.1. Giới thiệu

Nhận thức là việc tổ chức, xác định và diễn giải thông tin từ các giác quan để biểu diễn và hiểu môi trường xung quanh [Schacter, 2011]. Nhận thức liên quan đến các tín hiệu trong hệ thần kinh mà nó là kết quả từ sự kích thích vật lý hay hóa học của các cơ quan giác quan. Nhận thức không chỉ là sự tiếp nhận thụ động của những tín hiệu, mà nó còn được định hình thông qua việc học tập, ghi nhớ, phán đoán và sự tập trung. Nhận thức có thể được chia thành hai quá trình. Quá trình thứ nhất, xử lý các tín hiệu đầu vào từ các giác quan, biến đổi thông tin ở mức thấp này thành thông tin mức cao. Quá trình thứ hai, kết nối các tín hiệu đầu vào với các khái niệm và phán đoán tri thức của con người và các cơ chế chọn lọc có ảnh hưởng đến nhận thức.

Nhận thức âm thanh là khả năng cảm nhận âm thanh bằng cách phát hiện những rung động trong phạm vi tần số mà con người có khả năng nhận thức được. Giải tần số mà con người nghe được là từ 20Hz và 20.000 Hz [Rosen, 2011]. Hệ thính giác của con người bao gồm tai ngoài có chức năng thu thập các sóng âm và lọc âm thanh, tai giữa có chức năng chuyển áp lực âm thanh, và tai trong có chức năng sản xuất các tín hiệu thần kinh để đáp ứng với âm thanh. Bằng con đường thính giác tăng dần chúng được dẫn đến vỏ não thính giác chính trong thùy thái dương của não bộ con người, đó là nơi mà các thông tin thính giác đến trong vỏ não và được tiếp tục xử lý ở đó. Tiếng nói là một dạng âm thanh đặc biệt được tạo ra bởi con người. Nhận thức tiếng nói là quá trình mà qua đó ngôn ngữ nói được nghe, giải thích và hiểu rõ. Nghiên cứu trong nhận thức tiếng nói là tìm hiểu làm thế nào người nghe có thể nhận ra âm thanh tiếng nói và sử dụng thông tin này để hiểu ngôn ngữ nói.

Có nhiều cách tiếp cận khác nhau khi nghiên cứu về nhận thức của con người. Các nhà tâm lý học định lượng nghiên cứu mối quan hệ giữa các chất lượng vật lý của tín hiệu đầu vào từ giác quan và nhận thức trong khi các nhà khoa học thần kinh cảm giác nghiên cứu các cơ chế não dưới nhận thức. Các hệ thống nhận thức cũng có thể được nghiên cứu theo mô hình tính toán theo các thông tin mà chúng xử lý. Các vấn đề của nhận thức trong triết học bao gồm

mức độ của các đặc tính mà giác quan thu được như âm thanh, mùi vị, hay màu sắc, kích thước,.. tồn tại trong thực tế khách quan [Gregory, 1987].

Như vậy, có thể nói nhận thức tiếng nói là một phần trong nghiên cứu nhận thức của con người. Nhận thức tiếng nói không thể đứng độc lập so với nhận thức thông tin từ các giác quan khác mà phải trực tiếp hoặc gián tiếp thông qua các giác quan khác. Như vậy, nghiên cứu nhận thức tiếng nói xét trên một phương diện nào đó là nghiên cứu mối liên hệ thông tin thu được từ hệ thính giác và thông tin thu được từ các giác quan khác. Một trong những giác quan đem lại nhiều thông tin nhất cho người nhận thức đó là hình ảnh, vì vậy bài toán nhận thức tiếng nói là tìm ra mối liên hệ giữa tín hiệu âm thanh và tín hiệu hình ảnh thu được từ môi trường xung quanh. Nói cách khác, quá trình nhận thức tiếng nói được thực hiện thông qua việc học ánh xạ, hay mối quan hệ giữa một từ, cụm từ (một khái niệm) về sự vật hiện tượng với hình ảnh thu được của sự vật hiện tượng đó. Từ đó, người học sẽ xây dựng được một ánh xạ giữa các đặc trưng thu được từ hệ thính giác với các đặc trưng thu được từ thị giác và các giác quan khác.

Trong chương này, chúng tôi xây dựng mô hình nhận thức tiếng nói thông qua việc học mối quan hệ giữa các đặc trưng từ một cặp dữ liệu tiếng nói và hình ảnh xảy ra đồng thời mà người học thu nhận được thông qua hai cơ quan cảm giác chính đó là thính giác và thị giác. Mô hình nhận thức này sẽ có ý nghĩa trong lĩnh vực học máy nói chung và lĩnh vực điều khiển người máy nói riêng.

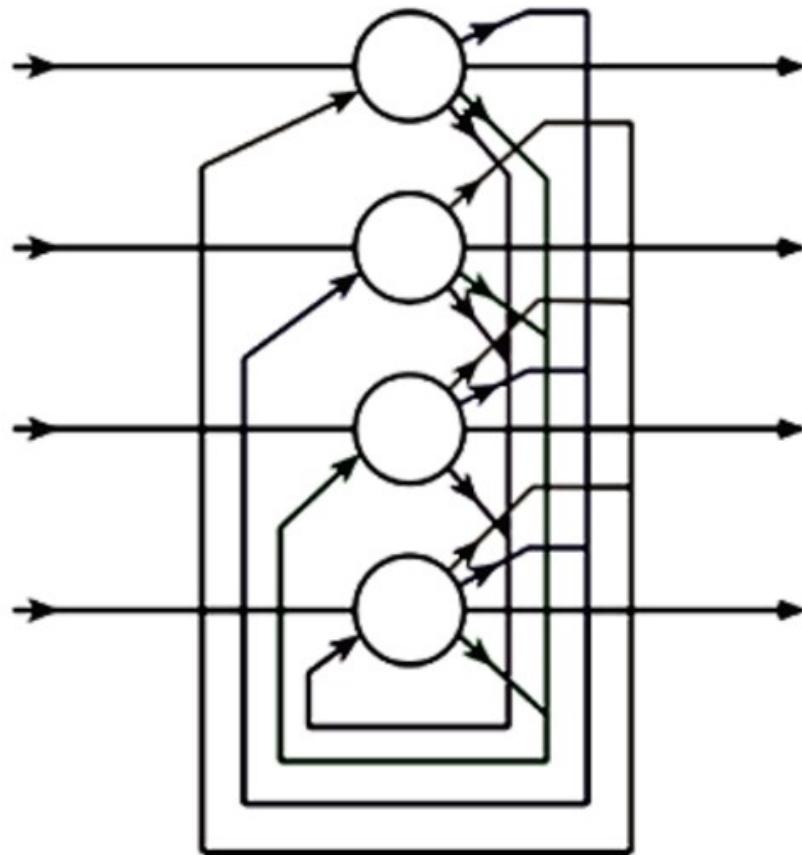
Kết quả của chương chứng minh một hướng tiếp cận mới trong lĩnh vực điều khiển người máy, hướng xây dựng mô hình huấn luyện người máy có được cách học để nhận thức các sự vật một cách tự nhiên như quá trình học của con người.

4.2. Các phương pháp học mối quan hệ

4.2.1. Học mối quan hệ bằng mạng nhân tạo

Mạng Hopfield [Raul, 1996] là mô hình điển hình của lớp mạng lan truyền ngược. Mạng Hopfield là mạng một lớp có rất nhiều ứng dụng, đặc biệt trong bộ nhớ liên kết và trong các bài toán tối ưu. Tín hiệu ra của nơ-ron thứ j

nào đó được truyền ngược lại làm tín hiệu vào cho các nơ-ron thông qua các trọng số tương ứng.

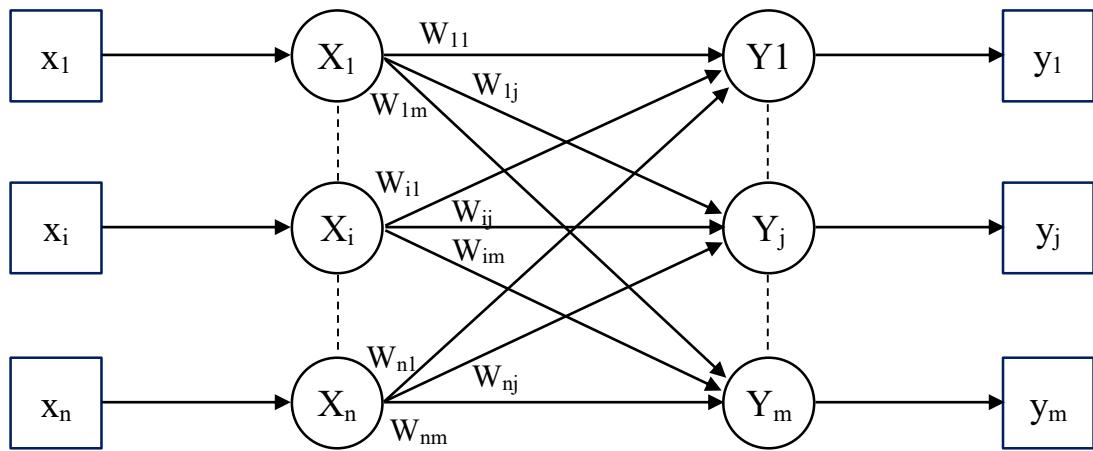


Hình 4. 1 Mô hình mạng Hopfield [Raul, 1996]

Như vậy, mạng Hopfield là một mô hình học liên kết giữa các nốt trong mạng hay gọi là liên kết nội tức là liên kết trong một miền dữ liệu. Xét trong bài toán học mối quan hệ, thì Hopfield là mô hình mạng cho phép học mối quan hệ của một miền dữ liệu, hay là giữa các mẫu của một miền dữ liệu. Cụ thể, đối với bài toán nhận thức tiếng nói dựa trên mô hình học mối quan hệ giữa tiếng nói và hình ảnh, thì mô hình mạng Hopfield chỉ có thể áp dụng được trong việc học mối quan hệ của các mẫu tiếng nói, hoặc là mối quan hệ của các mẫu hình ảnh. Vì vậy, mạng Hopfield chủ yếu được sử dụng trong lĩnh vực nhận dạng mẫu như nhận dạng hình ảnh, nhận dạng tiếng nói.

Mạng nhớ liên kết hai chiều BAM [Kosko, 1988] [Kosko, 1987] là một mạng nhớ thể hiện cấu trúc bộ nhớ liên kết với khả năng nhớ lại theo cả hai chiều. Mạng BAM được cấu tạo từ hai mạng Hopfield để thực hiện liên kết giữa hai mẫu dữ liệu. Mạng BAM có hai dạng gồm tự liên kết (khi mẫu vào và

mẫu ra trong một cặp là giống nhau) và liên kết khác loại (khi mẫu vào và mẫu ra trong một cặp là khác nhau).



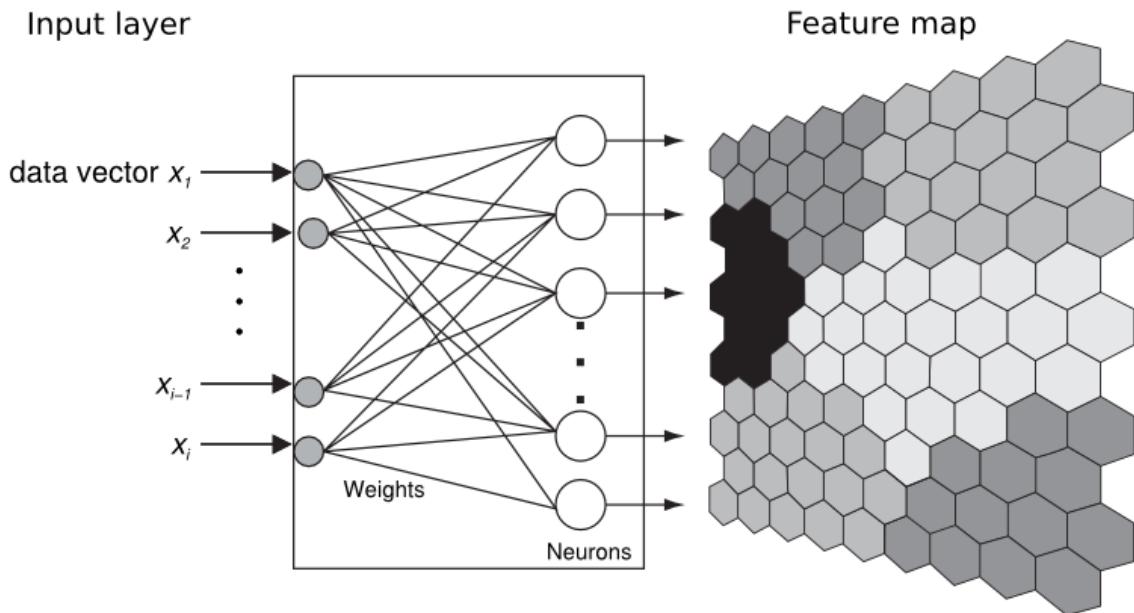
Hình 4. 2 Mô hình mạng BAM [Kosko, 1987]

Trong mô hình này, BAM lưu p liên kết khác loại giữa hai trường X và Y. Các cặp mẫu được ký hiệu là: $(X_1, Y_1), \dots, (X_p, Y_p)$. Khi cung cấp mẫu vào từ trường X thì mạng BAM sẽ nhớ lại mẫu đã lưu ở trường Y. Ngược lại, cung cấp mẫu vào từ trường Y thì thu được mẫu ra ở trường A.

Như vậy thực chất là mạng BAM đã học liên kết giữa các cặp dữ liệu và lưu trữ các liên kết đó trong một ma trận trọng số. Các cặp dữ liệu được đưa vào mô hình mạng BAM chính là các cặp dữ liệu có mối quan hệ. Và sau khi huấn luyện mạng BAM sẽ cho phép trả lại một mẫu đầu ra gần nhất với mẫu dữ liệu đầu vào trong các cặp mẫu đã huấn luyện. Tuy nhiên, các mẫu dữ liệu đổi với mạng BAM ở hai trường X và Y đều phải được biểu diễn thành véc tơ chứa các giá trị ở dạng hai cực gồm hai giá trị 0 và 1 hay -1 và 1 và dữ liệu đổi hỏi phải có tính trực giao đối với mỗi cặp mẫu huấn luyện.

Bản đồ tự tổ chức [Kohonen, 1982] (Self Organizing Map-SOM) là một mạng Neuron nhân tạo, được huấn luyện sử dụng kỹ thuật học không giám sát để biểu diễn dữ liệu với số chiều thấp hơn nhiều so với dữ liệu đầu vào nhiều chiều. Kết quả của mạng SOM gọi là bản đồ (Map). SOM là một mạng nhân tạo, tuy nhiên mạng tự tổ chức SOM khác với các mạng nhân tạo khác là không sử dụng các lớp ẩn mà chỉ có hai lớp là lớp đầu vào và lớp đầu ra. Mạng SOM sử dụng khái niệm láng giềng để giữ lại đặc trưng của các dữ liệu đầu vào trên bản đồ, nghĩa là các mẫu huấn luyện tương tự nhau thì được đặt gần nhau trên

bản đồ. Ưu điểm chính của SOM là biểu diễn trực quan dữ liệu nhiều chiều vào không gian ít chiều hơn (thường là 2 chiều) và đặc trưng của dữ liệu đầu vào được giữ lại trên bản đồ.



Hình 4. 3 Mô hình mạng tự tổ chức [Kohonen, 1982]

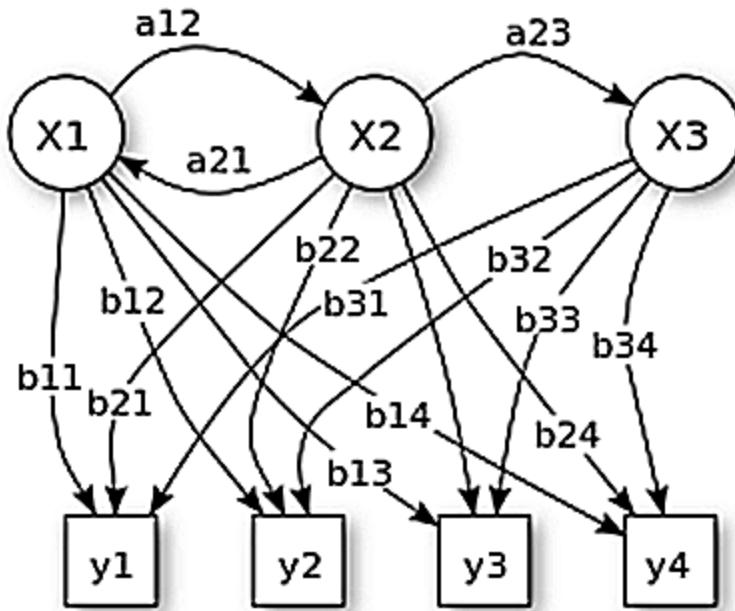
Như vậy, mạng SOM cũng là một mô hình học mồi liên kết giữa các đặc trưng, ở đây mạng SOM có thể coi là mạng liên kết giữa một miền dữ liệu nhiều chiều và một bên là dữ liệu ít chiều hơn. Việc học mồi liên kết ở đây chính là việc học ma trận trọng số liên kết hay ma trận chiếu từ không gian dữ liệu đầu vào tới không gian ít chiều hơn ở dữ liệu đầu ra.

4.2.2. Học mồi quan hệ bằng HMM

Mô hình Markov ẩn [Baum, 1966] (Hidden Markov Model - HMM) là mô hình thống kê trong đó hệ thống được mô hình hóa được cho là một quá trình Markov với các tham số không biết trước và nhiệm vụ là xác định các tham số ẩn từ các tham số quan sát được, dựa trên sự thừa nhận này. Các tham số của mô hình được rút ra sau đó có thể sử dụng để thực hiện các phân tích kế tiếp, ví dụ cho các ứng dụng nhận dạng mẫu.

Trong một mô hình Markov điển hình, trạng thái được quan sát trực tiếp bởi người quan sát, và vì vậy các xác suất chuyển tiếp trạng thái là các tham số duy nhất. Mô hình Markov ẩn thêm vào các điều ra: mỗi trạng thái có xác suất phân bố trên các biểu hiện đầu ra có thể. Vì vậy, nhìn vào dãy của các biểu hiện được sinh ra bởi HMM không trực tiếp chỉ ra dãy các trạng thái.

Trong mô hình này, HMM sẽ học và đoán nhận mối liên kết giữa các quan sát với một trạng thái nào đó của mô hình, và mối liên kết này được học và lưu trữ trong ma trận xác suất chuyển trạng thái của mô hình.



Hình 4. 4 Mô hình HMM [Baum, 1966]

Trong đó X_i là các trạng thái trong mô hình HMM, a_{ij} là các xác suất chuyển trạng thái, b_{ij} là các xác suất đầu ra và y_i là các dữ liệu quan sát.

4.2.3. Học mối quan hệ dựa trên luật

Phương pháp học mối quan hệ dựa trên luật thường được sử dụng trong bài toán học mối quan hệ giữa các thực thể trong một văn bản. Một hệ thống học mối quan hệ dựa trên luật phụ thuộc chủ yếu vào các luật đã được xây dựng bằng cách thủ công hoặc tự động học từ dữ liệu. Hệ thống học dựa trên luật bao gồm hai phần: một tập hợp các quy tắc và một bộ chính sách để kích hoạt những quy tắc này.

Quy tắc cơ bản được biểu diễn theo mẫu như sau:

Pattern Contextual → Action.

4.2.4. Học mối quan hệ dựa trên thống kê

Theo cách tiếp cận thống kê [Wróblewska, December 4-7, 2012] , học mối quan hệ là sự phân tích văn bản phi cấu trúc và gán nhãn các phần khác nhau của văn bản. Phương pháp này được sử dụng trong học mối quan hệ trong văn bản. Sự phân tích quan hệ có thể được thực hiện theo các cách sau:

- Gán nhãn cho các tokens, khôi từ, phân đoạn

Chúng ta biểu thị dãy các token như $x = x_1, x_2, \dots, x_n$.

Tại thời điểm trích xuất, mỗi token xi từ câu được phân vào một tập Z của nhãn và cho kết quả là một chuỗi nhãn $y = y_1, y_2, \dots, y_n$

Tập Z bao gồm các loại thực thể E và một nhãn đặc biệt “khác” cho các mã thông báo không thuộc bất kỳ loại thực thể nào khác.

Có thể áp dụng kiểu mã hoá BCOE (Begin, Continue, Other, End).

Trong mô hình này, các tính năng được định nghĩa qua các phân đoạn bao gồm nhiều tokens tạo thành một chuỗi thực thể. Bằng cách này các tính năng có thể nắm bắt các thuộc tính chung trên tất cả các thẻ tạo thành một phần của một thực thể.

- Các phương pháp dựa trên ngữ pháp

Một số hệ thống khai thác thực thể đòi hỏi phải giải thích tốt hơn về cấu trúc của nguồn. Một mô hình dựa trên ngữ pháp sử dụng một bộ quy tắc dẫn xuất, như trong văn phạm phi ngữ cảnh (Context Free Grammar - CFG), để thể hiện cấu trúc toàn cục của thực thể. Ví dụ, để nắm bắt tính đồng nhất giữa các tên tác giả trong trích dẫn, chúng ta có thể xác định một tập hợp các quy tắc dẫn xuất như thể hiện trong hình 4.5:

```
R: S -> AuthorsLF | AuthorsFL
R0: AuthorsLF -> NameLF_Separator AuthorsLF
R1: AuthorsFL -> NameFL_Separator AuthorsFL
R2: AuthorsFL -> NameFL
R3: AuthorsLF -> NameLF
```

Hình 4. 5 Ví dụ các luật theo văn phạm phi ngữ cảnh

Trong đó: R là các quy tắc dẫn xuất; S là ký hiệu bắt đầu; AuthorsLF, AuthorsFL là các ký hiệu không kết thúc; NameLF_Separator, NameFL, NameLF là các ký hiệu kết thúc.

- Trích xuất quan hệ với các phương pháp dựa trên đặc trưng

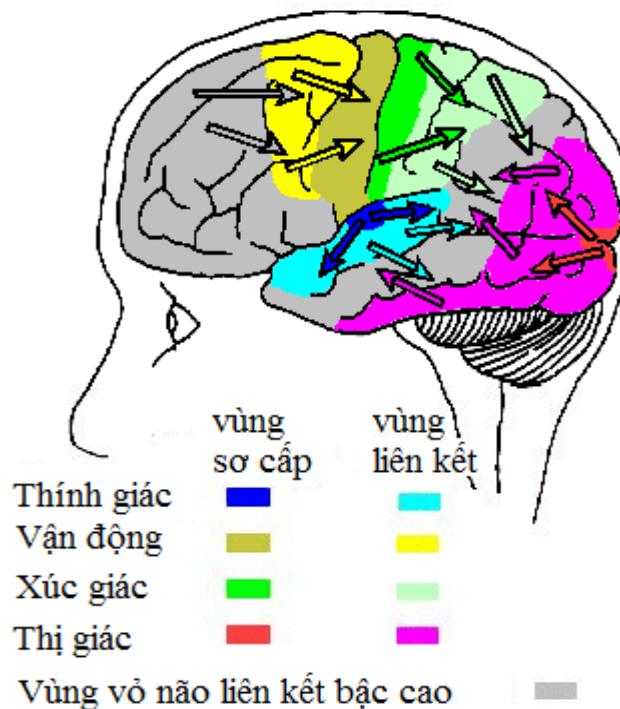
Hai hoặc nhiều thực thể có quan hệ với nhau với một số mối quan hệ được xác định trước. Ví dụ, quan hệ xác định trước “is employee of” là mối quan hệ giữa một thực thể là tên người và một thực thể là tên tổ chức. Mọi quan

hệ có thể là nhị phân (giữa hai thực thể) hoặc chúng có thể được đa chiều (giữa nhiều thực thể).

4.3. Đề xuất mô hình nhận thức tiếng nói

4.3.1. Cơ sở đề xuất mô hình

Vỏ não là lớp vỏ ngoài của chất xám trên bán cầu. Một số vùng vỏ não có chức năng đơn giản hơn, gọi là vỏ não sơ cấp [Wanda, 2017] [Milner, 1995]. Chúng bao gồm các khu vực trực tiếp nhận thông tin từ các cơ quan giác quan như thị giác, thính giác, cảm giác thần kinh hoặc trực tiếp tham gia điều khiển vận động chân tay hoặc sự chuyển động của mắt. Vùng vỏ não liên kết có các chức năng phức tạp hơn vùng vỏ não sơ cấp. Vùng vỏ não liên kết được chia làm hai loại là vùng vỏ não liên kết của các cơ quan cảm giác và vùng vỏ não liên kết đa giác quan.

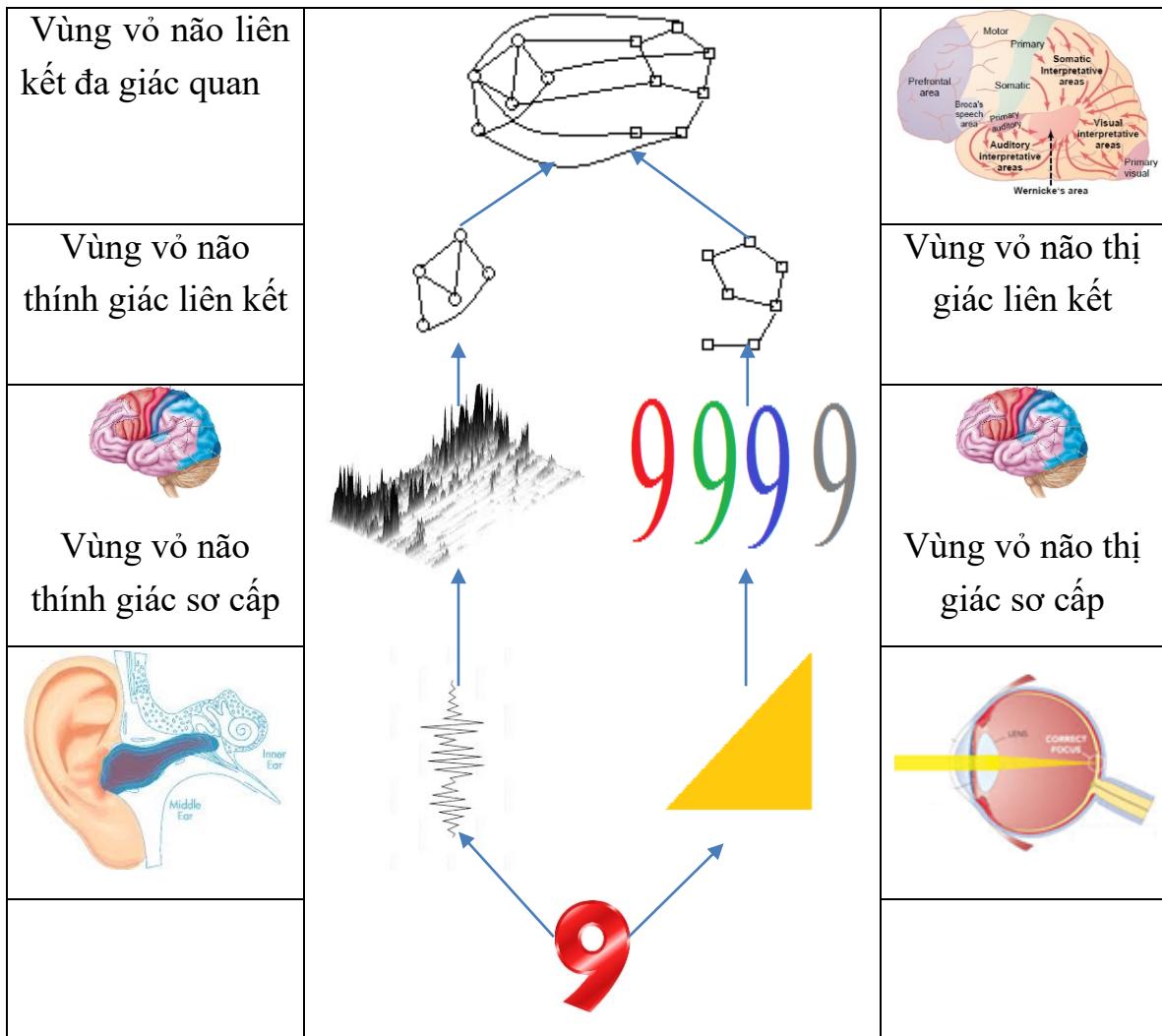


Hình 4.6 Sơ đồ các vùng vỏ não sơ cấp và vùng vỏ não liên kết¹⁰

Như vậy, con người có được nhận thức là nhờ vào sự thiết lập ánh xạ giữa các đặc trưng thu nhận được thông qua các giác quan trong đó lượng thông tin thu được nhiều nhất ở thị giác, tiếp đến dựa trên số lượng nơ-ron thần kinh cần thiết để tiếp nhận và xử lý thông tin từ các giác quan này [Leuba, 1994].

¹⁰ http://www.indiana.edu/~p1013447/dictionary/assn_cor.htm

Từ những phân tích đó, có thể coi quá trình nhận thức ở con người là quá trình học mối quan hệ, mối liên kết giữa các tập thuộc tính, hay đặc trưng của các sự vật, hiện tượng trong thế giới khách quan.



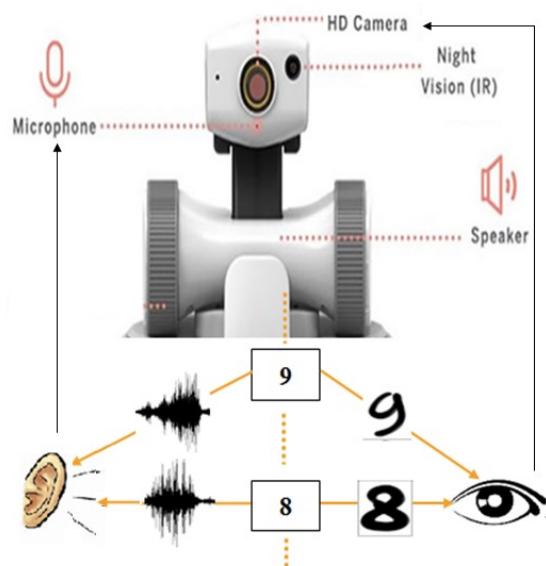
Hình 4. 7 Ví dụ minh họa tập dữ liệu thực nghiệm DIGITS

Theo hướng tiếp cận này, để máy tính nhận thức hay hiểu được tiếng nói thực chất là xây dựng được mạng quan hệ giữa tín hiệu tiếng nói với thông tin về các sự vật hiện tượng thu được từ các giác quan khác. Từ đó, chúng tôi đề xuất một mô hình nhận thức tiếng nói thông qua việc học mối liên kết giữa tiếng nói với hình ảnh là hai kênh thu nhận thông tin chính ở con người.

Tín hiệu âm thanh của một lớp trừu tượng nào đó sẽ được nhận thức bởi một số bởi một số đặc trưng nhất định. Tương tự vậy, các tín hiệu hình ảnh của cùng một lớp trừu tượng cũng sẽ được nhận thức bởi một số đặc trưng chung nhất định nào đó. Và nhận thức tiếng nói là quá trình xây dựng được mạng quan

hệ giữa các đặc trưng này. Trong mô hình học mồi quan hệ giữa tín hiệu tiếng nói và hình ảnh, chúng tôi sử dụng đặc trưng SIFT cho cả tín hiệu tiếng nói và hình ảnh. Trong mô hình ánh xạ giữa tín hiệu tiếng nói và hình ảnh chúng tôi đề xuất sử dụng mạng tích chập để trích chọn đặc trưng của tiếng nói dựa trên phổ tần số, và trích chọn đặc trưng hình ảnh trực tiếp.

Hình 4.7 mô phỏng quá trình nhận thức tiếng nói. Khi tín hiệu tiếng nói được hệ thính giác thu nhận, các tín hiệu này được các giác quan xử lý để biến đổi thành các đặc trưng tần số tương ứng. Các tần số đạt được năng lượng nhất định sẽ kích thích các sợi sinh học và truyền một xung điện đến vùng vỏ não thính giác sơ cấp để thu nhận đặc trưng phổ tần số theo thời gian của tín hiệu tiếng nói. Các thông tin này, sau đó sẽ tạo các thông tin liên kết giữa các đặc trưng này để biểu diễn tín hiệu tiếng nói ở vùng não liên kết thính giác. Đồng thời với quá trình này, tín hiệu hình ảnh cũng được hệ thị giác thu nhận và các đặc trưng của tín hiệu hình ảnh lưu trữ ở vùng não thị giác sơ cấp, vùng vỏ não liên kết thị giác. Tiếp theo, cả hai luồng thông tin này sẽ truyền tín hiệu tới vùng vỏ não liên kết đa giác quan để tạo liên kết giữa các đặc trưng thu được.



Hình 4.8 Mô hình nhận thức tiếng nói cho người máy

Mô hình nhận thức tiếng nói có thể áp dụng trong lĩnh vực điều khiển người máy bằng cách trang bị cho người máy hệ thống cảm biến âm thanh và hình ảnh như hình 4.8.

4.3.2. Mô hình nhận thức tiếng nói dựa trên học quan hệ giữa tín hiệu âm thanh và tín hiệu hình ảnh

Định nghĩa 1: quan hệ giữa một mẫu tiếng nói và một mẫu hình ảnh

Một mẫu tiếng nói thu được từ hệ thính giác đồng thời với một hình ảnh của sự vật, hiện tượng từ môi trường xung quanh tại cùng một thời điểm thì được gọi là có quan hệ (quan hệ cùng xuất hiện).

Ở đây các mẫu tín hiệu tiếng nói và hình ảnh đều được biểu diễn bằng một tập các véc tơ đặc trưng nào đó. Các véc tơ đặc trưng của các mẫu tiếng nói khác nhau thì có cùng cấu trúc. Các véc tơ đặc trưng của các mẫu hình ảnh là có cùng cấu trúc. Véc tơ đặc trưng của tiếng nói có thể có cấu trúc khác với véc tơ đặc trưng của mẫu hình ảnh. Ví dụ tiếng nói có thể được biểu diễn bằng tập các hệ số MFCC, LPC, SIFT,.. Trong khi tập đặc trưng của hình ảnh có thể là SIFT, SURF, HOG,... Từ đó, ta định nghĩa quan hệ giữa một đặc trưng của tiếng nói với một đặc trưng của hình ảnh.

Định nghĩa 2. Quan hệ một đặc trưng tiếng nói với một đặc trưng hình ảnh

Giả sử có một mẫu tiếng nói S được biểu diễn bằng một tập các đặc trưng $\{f_1, f_2, \dots\}$, và một mẫu hình ảnh I được biểu diễn bởi tập đặc trưng $\{g_1, g_2, \dots\}$. Khi đó đặc trưng f_i và đặc trưng g_j được gọi là có quan hệ nếu S có quan hệ với I .

$$R(f_m, g_n) = \begin{cases} 1 & \text{if } S \text{ relate to } I \\ 0 & \text{if } S \text{ does not relate to } I \end{cases} \quad (4.1)$$

Như vậy bài toán có thể được mô hình hóa như sau: Cho một tập dữ liệu huấn luyện là một tập các cặp mẫu gồm một tín hiệu tiếng nói và một hình ảnh mà hai giác quan thu được tại cùng một thời điểm. Như vậy mỗi mẫu huấn luyện là một cặp $\langle S_i, I_i \rangle$. Như vậy, khi cho một mẫu mới là một cặp $\langle S, I \rangle$ bất kỳ, hỏi cặp mẫu $\langle S, I \rangle$ này là có quan hệ với nhau hay không?

Để áp dụng được phương pháp phân lớp LNBNN cho bài toán này có ba vấn đề cần phải giải quyết đó là: Vấn đề 1 là xây dựng được véc tơ đặc trưng mô tả mối quan hệ giữa 2 đặc trưng của tiếng nói và hình ảnh. Vấn đề thứ 2 là làm thế nào để ước lượng được khoảng cách giữa 2 véc tơ đặc trưng đó và bài

toàn thứ 3 là làm thế nào để tìm được K hàng xóm gần nhất của một đặc trưng đầu vào bất kỳ. Trong nghiên cứu này chúng tôi đề xuất một phương pháp tổ chức các véc tơ đặc trưng thành hai tập đặc trưng riêng của từng loại dữ liệu. Khi đó khoảng cách giữa hai véc tơ đặc trưng kết hợp sẽ được tính như sau:

Giả sử f^1 và f^2 là hai véc tơ đặc trưng kết hợp như sau:

$$f^1 = \langle fS^1, fI^1 \rangle = \langle fS_1^1, fS_2^1, \dots, fS_{d1}^1, fI_1^1, fI_2^1, \dots, fI_{d2}^1 \rangle$$

$$f^2 = \langle fS^2, fI^2 \rangle \geq \langle fS_1^2, fS_2^2, \dots, fS_{d1}^2, fI_1^2, fI_2^2, \dots, fI_{d2}^2 \rangle$$

Khi đó khoảng cách giữa hai véc tơ f^1 và f^2 sẽ là

$$d(f^1, f^2) = d(\langle f_S^1, f_I^1 \rangle, \langle f_S^2, f_I^2 \rangle) = d_S^2 + d_I^2 \quad (4.2)$$

ở đây d_S là khoảng cách từ véc tơ thành phần của tiếng nói từ véc tơ đặc trưng kết hợp f^1 đến f^2 , và d_I là khoảng cách từ thành phần hình ảnh của f^1 đến f^2 .

Đồng thời chúng tôi xây dựng một ma trận trọng số mô tả mối quan hệ giữa các cặp đặc trưng. Ma trận trọng số được xây dựng như sau:

$$W = \{w_{ij}\} = \sum R(f_i, g_j) \quad \forall f_i \in \{S\}, \forall g_j \in \{I\}. \quad (4.3)$$

Ma trận trọng số W khi đó sẽ lưu số lần xuất hiện của mỗi cặp đặc trưng có mối quan hệ. Như vậy, nếu cặp đặc trưng nào có trọng số là 0 nghĩa là cặp đặc trưng đó không có mối quan hệ. Cặp đặc trưng nào có trọng số khác 0 nghĩa là có mối quan hệ. Trọng số càng lớn thể hiện cặp đặc trưng đó xuất hiện nhiều lần trong dữ liệu huấn luyện hay mật độ của đặc trưng đó lớn.

Ma trận trọng số được xây dựng theo thuật toán 4.1. Thuật toán sẽ sử dụng tất cả các đặc trưng trong tập dữ liệu huấn luyện để xây dựng hai cây KD-TREE, một cây cho dữ liệu tiếng nói ký hiệu là SP-KDTREE, một cây cho dữ liệu hình ảnh ký hiệu là IM-KDTREE.

Trong pha phân lớp, dữ liệu đầu vào của thuật toán là hai cây KD-TREE lưu trữ tập các điểm đặc trưng của tiếng nói và hình ảnh tương ứng, ma trận trọng số quan hệ đồng xuất hiện giữa các cặp đặc trưng, một cặp mẫu truy vấn

mới và một tham số ngưỡng cho trước. Thuật toán sẽ trả về kết quả là cặp đặc trưng mới đó có quan hệ hay không? (Thuật toán 4.2)

Thuật toán 4. 1 Thuật toán học mối quan hệ RELATION- Pha huấn luyện

Thuật toán: RELATION - Pha huấn luyện
Đầu vào: S là tập dữ liệu tiếng nói, I là tập dữ liệu hình ảnh
Đầu ra: Ma trận trọng số quan hệ w
1: Tạo cây SP-KDTREE cho tập các đặc trưng của tiếng nói 2: Tạo cây IM_KDTREE cho tập các đặc trưng của hình ảnh 3: For each sp in S 4: For each im in I 5: If related(sp, im) then 6: Search S_index in SP_KDTREE 7: Search I_index in IM_KDTREE 8: For each i in S_index 9: For each j in I_index 10: w(i,j) = w(i,j) + 1 11: Endfor 12: End for 13:End If

Thuật toán 4.2 được mô tả như sau:

Bước 1: Tìm K+1 hàng xóm gần nhất của thành phần tiếng nói trong tập tất cả các véc tơ đặc trưng của tiếng nói từ dữ liệu huấn luyện.

Bước 2: Tìm K+1 hàng xóm gần nhất của thành phần hình ảnh trong tập tất cả các véc tơ đặc trưng của hình từ dữ liệu huấn luyện.

Bước 3: Tìm K+1 hàng xóm gần nhất của véc tơ đặc trưng truy vấn trong KxK cặp đặc trưng được kết hợp từ kết quả của Bước 1 và Bước 2.

Bước 4: Tính khoảng cách biên, là khoảng cách từ véc tơ truy vấn đến hàng xóm thứ K+1 trong KxK hàng xóm gần nhất tìm được.

Bước 5: Tìm khoảng cách nhỏ nhất từ véc tơ truy vấn tới các lớp của véc tơ đặc trưng trong K hàng xóm gần nhất. Cập nhật hiệu khoảng cách từ khoảng cách biên tới khoảng cách nhỏ nhất đến mỗi lớp tìm được trong K hàng xóm gần nhất.

Bước 6: Nếu tổng khoảng cách tới lớp + nhỏ hơn thì cặp véc tơ truy vấn là có quan hệ, ngược lại thì không có quan hệ.

Thuật toán 4. 2 Thuật toán học mối quan hệ RELATION - Phân lớp

Thuật toán: RELATION -Phân lớp

Đầu vào:

SF: cây đặc trưng của dữ liệu huấn luyện tiếng nói

IF: cây đặc trưng của dữ liệu huấn luyện hình ảnh

W: Ma trận trọng số quan hệ

{sp, im}: một cặp mẫu truy vấn {speech, image}

Threshold: tham số ngưỡng

Đầu ra: cặp mẫu truy vấn {sp, im} có quan hệ hay không

```

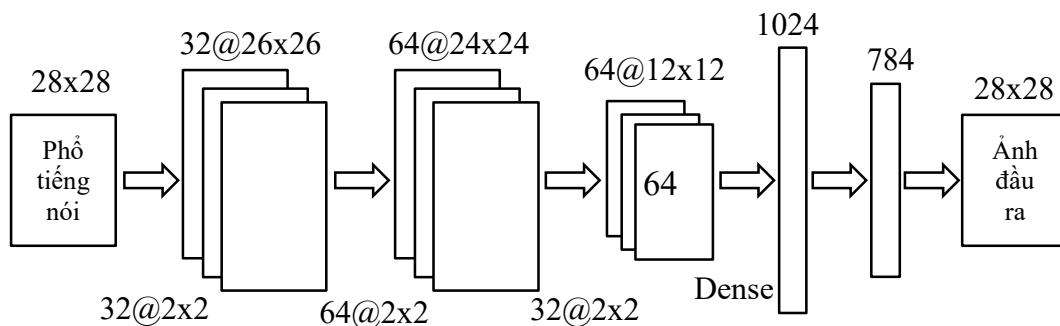
1: TotalWeight = 0;
2: Tìm tập SP_index là K+1 hàng xóm gần nhất của các đặc trưng của mẫu tiếng nói trong
cây SF
3: Tìm tập IM_index là chỉ số của K+1 hàng xóm gần nhất của các đặc trưng trong mẫu
hình ảnh trong cây IM
4: For each i in SP_index
5:   For each j in IM_index
6:     Tính distB khoảng cách tới cặp biên gần thứ K+1 của tiếng nói và hình ảnh
7:     For ik=1 to K
8:       For jk=1 to K
9:         Tính distC khoảng cách của cặp ngắn nhất có trong ma trận trọng số w
10:        End for
11:      End for
12:      TotalWeight = TotalWeight + w(i,j)*(distC - distB)/(N*M)
13:    End for
14:  End for
15 If TotalWeight < Threshold Then
16   return true
17 Else if
18   return false
19 End if

```

4.3.3. Mô hình nhận thức tiếng nói dựa trên ánh xạ giữa tín hiệu âm thanh và tín hiệu hình ảnh bằng mạng tích chập

Như đã đề cập trong phần 2.2.4 về mạng tích chập, ngày nay, mạng tích chập đã được ứng dụng rất thành công trong các bài toán nhận dạng ảnh, nhận dạng đối tượng trong lĩnh vực thị giác máy; Nhận dạng tiếng nói trong lĩnh vực nhận thức tiếng nói; và trong lĩnh vực xử lý ngôn ngữ tự nhiên. Bản chất của mô hình mạng tích chập là mạng truyền tới nhiều lớp trong đó có một số lớp tích chập và lớp lấy mẫu. Lớp cuối cùng là lớp phân lớp được kết nối đầy đủ tới véc-tơ đầu ra của mô hình. Các lớp tích chập và lớp lấy mẫu được kết hợp với nhau nhằm tự động học và trích chọn các đặc trưng tiềm ẩn trong dữ liệu huấn luyện. Thông thường, trong các bài toán nhận dạng véc tơ đầu ra là một

véc tơ nhị phân phản ánh nhãn của dữ liệu huấn luyện. Đối với một mạng truyền tới nói chung và mạng tích chập nói riêng, véc tơ đầu ra của mô hình nhìn chung có thể là giá trị thực bất kỳ. Vì vậy, trong mô hình này chúng tôi ghép cặp dữ liệu để thực hiện học ánh xạ giữa các cặp tín hiệu tiếng nói và hình ảnh. Trong đó, tín hiệu tiếng nói được chuyển đổi biểu diễn dưới dạng phổ tần số và biểu diễn dưới dạng một ảnh đa cấp xám, còn dữ liệu ảnh được biểu diễn dưới dạng ảnh đa cấp xám với các giá trị từ 0 đến 255. Như vậy, mô hình sẽ nhận phổ tần số của tín hiệu tiếng nói làm đầu vào và một ảnh đa cấp xám làm đầu ra như hình 4.9.



Hình 4. 9 Mô hình học ánh xạ giữa tiếng nói và hình ảnh bằng mạng CNN

4.4. Thực nghiệm và kết quả

4.4.1. Thực nghiệm mô hình nhận thức tiếng nói dựa trên học quan hệ giữa tín hiệu âm thanh và tín hiệu hình ảnh

Trong thực nghiệm này, chúng tôi tiến hành trên hai bộ dữ liệu. Bộ dữ liệu thực nghiệm thứ nhất được xây dựng từ bộ dữ liệu DIGITS là bộ dữ liệu tiếng nói các chữ số từ 0 đến 9 bằng tiếng Anh và bộ dữ liệu ảnh MNIST là bộ dữ liệu chữ viết tay các số từ 0 đến 9. Từ hai bộ dữ liệu này chúng tôi chọn ngẫu nhiên 454 mẫu huấn luyện và chia thành hai tập, tập huấn luyện gồm 266 mẫu và tập kiểm tra là 188 mẫu.

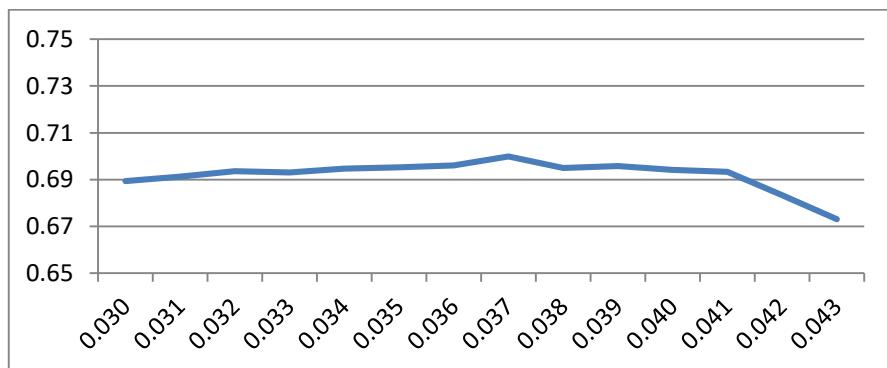
Bộ dữ liệu thứ hai được xây dựng từ bộ dữ liệu tiếng nói là tên gọi của 3 đối tượng (Pen, Ball và Mobile phone) và một bộ dữ liệu ảnh chụp ba đối tượng đó ở khoảng cách và góc chụp khác nhau. Bộ dữ liệu này gồm 50 mẫu huấn luyện và 20 mẫu kiểm tra cho mỗi lớp.

Cặp dữ liệu <tiếng nói, hình ảnh> làm dữ liệu huấn luyện được ghép cặp với nhau một cách ngẫu nhiên từ tập huấn luyện theo cách như sau. Chọn một mẫu tiếng nói ngẫu nhiên và một mẫu hình ảnh ngẫu nhiên của cùng một lớp

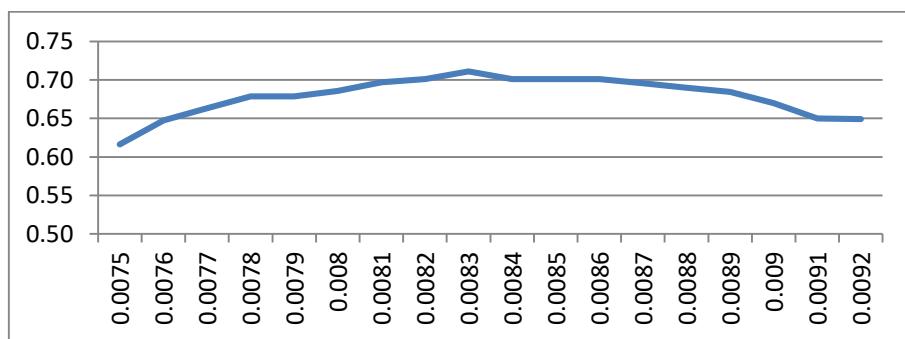
dữ liệu được gọi là cặp có quan hệ và đưa vào mô hình để huấn luyện. nếu chúng thuộc về 2 lớp khác nhau thì gọi là không có quan hệ. Ví dụ, chọn một mẫu dữ liệu tiếng nói trong lớp từ số 1 (one) đồng thời chọn ngẫu nhiên một mẫu hình ảnh của số 1 trong để ghép cặp thì cặp đó là có quan hệ còn ghép một mẫu tiếng nói của lớp từ một với mẫu hình ảnh của lớp số 2 thì cặp đó là không có quan hệ. Như vậy với tập dữ liệu DIGITS mỗi lần chạy sẽ tạo ra 266 cặp mẫu huấn luyện và 188 cặp kiểm tra, còn đối với bộ dữ liệu OBJECTS thì có 50 cặp mẫu huấn luyện và 20 cặp mẫu kiểm tra.

Để trích đặc trưng cho dữ liệu huấn luyện và dữ liệu kiểm tra, trong thực nghiệm này chúng tôi sử dụng đặc trưng bất biến SIFT. Đối với dữ liệu tiếng nói, để trích được đặc trưng bất biến SIFT, trước tiên dữ liệu tiếng nói được chuyển đổi thành phổ tần số của tín hiệu tiếng nói và trích đặc trưng SIFT từ phổ tần số này.

Thuật toán cũng được tiến hành thực nghiệm trên hai bộ dữ liệu DIGITS và OBJECT với các tham số ngưỡng khác nhau cho từng bộ dữ liệu. Hình 4.10 là kết quả phân lớp đối với bộ dữ liệu DIGITS, hình 4.11 là kết quả phân lớp trên bộ dữ liệu OBJECTS.



Hình 4. 10 Độ chính xác của mô hình trên bộ dữ liệu DIGITS

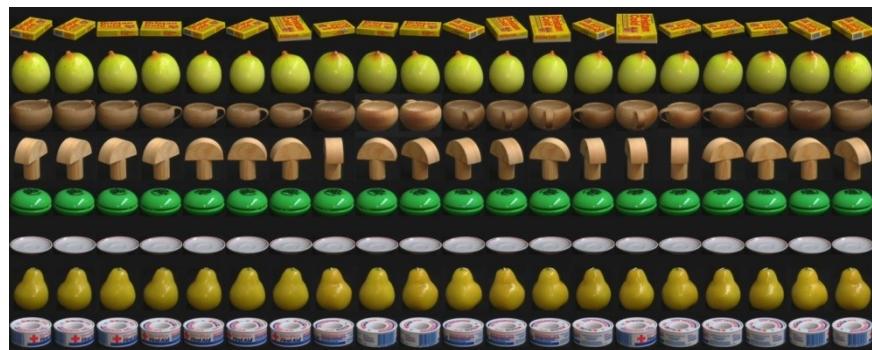


Hình 4. 11 Độ chính xác của mô hình trên bộ dữ liệu OBJECTS

4.4.2. Thực nghiệm mô hình nhận thức dựa trên mạng tích chập

Trong phần này, chúng tôi đề xuất mô hình nhận thức tiếng nói thông qua học mối quan hệ giữa tín hiệu tiếng nói với tín hiệu hình ảnh. Bài toán được mô hình như sau: Với tập huấn luyện gồm các cặp tín hiệu tiếng nói và hình ảnh có quan hệ như định nghĩa trên. Sau khi huấn luyện, với một tín hiệu tiếng nói mới được đưa vào, mô hình sẽ nhớ lại được hình ảnh liên kết với tín hiệu đã được huấn luyện.

Để tiến hành thực nghiệm mô hình này, chúng tôi xây dựng 3 bộ dữ liệu huấn luyện như sau: Bộ dữ liệu thứ nhất là ghép cặp giữa dữ liệu tiếng nói VN Places với 8 lớp hình ảnh của bộ dữ liệu Coil. Bộ dữ liệu thứ hai được tạo bằng cách ghép cặp cho một mẫu dữ liệu tiếng nói trong cơ sở dữ liệu tiếng nói DIGITS với một mẫu hình ảnh trong bộ dữ liệu MNIST. Bộ dữ liệu thứ ba được tạo bằng cách ghép cặp tương tự như bộ dữ liệu thứ nhất giữa bộ dữ liệu ISOLET với các lớp hình ảnh chữ cái A-Z của bộ dữ liệu FNT.



Hình 4. 12 Hai mươi mẫu huấn luyện của 8 lớp trong bộ dữ liệu COIL

A A A Q & A A A A A A A A A A A A A A
 B
 C
 D
 E
 F
 G
 H
 I
 J
 K
 L
 M
 N
 O
 P
 Q
 R
 S
 T
 U
 V
 W
 X
 Y
 Z

Hình 4. 13 Hai mươi mẫu huấn luyện của bộ dữ liệu FNT từ A đến Z

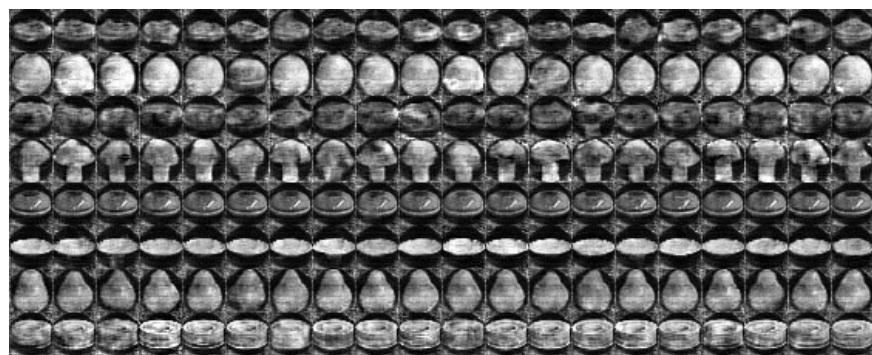
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9

Hình 4. 14 Hai mươi mẫu huấn luyện chữ số viết tay trong MNIST

Trong thí nghiệm này, chúng tôi cũng tiến hành chia dữ liệu thành hai bộ: tập huấn luyện và xác nhận lần lượt chiếm 80% và 20%. Mô hình đào huấn

luyện với 500 vòng. Chúng tôi tiến hành thay đổi kích thước dữ liệu đầu vào, số lớp và số nốt của từng lớp, các tham số của mô hình, rồi tiến hành so sánh các kết quả để tìm bộ tham số phù hợp nhất của mô hình với tập dữ liệu thực nghiệm. Kết quả tốt nhất khi mô hình là gồm các lớp như sau: Lớp C1: 32@26x26, lớp C2: 64@24x24, lớp M1: 64@12x12, lớp Dense1: 1024, lớp Dense 2: 784 tương ứng với kích thước ảnh đầu ra là 28x28, số học 1.0, rho = 0.95, epsilon là 1e-6, với kích thước đầu vào là 28x28.

Kết quả thực nghiệm, mô hình nhớ lại được các hình ảnh tương ứng với các tín hiệu tiếng nói đã được học được thể hiện ở các hình 4.15-4.17



Hình 4. 15 Hai mươi mẫu hình ảnh do mô hình sinh ra của bộ dữ liệu COIL

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9

Hình 4. 16 Hai mươi mẫu hình ảnh do mô hình sinh ra của bộ dữ liệu MNIST

Từ hình 4.15, 4.16, và 4.17 nếu đánh giá bằng mắt, có thể thấy hầu hết các hình ảnh do mô hình sinh ra đều phù hợp với dữ liệu huấn luyện. Để đánh giá kết quả một cách tự động, trong thực nghiệm này, chúng tôi sử dụng một mạng tích chập đã được trình bày tại phần 3.7. Trong mô hình đánh giá này, các kết quả đầu ra của mô hình nhận thức được biểu diễn dưới dạng ảnh có kích thước 28x28px làm dữ liệu đầu vào cho một mạng CNN để phân lớp. Bảng 4.1

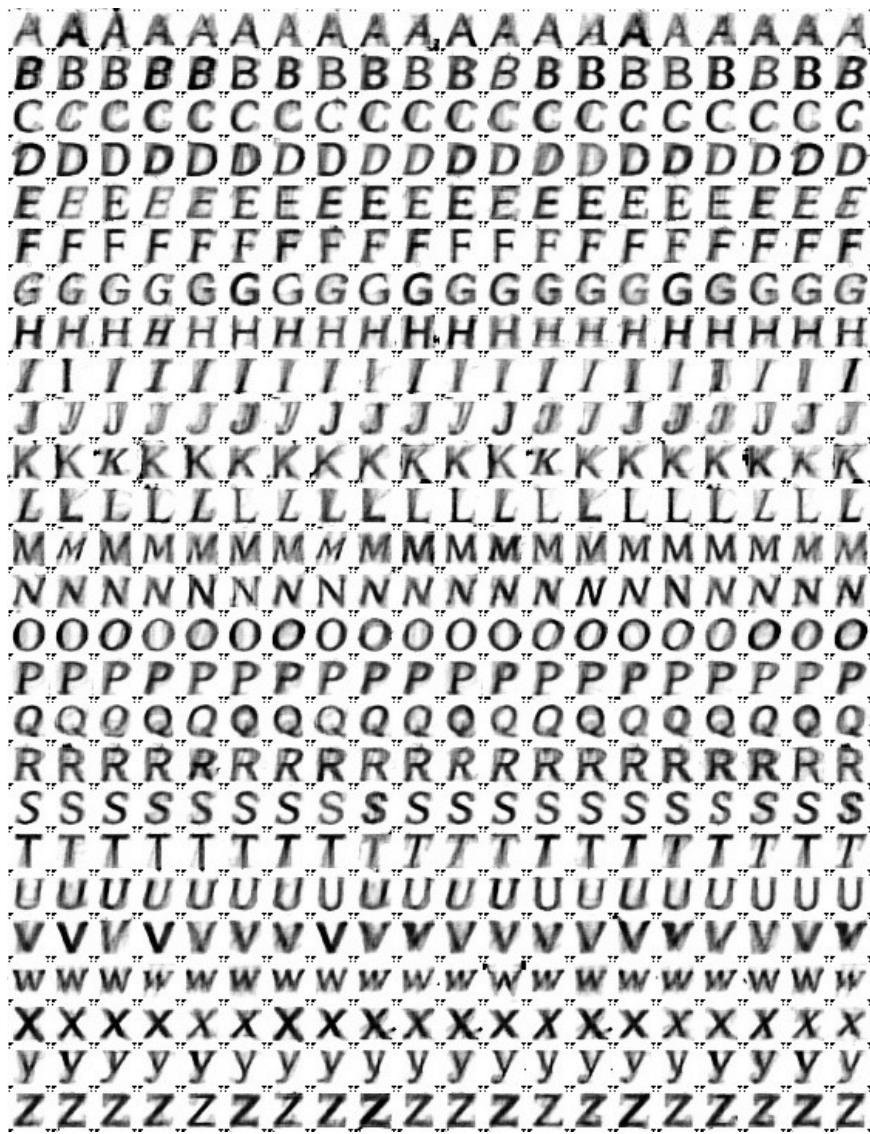
là kết quả phân lớp trung bình của 10 lần chạy đồng thời mô hình nhận thức và mô hình phân lớp.

Từ bảng 4.1 cho thấy, độ chính xác phân lớp ở bộ dữ liệu ISOLET-FNT cho kết quả thấp nhất, nguyên nhân chính là do số mẫu huấn luyện quá ít. Ở bộ dữ liệu ánh xạ giữa tập số từ 0 đến 9 phát âm trong tiếng anh với bộ dữ liệu MNIST có 414 mẫu cho kết quả cao nhất đạt 87% trong khi bộ dữ liệu ghép giữa tập tiếng gồm 8 địa danh phát âm trong tiếng Việt với bộ dữ liệu COIL là hình ảnh các vật thể có 72 mẫu thì kết quả lại chỉ đạt 41%.

Bảng 4.1 Kết quả phân lớp trung bình hình ảnh do mô hình nhận thức tiếng nói sinh ra bằng mạng tích chập

Kết quả	Số mẫu trong tập huấn luyện	Độ chính xác
DIGITS-MNIST	414	87%
ISOLET-FNT	26	30%
VN Places –COIL	72	41%

Đánh giá chung, kết quả phân lớp thực hiện bởi mô hình phân lớp CNN nhìn chung là thấp hơn so với hình ảnh thực tế nhìn bằng mắt thường. Nguyên nhân chính là do bộ dữ liệu huấn luyện còn ít, dẫn đến chất lượng hình ảnh ở đầu ra của mô hình nhận thức còn thấp, ảnh có đường nét, hình dáng của vật thể, nhưng mờ, nhòe nhiều, do đó khi sử dụng ảnh này làm dữ liệu đầu vào cho mô hình phân lớp kết quả còn thấp. Cũng từ thực nghiệm cho thấy, ảnh có độ phức tạp cao thì cần phải có số mẫu huấn luyện nhiều hơn so với ảnh đơn giản hơn. So sánh giữa bộ dữ liệu COIL và bộ DIGITS, ta thấy, bộ dữ liệu COIL cho kết quả thấp là do số mẫu huấn luyện thấp và độ phức tạp trong hình ảnh vật thể COIL phức tạp hơn so với bộ chữ viết tay các số từ 0 đến 9.



Hình 4. 17 Hai mươi mẫu hình ảnh kết quả do mô hình sinh ra đối với bộ dữ liệu FNT

Như vậy, có thể nói mô hình sẽ cho kết quả cao hơn nếu có nhiều dữ liệu huấn luyện, đồng thời, nếu dữ liệu hình ảnh và tiếng nói càng phức tạp thì càng đòi hỏi phải có nhiều mẫu huấn luyện hơn.

4.5. Kết luận

Chương này chúng tôi đề xuất một hướng tiếp cận cho bài toán nhận thức tiếng nói dựa trên mô hình học mối quan hệ giữa các đặc trưng của tiếng nói thu được qua cảm biến âm thanh (thính giác) với các đặc trưng thu được của hình ảnh thông qua bộ cảm biến thị giác bằng cách áp dụng phương pháp phân lớp LNBNN và mô hình học ánh xạ giữa tín hiệu tiếng nói với tín hiệu hình ảnh thông qua mạng tích chập. Mô hình thứ nhất dựa trên phương pháp phân lớp LNBNN kết hợp với đặc trưng SIFT trích chọn từ phổ tần số của tín hiệu tiếng

nói và tín hiệu hình ảnh. Mô hình này, cho phép nhận biết một cặp dữ liệu tiếng nói và hình ảnh mới đưa vào có quan hệ với nhau không, hay nói cách khác chúng có liên kết với nhau không. Mặc dù kết quả phân lớp chưa cao, nhưng mô hình cũng chứng tỏ có thể áp dụng trong lĩnh vực điều khiển người máy. Đối với mô hình thứ hai, kết quả hình ảnh mà mô hình nhận thức sinh ra nếu quan sát bằng mắt thường thì hầu hết là đúng với dữ liệu được huấn luyện. Tuy nhiên, do chất lượng hình ảnh sinh ra còn thấp, nguyên nhân do dữ liệu huấn luyện còn ít, vì vậy khi đem phân lớp bằng một mạng tích chập khác thì kết quả chưa cao. Như vậy, nếu với dữ liệu huấn luyện đủ lớn như trong bộ dữ liệu MNIST thì mô hình hoàn toàn đáp ứng được cho việc huấn luyện người máy nhận thức tiếng nói trực tiếp từ tín hiệu tiếng nói và hình ảnh do các cảm biến thu được.

Kết quả thực nghiệm cho thấy mô hình nhận thức này là một tiếp cận mới có thể cải tiến áp dụng cho việc huấn luyện người máy trong việc nhận thức tiếng nói một cách tự nhiên hơn, giống như quá trình nhận thức tiếng nói ở con người.

Kết quả nghiên cứu nêu trên được công bố tại kỳ yếu có phản biện của Hội nghị quốc tế lần thứ 8 về *Knowledge and Systems Engineering - KSE 2016* (công trình khoa học số 5) và kỳ yếu có phản biện của Hội nghị quốc tế The 5th NAFOSTED Conference on Information and Computer Science, NICS 2018 (công trình khoa học số 6).

Chương 5. MỘT SỐ CÁI TIẾN CHO BÀI TOÁN NHẬN THỨC TIẾNG NÓI DỮ LIỆU LỚN

5.1. Giới thiệu

Trong chương 3 và 4, chúng tôi đã đề xuất trích chọn đặc trưng SIFT từ phổ tần số của tiếng nói cho bài toán nhận thức tiếng nói. Kết quả thực nghiệm cho thấy đặc trưng SIFT là một đặc trưng phù hợp cho bài toán này. Mỗi điểm đặc trưng SIFT được mô tả bằng một véc tơ gồm 128 chiều mô tả đặc trưng của các hướng cho một vùng ảnh xung quanh điểm đặc trưng. Giá trị của mỗi chiều được lượng tử hóa trong đơn vị 1 byte dữ liệu do đó mỗi thành phần của điểm đặc trưng có giá trị từ 0 đến 255. Sau khi trích được các điểm đặc trưng bát biến của phổ tần số của tín hiệu tiếng nói, chúng tôi đề xuất sử dụng phương pháp phân lớp LNBNN để phân lớp các đặc trưng này. Phương pháp phân lớp LNBNN sẽ gộp tất cả các điểm đặc trưng thu được từ tất cả các mẫu để xây dựng một cơ sở dữ liệu đặc trưng bao gồm cả nhãn của các mẫu. Như vậy, đòi hỏi một không gian lớn để lưu trữ đồng thời với số lượng mẫu càng lớn thì cần phải có nhiều không gian và thời gian phân lớp sẽ trở thành một thách thức lớn.

Để giải quyết các bài toán dữ liệu lớn, có hai hướng tiếp cận chính. Một là rút gọn dữ liệu, làm cho kích thước của dữ liệu nhỏ hơn để có thể thực thi được mà không làm giảm nhiều đến độ chính xác của bài toán. Hai là hướng tiếp cận sử dụng các công nghệ tính toán cho bài toán dữ liệu lớn. Ngày nay có nhiều nền tảng công nghệ như Hadoop, Apache Spark cho phép các ứng dụng có thể làm việc với hàng ngàn máy tính khác nhau và hàng petabyte dữ liệu. Hadoop được phát triển dựa trên ý tưởng từ các công bố của Google về mô hình MapReduce và hệ thống dữ liệu phân tán, còn Apache Spark được phát triển vào năm 2009 bởi AMPLab tại đại học California, sau đó được tổ chức phần mềm Apache phát triển cho đến nay. Apache Spark dựa trên Hadoop MapReduce và nó mở rộng mô hình MapReduce để sử dụng hiệu quả nó cho nhiều loại tính toán hơn, bao gồm các truy vấn tương tác và xử lý luồng. Spark tính toán nhanh hơn Hadoop nhờ vào việc tính toán được thực hiện ở bộ nhớ trong. Bên cạnh đó, với sự phát triển mạnh mẽ của phần cứng máy tính, GPU đã trở thành một thành phần không thể thiếu trong các hệ thống tính toán dữ liệu lớn, trong các mạng học sâu. Sự kết hợp giữa Hadoop, Spark với GPU đã

cung cấp một cách tiếp cận khá độc đáo để đưa tất cả các công nghệ đó vào một cụm duy nhất.

Như vậy Hadoop chính là một nền tảng công nghệ cho việc xử lý các bài toán dữ liệu lớn. Tuy nhiên, đến nay, Hadoop mới chỉ hỗ trợ một số phương pháp học máy như naive Bayes, Decision trees, random forests, gradient-boosted trees, K-means, Gaussian mixtures (GMMs),... chưa có cài đặt cho thuật toán LNBNN.

Trong phần này, chúng tôi đề xuất hai cải tiến cho bài toán nhận thức tiếng nói với dữ liệu lớn đáp ứng yêu cầu gia tăng dữ liệu trong thực tế của mô hình. Một là, chúng tôi đề xuất phương pháp rút gọn dữ liệu để giảm bớt không gian lưu trữ các đặc trưng nhằm cải tiến tốc độ phân lớp. Hai là, đề xuất cài đặt phương pháp phân lớp LNBNN trên nền tảng xử lý song song, phân tán dựa trên Hadoop cho bài toán nhận thức tiếng nói dữ liệu lớn.

5.2. Rút gọn đặc trưng

5.2.1. Giới thiệu về rút gọn đặc trưng

Rút gọn đặc trưng là quá trình rút gọn hoặc biến đổi không gian dữ liệu ban đầu thành một không gian mới có số chiều nhỏ hơn không gian ban đầu. Rút gọn đặc trưng là một trong các phương pháp thường được sử dụng để cải tiến độ chính xác phân lớp của các kỹ thuật học máy, đồng thời tiết kiệm thời gian tính toán và bộ nhớ lưu trữ. Các hướng tiếp cận cho bài toán rút gọn dữ liệu có thể chia thành hai hướng chính là: Trích xuất đặc trưng và lựa chọn đặc trưng.

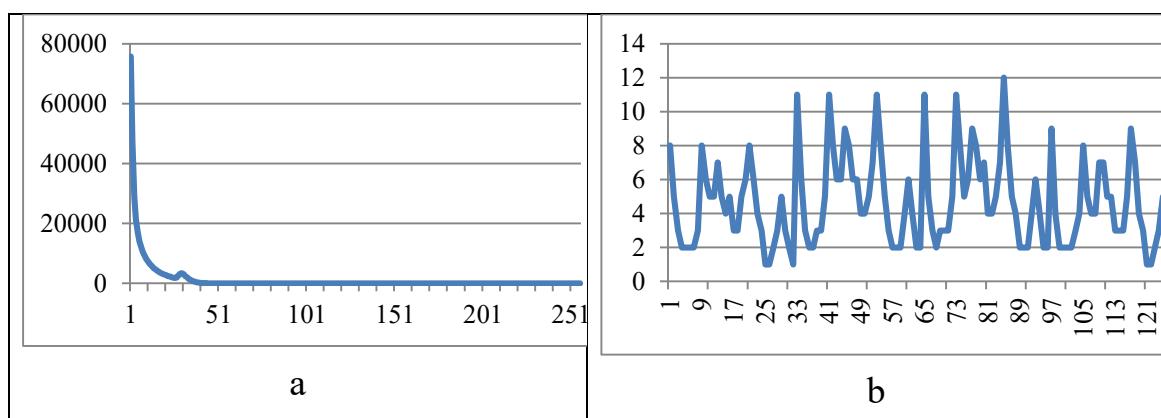
Lựa chọn đặc trưng là quá trình lựa chọn một tập con từ tập các đặc trưng ban đầu mà không hề có sự biến đổi nào đối với các đặc trưng. Đối với bài toán lựa chọn đặc trưng, các nhà nghiên cứu hiện nay chủ yếu tập trung vào phát triển các kỹ thuật lựa chọn đặc trưng theo hai hướng chính là chiến lược tìm kiếm [Angelis, 2006], [Gheyas, 2010] và tiêu chí đánh giá [Sun, 2007].

Trích xuất đặc trưng là quá trình biến đổi các đặc trưng ban đầu sang một không gian khác có chiều thấp hơn. Hay nói cách khác là nó xây dựng một tập đặc trưng mới từ tập đặc trưng ban đầu. Trong giảm chiều dữ liệu thì trích xuất đặc trưng liên quan tới việc tạo ra tập đặc trưng “mới” từ tập đặc trưng ban đầu, thông qua việc áp dụng một số ánh xạ. Trích xuất đặc trưng thực hiện một số

biến đổi của đặc trưng ban đầu để tạo ra các đặc trưng khác có ý nghĩa hơn. Trích xuất đặc trưng gồm các phương pháp như phương pháp dựa trên lý thuyết phân tích thống kê như phương pháp phân tích thành phần chính (PCA), phương pháp phân tích đa thành phần chính (Multiple Principal Component Analysis - MPCA) được xây dựng dựa trên PCA [Zhang, 2009] ; Phương pháp phân tích thành phần độc lập (ICA) [Soliz, 2008] [Yuen, 2002] ; Phương pháp phân tích biệt thức tuyến tính (LDA) [Balakrishnama, 1999] là một đại diện tiêu biểu của các phương pháp trích xuất đặc trưng tuyến tính [Yang, 2009] [Park, 2008] , phân tích biệt thức tuyến tính cân bằng [Wang, 2006] ; Phương pháp trích xuất đặc trưng dựa trên phân tích ma trận giá trị riêng [Wiener, 1995] ; Phương pháp phân tích biệt thức không tương quan thống [Jin, 2001] ; Phương pháp phân tích thành phần chính dựa trên hàm nhân (KPCA) [Yang, 2005] có ý tưởng chính là ánh xạ từ dữ liệu đầu vào sang một không gian đặc trưng thông qua một ánh xạ phi tuyến.

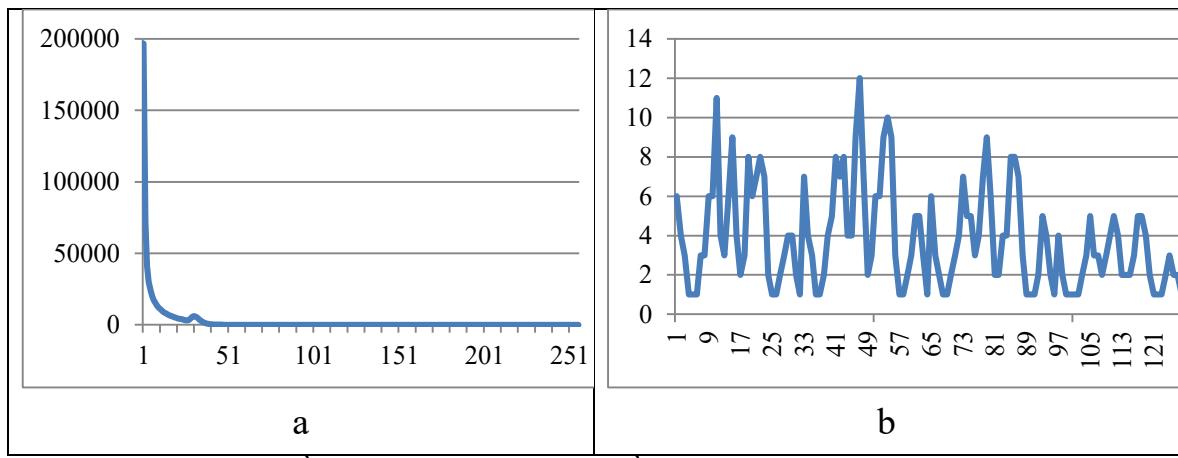
5.2.2. Rút gọn đặc trưng SIFT

Mỗi véc tơ đặc trưng SIFT gồm 128 chiều trong đó các thành phần của véc tơ đặc trưng được lượng tử hóa có giá trị là một số nguyên từ 0 đến 255. Như vậy không gian của đặc trưng SIFT có tổng số $256^{128} \approx 1,8 \times 10^{308}$ điểm rời rạc khác nhau tương đương với $1,8 \times 10^{308}$ véc tơ đặc trưng. Số lượng các hạt trong vũ trụ được ước lượng vào khoảng 10^{80} trong khi số lượng các điểm đặc trưng khác nhau trong không gian đặc trưng SIFT là xấp xỉ $1,8 \times 10^{308}$ như vậy về mặt lý thuyết không gian đặc trưng này có khả năng phân biệt mọi điểm trong thế giới [Kadir, 2011] .

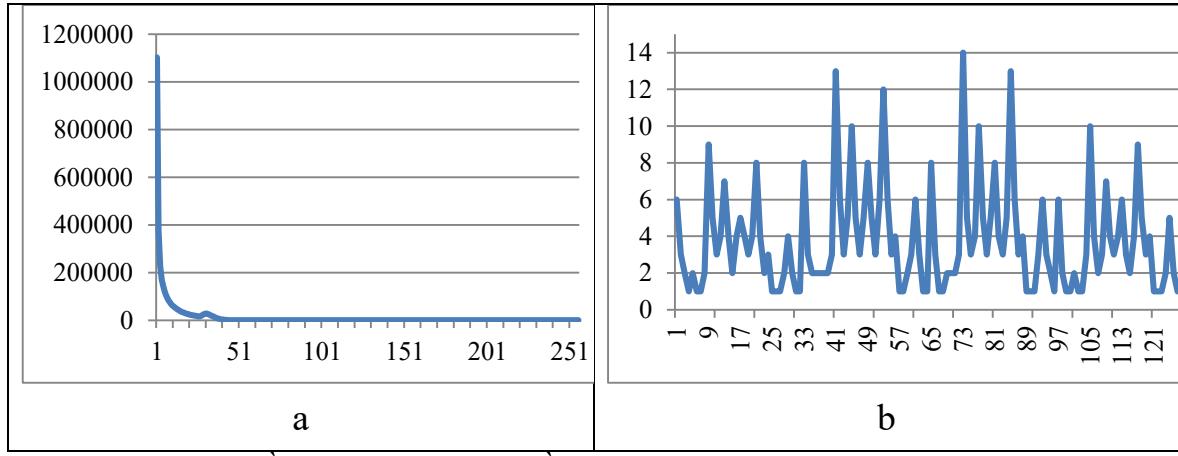


Hình 5. 1 a. Lược đồ giá trị các thành phần của điểm đặc trưng SIFT, b. Medians của các thành phần của SIFT trên dữ liệu ISOLET

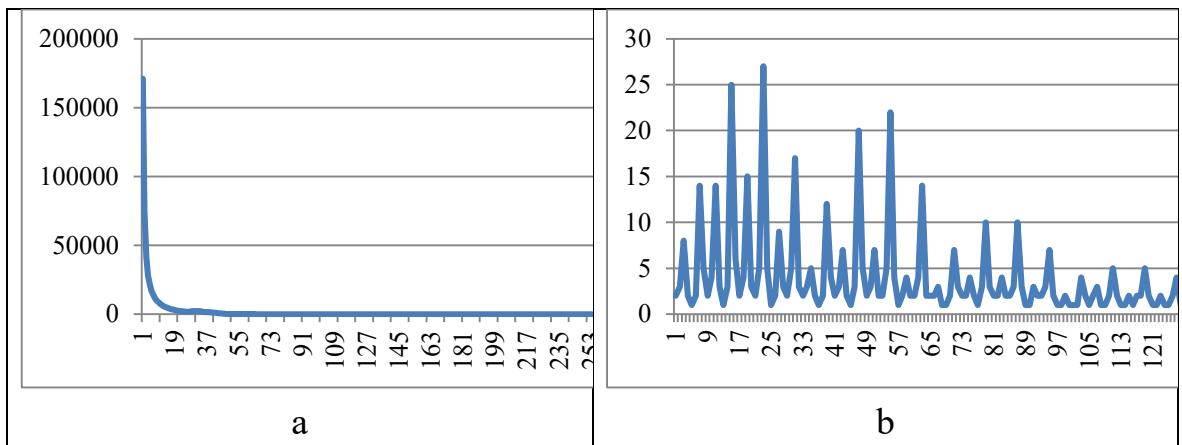
Về mặt thực nghiệm, để đánh giá phân bố giá trị của các thành phần của đặc trưng SIFT, chúng tôi tiến hành thống kê giá trị của chúng trên một số bộ dữ liệu thực nghiệm. Các hình 5.1a- 5.5a cho thấy tần suất giá trị của các thành phần của đặc trưng SIFT trong năm bộ dữ liệu trong khi các biểu đồ 5.1b - 5.5b cho biết giá trị trung vị của các thành phần. Từ thực nghiệm cho thấy hầu hết giá trị của các thành phần của đặc trưng SIFT có giá trị nhỏ hơn 50 (xem hình 5.1a- 5.5a). Như vậy, đặc trưng SIFT chỉ chiếm một phần rất nhỏ trong miền giá trị của không gian đặc trưng SIFT kể cả trong không gian đặc trưng SIFT nhị phân.



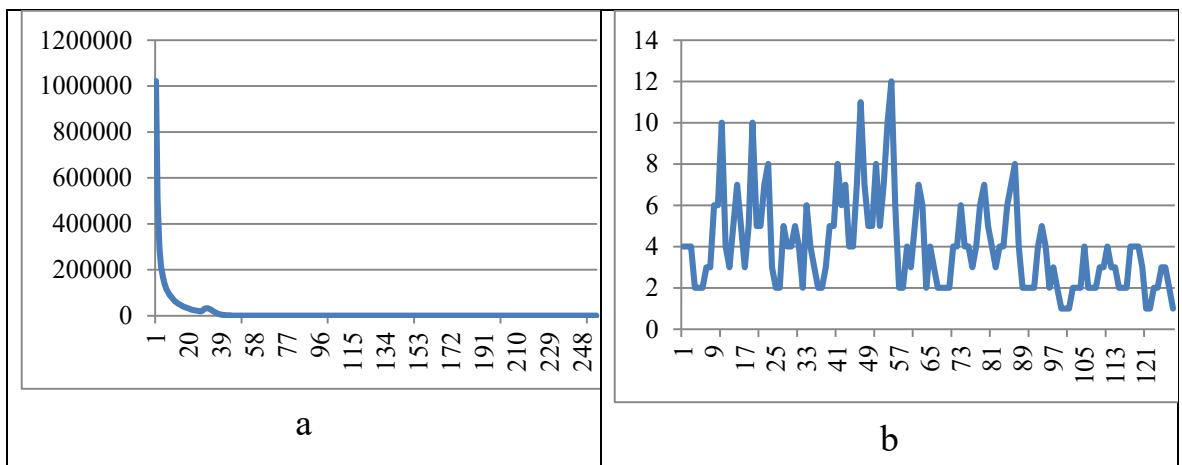
Hình 5. 2 a. Lược đồ giá trị của các thành phần của SIFT trên cơ sở dữ liệu Digits, b. Medians của các thành phần của SIFT trên dữ liệu Digits



Hình 5. 3 Lược đồ giá trị các thành phần của đặc trưng SIFT trên dữ liệu PLACES, b. Median của SIFT trên dữ liệu PLACES



Hình 5. 4 a. Lược đồ giá trị các thành phần của SIFT trên dữ liệu JVPD, b. Trung vị của các thành phần của SIFT trên dữ liệu JVPD



Hình 5. 5 Lược đồ giá trị các thành phần của SIFT trên dữ liệu TMW, b. Medians của các thành phần của SIFT trên dữ liệu TMW

Từ những phân tích trên chúng tôi đề xuất một phương pháp rút gọn dữ liệu bằng cách lượng tử hóa các thành phần của đặc trưng SIFT để mỗi mỗi thành phần của đặc trưng SIFT chỉ cần một bít dữ liệu hay nói cách khác là lượng tử hóa các thành phần của đặc trưng SIFT từ một byte về một bít. Đầu tiên các đặc trưng SIFT sẽ được trích từ phổ tần số của tiếng nói, sau đó tính giá trị trung vị (median) cho từng thành phần của tất cả các đặc trưng sau đó lượng tử hóa giá trị của chúng theo các giá trị trung vị này theo công thức

$$f'_i = \begin{cases} 0 & \text{nếu } f_i < \text{median}_i \\ 1 & \text{nếu ngược lại} \end{cases}$$

Như vậy, sau khi lượng tử hóa với các giá trị median của các thành phần, mỗi điểm đặc trưng SIFT mà các thành phần của nó có giá trị thuộc đoạn [0,

255] sẽ trở thành một véc tơ mà các thành phần của nó chỉ nhận giá trị 0 hoặc 1. Như vậy, mỗi điểm đặc trưng chỉ cần 128 bit để lưu trữ. Tiếp theo, chúng tôi đề xuất mã hóa lại các đặc trưng này bằng cách nhóm 8 đặc trưng liên tiếp thành một byte như vậy mỗi điểm đặc trưng SIFT nhị phân sẽ được mã hóa thành một điểm đặc trưng 16 chiều trong đó giá trị của mỗi thành phần sẽ nhận từ 0 đến 255. Bằng cách mã hóa này thì 2 điểm đặc trưng được coi là gần nhau theo khoảng cách Euclide trở thành 2 điểm gần nhau theo khoảng cách Hamming. Thuật toán được mô tả như sau:

Thuật toán 5. 1 Thuật toán rút gọn đặc trưng SIFT_REDUC

Thuật toán: SIFT_REDUC

Đầu vào: tập đặc trưng SIFT từ dữ liệu huấn luyện

Đầu ra: Tập đặc trưng SIFT thu gọn

1. Tính giá trị trung vị cho các thành phần của đặc trưng SIFT
2. Với mỗi đặc trưng SIFT trong dữ liệu
3. Với mỗi thành phần của điểm đặc trưng SIFT
4. Nếu giá trị của thành phần lớn hơn giá trị median_i thì giá trị được gán thành 1.
5. Ngược lại giá trị được gán là 0
6. Mã hóa 128 bit thành 16 byte.

Các điểm đặc trưng SIFT thu gọn này sẽ được đưa vào huấn luyện bằng phương pháp LNBNN. Để giảm bớt thời gian tìm kiếm K-hàng xóm gần nhất, LNBNN sử dụng thư viện FLANN hỗ trợ tạo cấu trúc dữ liệu KD-TREE, một cấu trúc dữ liệu cho phép tìm kiếm K hàng xóm gần nhất của một véc tơ có số chiều lớn trong thời gian ngắn. Chúng tôi so sánh thời gian chạy của thuật toán với dữ liệu SIFT nhị phân bằng các phương pháp tìm kiếm khác nhau như tìm kiếm tuyến tính, tìm kiếm với multi-index hashing với thời gian chạy của SIFT ban đầu.

5.2.3. Bảng băm đa chỉ số

Bảng băm đa chỉ số (Multi index hashing - MIH) được Norouzi đề xuất cho bài toán tìm kiếm K hàng xóm gần nhất với dữ liệu nhị phân với độ đo khoảng cách Hamming [Norouzi, 2012]. Trong phương pháp tìm kiếm này,

các mă nhị phân từ cơ sở dữ liệu được chia thành m phần rời nhau, sau đó xây dựng bảng chỉ mục ứng với mỗi phần. Ứng với mỗi truy vấn, K hàng xóm gần nhất của mỗi đoạn con của truy vấn sẽ được coi là ứng viên của K hàng xóm gần nhất của toàn bộ truy vấn. Các hàng xóm tiềm năng này sẽ được kiểm tra bằng cách sử dụng toàn bộ mă truy vấn nhị phân ban đầu để loại bỏ các hàng xóm không thuộc K-hàng xóm gần nhất. Độ lớn của các chuỗi con phải được chọn sao cho tập ứng viên nhỏ và đảm bảo hợp lý trong việc lưu trữ. Thuật toán xây dựng bảng băm đa chỉ số được trình bày trong thuật toán 5.2, thuật toán tìm kiếm K hàng xóm gần nhất được trình bày trong thuật toán 5.3.

Thuật toán 5. 2 Thuật toán xây dựng bảng băm đa chỉ số MIH

Thuật toán: MIH

Đầu vào:

- n đặc trưng được biểu diễn bằng mă nhị phân $H = \{h_i\}$ với $i = 1, \dots, n$
- tham số m

Đầu ra: m bảng băm con

1. For $j=1$ to m do
2. Khởi tạo bảng băm thứ j
3. For $i = 1$ to n do
4. Chèn h_i vào bảng băm thứ j
5. End for
6. End for

Thuật toán 5. 3 Thuật toán tìm kiếm K hàng xóm gần nhất MIH_KNN

Đầu vào: MIH_KNN

- Truy vấn $g = \{g_j\}$ với $j = 1, m$
- Tham số $k' =$ phần nguyên của k/m , và $a = k - mk'$

Đầu ra: tập k-hàng xóm gần nhất của truy vấn g

1. For $j=1$ to $a+1$ do
2. Tìm k' hàng xóm gần nhất của g_j từ bảng băm thứ j
3. End for
4. For $j=a+2$ to m do
5. Tìm $(k'-1)$ hàng xóm gần nhất của g_j từ bảng băm thứ j
6. End for
7. Loại bỏ các kết quả không phải là k-hàng xóm gần nhất

5.2.4. Thực nghiệm và kết quả

Trong các thực nghiệm này, chúng tôi sử dụng 05 bộ dữ liệu tiếng nói đó là ISOLET, DIGITS, VNPLACES, TMW, và JVPD.

Bảng 5.1 và bảng 5.2 dưới đây so sánh độ chính xác phân lớp và thời gian chạy của thuật toán LNBNN với các phương pháp tìm kiếm K-hàng xóm gần nhất khác nhau. Trong Bảng 5.1, và Bảng 5.1 cột 1 là độ chính xác và thời gian chạy đối với đặc trưng SIFT gốc, cột 2, 3, 4 là kết quả phân lớp với đặc trưng SIFT nhị phân trong đó cột 2 là sử dụng phương pháp tìm kiếm tuyến tính, cột 3 là tìm kiếm với phương pháp phân cụm phân cấp, và cột 4 là sử dụng phương pháp đa chỉ số MIH.

Bảng 5. 1 So sánh độ chính xác phân lớp trên các bộ dữ liệu

Database	<i>Origin SIFT KD-TREE</i>	<i>Binary SIFT Linear Brute Force</i>	<i>Binary SIFT Hierarchical Clustering</i>	<i>Binary SIFT MIH</i>
ISOLET	56.3	56.3	56.3	56.3
EN DIGITS	95.4	95.8	95.3	96.2
VN PLACES	91.2	90.5	89.8	90.8
JVPD	95.1	94.6	93.7	95.0
TMW	83.1	89.9	89.9	89.9

Bảng 5.1 cho thấy trong các bộ dữ liệu thực nghiệm độ chính xác phân lớp trước và sau khi lượng tử hóa SIFT hầu như không thay đổi trong khi kích thước của dữ liệu giảm được 8 lần từ 128 byte xuống còn 16 byte cho mỗi điểm đặc trưng.

Bảng 5. 2 So sánh thời gian chạy trên các dữ liệu khác nhau (giây)

Databases	<i>Num descriptor</i>	<i>Origin SIFT KD-TREE</i>	<i>Binary SIFT Linear Brute Force</i>	<i>Binary SIFT Hierarchical Clustering</i>	<i>Binary SIFT MIH</i>
ISOLET	327,396	657	654	124	473
EN. DIGITS	581,134	1,584	3,848	643	2,331
VN PLACES	856,121	725	13,359	307	1,919
JVPD	489,998	11,144	1,613	228	901
TMW	3,605,234	25,364	73,595	1,892	43,295

Bảng 5.2 cho thấy trong các bộ dữ liệu thực nghiệm cho thấy đối với Binary SIFT, thời gian thực hiện giảm nhiều nhất đối với phương pháp tìm kiếm phân cấp, tiếp đến là phương pháp tìm kiếm đa chỉ số MIH.

5.3. Cài đặt phương pháp phân lớp LNBNN cho bài toán nhận thức tiếng nói dữ liệu lớn

5.3.1. Giới thiệu Framework Hadoop

Apache Hadoop là một framework dùng để chạy những ứng dụng trên một cụm máy tính được xây dựng bằng cách kết nối nhiều máy tính có cấu hình phần cứng thông thường. Hadoop có hai thành phần chính là hệ thống quản lý dữ liệu phân tán HDFS (Hadoop Distributed File System) và hệ thống xử lý song song MapReduce.

HDFS là một hệ thống quản lý lưu trữ chính trong Hadoop. HDFS cho phép truy cập dữ liệu trên các cụm Hadoop một cách hiệu quả. HDFS thường được triển khai trên các phần cứng chi phí thấp, rất dễ xảy ra lỗi phần cứng. Vì vậy, HDFS được xây dựng để có khả năng chịu lỗi cao với tốc độ truyền dữ liệu giữa các nút trong HDFS là rất cao, dẫn đến giảm thiểu nguy cơ lỗi.

MapReduce là quy trình giúp xử lý tập hợp dữ liệu siêu lớn đặt tại các máy tính phân tán, có thể xử lý được cả dữ liệu không cấu trúc và dữ liệu cấu trúc. Trong MapReduce, các máy tính chứa dữ liệu đơn lẻ được gọi là các nút (node).

Ngoài hai thành phần chính là HDFS và MapReduce, Hadoop còn có một số thành phần hỗ trợ khác, đó là:

-**Hadoop Streaming** là một tiện ích để tạo nên mã MapReduce bằng bất kỳ ngôn ngữ nào như C, Perl, Python, C++, Bash,...;

-**Hive và Hue** là tiện ích cho phép chuyển đổi câu lệnh SQL thành một tác vụ MapReduce;

-**Pig** là một môi trường lập trình mức cao hơn để viết mã MapReduce. Ngôn ngữ Pig được gọi là Pig Latin. Bạn có thể thấy các quy ước đặt tên hơi khác thường một chút, nhưng bạn sẽ có tỷ số giá-hiệu năng đáng kinh ngạc và tính sẵn sàng cao;

-**Sqoop** cung cấp việc truyền dữ liệu hai chiều giữa Hadoop và cơ sở dữ liệu quan hệ yêu thích của bạn;

-**Oozie** quản lý luồng công việc Hadoop;

-**HBase** là kho lưu trữ key-value có thể mở rộng quy mô rất lớn. Nó hoạt động rất giống như một hash-map để lưu trữ lâu dài;

-**FlumeNG** là trình nạp thời gian thực để tạo luồng dữ liệu của bạn vào Hadoop. Nó lưu trữ dữ liệu trong HDFS và HBase. Bạn sẽ muốn bắt đầu với FlumeNG, để cải thiện luồng ban đầu;

-**Whirr** cung cấp đám mây cho Hadoop.

-**Mahout**: Máy học dành cho Hadoop. Được sử dụng cho các phân tích dự báo và phân tích nâng cao khác.

-**Fuse**: Làm cho hệ thống HDFS trông như một hệ thống tệp thông thường.

- **Zookeeper**: Được sử dụng để quản lý đồng bộ cho hệ thống.

5.3.2. Cài đặt thuật toán phân lớp LNBNN trên nền Hadoop

Để cài đặt được phương pháp phân lớp LNBNN trên nền Hadoop, chúng ta cần phải hiểu được cơ chế điều khiển dữ liệu vào ra của Hadoop. Hadoop sử dụng framework MapReduce để thực hiện các thao tác xử lý với dữ liệu.

MapReduce được chia thành hàm là Map và Reduce. Những hàm này được định nghĩa bởi người dùng và là hai giai đoạn liên tiếp trong quá trình xử lý dữ liệu.

+ Map nhận đầu vào là tập các cặp <khóa, giá trị> và đầu ra là tập các cặp <khóa, giá trị trung gian> ghi xuống đĩa cứng và thông báo cho Reduce nhận dữ liệu để xử lý.

+ Reduce sẽ nhận khóa trung gian I và tập các giá trị ứng với khóa đó, ghép nối chúng lại để tạo thành một tập khóa nhỏ hơn. Các cặp khóa/giá trị trung gian sẽ được đưa vào cho hàm Reduce thông qua một con trỏ vị trí (iterator). Điều này cho phép ta có thể quản lý một lượng lớn danh sách các giá trị để phù hợp với bộ nhớ.

Thực chất giữa bước Map và Reduce còn có một bước phụ mà bước này thực hiện song song với bước reduce đó là sắp xếp (Shuffle). Tức là sau khi map thực hiện xong toàn bộ công việc của mình, kết quả của Map được đặt rải rác trên các cụm khác nhau nên Shuffle sẽ làm nhiệm vụ thu thập các cặp

<khóa-giá trị trung gian> do Map sinh ra mà có cùng khóa, sắp xếp lại và chuyển cho Reduce thực hiện.

MapReduce thực hiện các thủ tục Map và Reduce song song và độc lập nhau. Tất cả thủ tục Map có thể chạy song song và khi mỗi nốt hoàn thành tác vụ thì chúng gửi trả về nốt chủ. Thủ tục này có thể rất hiệu quả khi nó được thực hiện trên một số lượng rất lớn dữ liệu.

MapReduce có 5 bước khác nhau:

- Chuẩn bị dữ liệu đầu vào cho Map
- Thực thi mã Map được cung cấp bởi người dùng
- Trộn dữ liệu xuất của Map vào bộ xử lý Reduce
- Thực thi mã Reduce được cung cấp bởi người dùng
- Tạo dữ liệu xuất cuối cùng

MapReduce xử lý dữ liệu dưới dạng một cặp giá trị `<key, value>`, vì vậy để cài đặt phương pháp phân lớp LNBNN trên nền Hadoop, cần phải biểu diễn dữ liệu dưới dạng mà MapReduce có thể nhận và xử lý được. Dữ liệu huấn luyện và dữ liệu kiểm tra được lưu trữ trong tệp dữ liệu có cấu trúc như sau:

`key: key1; value: 3; len: 2; 0:3; 1:4;`

trong đó:

- **Key:** `key1` là khóa xác định của điểm đặc trưng của dữ liệu
- **Value:** `3` là nhãn của điểm đặc trưng, `3` ở đây chỉ ra rằng nhãn của điểm đặc trưng này là thuộc lớp `3`
- **Len:** `2` số chiều của véc tơ điểm đặc trưng, trong ví dụ này điểm đặc trưng có `2` chiều
 - các cặp số sau khóa len là chỉ số và giá trị của các thành phần trong véc tơ điểm đặc trưng của dữ liệu.

Bước đầu tiên trong quy trình xử lý của Map là chuẩn bị dữ liệu, ở đây ta cung cấp các thông tin cần thiết làm tham số đầu vào cho thủ tục Map sử dụng trong quá trình xử lý.

Thuật toán 5. 4 Thuật toán LNBNN-HADOOP-SETUP

Thuật toán: LNBNN-HADOOP-SETUP

Input: a testing file name

Output: testList is a set of feature point of query.

1. **For** each line in testing file **do**
2. Convert string line to a query vector
3. Add vector to **testList**
4. **End for**

Thủ tục này được đặt tên là Setup. Thuật toán 5.4 mô tả các bước chính của thủ tục Setup, thủ tục chuẩn bị dữ liệu đầu vào cho thủ tục Map. Trong bước chuẩn bị dữ liệu này, đầu vào là tên tệp chứa tập hợp các điểm đặc trưng của dữ liệu truy vấn (test). Đầu ra của thủ tục là một biến chứa các điểm đặc trưng này trong bộ nhớ. Thủ tục này có nhiệm vụ đọc từng dòng dữ liệu của tệp truy vấn và chuyển đổi thành một danh sách các vec tơ điểm đặc trưng tương và được lưu trong một biến toàn cục (testList).

Thuật toán 5. 5 Thuật toán LNBNN-HADOOP-MAP

Thuật toán: LNBNN-HADOOP-MAP

Đầu vào:

Value là dòng dữ liệu trong tập huấn luyện bao gồm cả dữ liệu và nhãn

Đầu ra:

A list of **<KeyOut, ValueOut> pair**.

1. Convert Value (current line in training) to a vector **curVec**
2. **For each test_vector in testList do**
3. Calculate distance from curVec to test_vector
4. Create KeyOut =<feature_id, distance > is a pair of feature point id in query (**test_vector**) and its distance to the current feature point in training set (**curVec**)
5. Create ValueOut =<label, distance> is a pair of class label and its distance from a feature point id in query (**test_vector**) to the current feature point in training set (**curVec**)
6. Context.write(KeyOut,ValueOut)
7. **End for**

Thuật toán 5.5 mô tả các bước chính trong thủ tục Map, đầu vào của thủ tục này là các dòng dữ liệu được đưa vào dựa vào các thông tin được cung cấp từ thủ tục Setup. Đầu ra của thủ tục này là các cặp gồm Key và Value.

Trong pha Map, dữ liệu huấn luyện được phân chia thành các phần và được xử lý song song bởi các tiến trình. Mặc định MapReduce nhận dữ liệu từ hệ thống HDFS. Thủ tục Map đọc từng hàng dữ liệu trong tập huấn luyện. Mỗi hàng trong tập huấn luyện được chuyển đổi thành một véc tơ điểm đặc trưng. Ứng với mỗi véc tơ điểm đặc trưng trong truy vấn (testList), thủ tục Map tính khoảng cách từ điểm đặc trưng hiện hành với từng điểm đặc trưng trong truy vấn sau đó xuất chúng ra cùng với nhãn của điểm đặc trưng tương ứng. Kết quả của thủ tục map sẽ sinh ra hàng loạt các cặp $\langle \text{key}, \text{value} \rangle$. Các cặp dữ liệu này sẽ được chuyển tới thủ tục Reduce để xử lý.

Thuật toán 5. 6 thuật toán LNBNN-HADOOP-REDUCE

Thuật toán: LNBNN-HADOOP-REDUCE

Đầu vào:

- **K** là số hàng xóm gần nhất cần tìm
- **Key** là một cặp gồm chỉ số của điểm đặc trưng và khoảng cách (Feature point Id of query, distance),
- **Value** là tập các cặp (class label, distance)

Đầu ra:

Totals: tổng khoảng cách từ truy vấn tới tất cả các lớp

1. Count =0;
2. **For** each RecordKey in Value **do**
3. **If** Count = K **then**
4. BG_distance = recordKey.getDistance()
5. break;
6. **Else**
7. Count = Count +1;
8. **End if**
9. **If** recordKey not in NeighborList **then**
10. Add recordKey to NeighborList
11. **End if**
12. **End for**
13. **For** each neighbor in NeighborList **do**
14. **Totals**[neighbor] += neighbor.Distance() – BG_distance;
15. **End For**

Thủ tục Reduce nhận dữ liệu đầu vào là các cặp dữ liệu được sinh ra từ các thủ tục map. Thủ tục Reduce có nhiệm vụ xử lý, hợp nhất các cặp dữ liệu

này để cho kết quả cuối cùng. Thủ tục Reduce chọn **K** phần tử có khoảng cách ngắn nhất tương ứng với từng điểm đặc trưng của dữ liệu truy vấn để tính tổng khoảng cách ngắn nhất từ truy vấn đến tất cả các lớp tìm được trong K hàng xóm gần nhất của mỗi điểm đặc trưng trong truy vấn. Khoảng cách biên được tính bằng khoảng cách từ hàng xóm gần nhất thứ **K+1** của mỗi điểm đặc trưng trong truy vấn. Cuối cùng, hiệu giữa khoảng cách nhỏ nhất từ một điểm đặc trưng của truy vấn đến tất cả các điểm đặc trưng trong tập huấn luyện với khoảng cách biên sẽ được cộng dồn vào tổng khoảng cách tới lớp tương ứng. Reduce lưu các kết quả tính toán của nó trong biến toàn cục (Totals – tổng khoảng cách nhỏ nhất từ truy vấn đến các lớp trong tập huấn luyện). Để tìm kết quả phân lớp, thuật toán cần tìm lớp có tổng khoảng cách nhỏ nhất từ biến toàn cục Totals.

Thuật toán 5. 7 Thuật toán LNBNN-HADOOP-CLEANUP

Thuật toán: LNBNN-HADOOP-CLEANUP

Đầu vào:

Totals is total distance from all feature points in query to all classes found in KNN search in training database.

Đầu ra:

BestClass is the class with minimum distance

1. Min_dist = 999999
2. BestClass = null;
3. **For** each Entry in **Totals** **do**
4. **If** min_dist > Entry.dist **then**
5. BestClass = Entry.getKey();
6. Min_dist = Entry.getValue();
7. **Endif**
8. **End for**

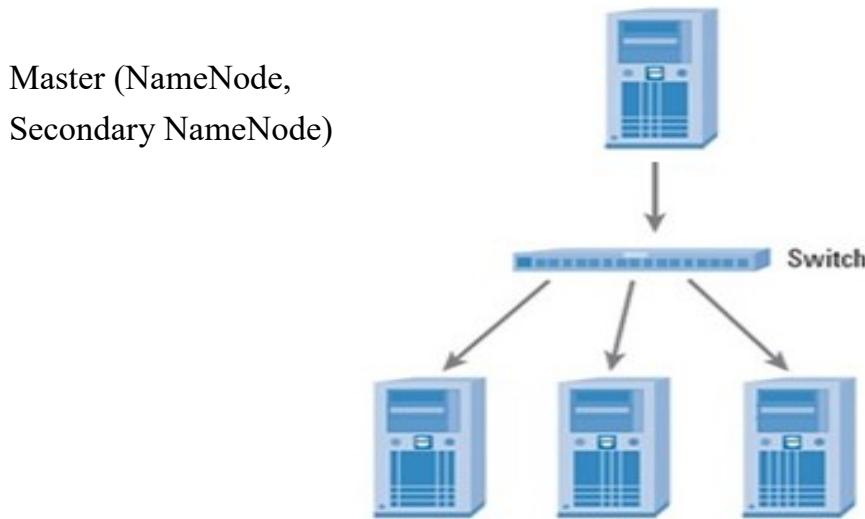
5.3.3. Thực nghiệm

5.3.3.1. Dữ liệu thực nghiệm

Trong các thực nghiệm này, chúng tôi sử dụng 04 cơ sở dữ liệu là DIGITS, VN PLACES, TMW, JVPD.

Phương pháp phân lớp LNBNN yêu cầu cho trước tham số **K** là số lượng hàng xóm gần nhất. Với tham số **K** nhỏ thì độ chính xác phân lớp thấp, với tham số **K** lớn thì phương pháp phân lớp này trở về thuật toán gốc NBNN và thời gian phân lớp tăng. Thực nghiệm và cho thấy **K** nằm trong khoảng từ 10

đến 20 là cho kết quả phân lớp tối ưu với dữ liệu thực nghiệm là phân lớp ảnh [Sancho, 2012] . Trong thực nghiệm này, chúng tôi sử dụng $K=10$.



Hình 5. 6 Mô hình cụm máy tính thực nghiệm

5.3.3.2. Thiết lập môi trường thực nghiệm

Trong thực nghiệm này chúng tôi thiết kế một hệ thống phân tán bao gồm 03 nút được kết nối thông qua mạng cục bộ. Nút chủ (Master node) có cấu hình Intel core i3 processors with 2.4 Hz với 6 GB bộ nhớ hoạt động với vai trò Namenode, Secondary Namenode and Datanode 1.

Nút khách (Slave 1) có cấu hình Intel core i3 processors 2.4 Ghz, 4 GB bộ nhớ đóng vai trò datanode 2. Master và Slave 1 chạy hệ điều hành Mac Os version 10.9.4.

Nút khách 2 (Slave 2) có cấu hình Intel Core™ 2 processors 2.13 Hz với 1 GB memory đóng vai trò Datanode 3. Slave 2 chạy hệ điều hành Ubuntu phiên bản 14.04.2.

Ngôn ngữ lập trình được sử dụng để cài đặt thuật toán LNBNN trên nền Hadoop là JAVA. Apache Hadoop version 2.7.2 được cài đặt trên tất cả các nút. Để trích chọn đặc trưng cho tín hiệu tiếng nói chúng tôi sử dụng bộ trích chọn đặc trưng SIFT trên phổ tần số của tín hiệu tiếng nói.

5.3.3.3. Kết quả thực nghiệm

Kết quả thực nghiệm mô hình phân lớp LNBNN trên nền Hadoop cho bài toán nhận thức tiếng nói được trình bày ở bảng 5.4.

Bảng 5. 3 So sánh độ phân lớp chính xác trên các dữ liệu thực nghiệm

Dữ liệu	Độ chính xác	
	Mapreduce	Memory
JVPD	96.9%	96.9%
English Digits	96.2%	96.2%
VN Places	95.0%	95.0%
TMW	89.9%	NA
VN Places + TMW	87.3%	NA
JVPD+Digits + VN Places + TMW	87.2%	NA

Bảng 5.4 cho thấy độ chính xác của phương pháp phân lớp LNBNN được cài đặt chạy trên bộ nhớ trong và LNBNN được cài đặt chạy song song trên nền Hadoop có kết quả như nhau đối với các bộ dữ liệu được cài đặt bằng cả 2 cách. Đối với cách cài đặt trên nền Hadoop, để có bộ dữ liệu phân lớp lớn hơn để minh họa khả năng xử lý song song, phân tán của Hadoop chúng tôi đã kết hợp các bộ dữ liệu lại với nhau và cho chạy phân lớp trên bộ dữ liệu hợp.

Trong thí nghiệm thứ 2 để so sánh thời gian thực hiện một truy vấn đối với các trường hợp có kích thước dữ liệu huấn luyện khác nhau.

Bảng 5. 4 So sánh thời gian truy vấn trung bình một đặc trưng trên các dữ liệu khác nhau (tính bằng giây)

Bộ dữ liệu	Số đặc trưng	1 nốt	2 nốt	3 nốt
JVPD	489,998	295	302	201
English Digits	581,134	363	245	261
VN Places	3,190,303	1,902	1,858	1,927
TMW	3,605,234	2,253	1,606	1,471
VN Places + TMW	6,795,537	4,281	4,088	4,253
JVPD + English Digits + VN Places + TMW	7,866,669	4,806	4,700	4,938

Bảng 5.5 cho thấy dữ liệu huấn luyện có kích thước càng lớn thì thời gian truy vấn càng lớn, nghĩa là thời gian truy vấn phụ thuộc vào kích thước của dữ liệu huấn luyện. Đối với dữ liệu có kích thước nhỏ việc sử dụng hệ thống hadoop nhiều máy tính không cải thiện được nhiều thời gian truy vấn. Cụ thể

thời gian truy vấn một đặc trưng trong trường hợp chỉ có một máy chủ (Master) và một máy trạm tham gia còn tăng lên đối với bộ dữ liệu JVPD. Đối với dữ liệu huấn luyện lớn như trong bộ dữ liệu kết hợp các bộ dữ liệu thì việc sử dụng hệ thống Hadoop nhiều máy tính sẽ cải thiện đáng kể thời gian truy vấn của một đặc trưng.

Ngoài ra, hiệu năng của hệ thống còn phụ thuộc vào một số yếu tố như phân bố của dữ liệu, năng lực của từng máy tính tham gia vào hệ thống. Trong đó, năng lực của từng nốt có vai trò tương đối quan trọng. Trong thực nghiệm của chúng tôi, máy tính thứ 3 có cấu hình thấp hơn so với các máy tính khác trong cụm, điều này dẫn tới khi thêm máy tính này vào hệ thống và khi chạy với dữ liệu lớn thì thời gian chạy lại có xu hướng tăng lên.

5.4. Kết luận

Ngày nay, với sự gia tăng nhanh chóng của các dịch vụ mạng trực tuyến đã làm gia tăng sự bùng nổ thông tin, đặc biệt là các thông tin đa phương tiện. Do vậy, việc xử lý dữ liệu lớn ngày càng trở nên quan trọng và cấp thiết. Trong chương này chúng tôi đề xuất hai cải tiến cho phương pháp phân lớp LNBNN cho bài toán nhận dạng tiếng nói dựa trên đặc trưng SIFT trích chọn từ phổ tần số của tín hiệu tiếng nói. Một là, chúng tôi đề xuất phương pháp rút gọn đặc trưng bằng việc biến đổi đặc trưng SIFT từ 128 chiều, với mỗi chiều là một byte thành đặc trưng SIFT nhị phân, sau đó mã hóa lại thành một véc tơ 16 chiều để giảm kích thước lưu trữ và tăng tốc độ tính toán. Hai là, chúng tôi đề xuất cài đặt phương pháp phân lớp LNBNN song song, phân tán trên nền tảng Hadoop, một framework nền tảng cho bài toán xử lý dữ liệu lớn. Với việc cài đặt thuật toán LNBNN trên nền tảng Hadoop sẽ cho phép tận dụng được các máy tính hiện có để tạo lập thành cụm máy tính giúp giải quyết được các bài toán dữ liệu lớn trong nhận dạng tiếng nói dựa trên phổ tần số. Từ đó giải quyết được hạn chế lớn nhất của thuật toán LNBNN, đó là việc phải lưu trữ tất cả các đặc trưng của tập huấn luyện. Với xu hướng phát triển mạnh mẽ của các bộ xử lý đồ họa GPU mạnh mẽ, Hadoop được cài đặt trên các máy có trang bị GPU sẽ là một giải pháp hiệu quả vừa tận dụng được sức mạnh của hệ phân tán và tận dụng được sức mạnh của các bộ xử lý đồ họa.

Kết quả nghiên cứu nêu trên được công bố tại kỷ yếu có phản biện của Hội nghị quốc tế lần thứ 3 về National Foundation for Science and Technology Development Conference on Information and Computer Science- NICS 2016 (công trình khoa học số 3) và kỷ yếu có phản biện của Hội nghị quốc tế Công nghệ thông tin và Truyền thông lần thứ 7 – The Seventh Symposium on Information and Communication Technology- SoICT 2016 (công trình khoa học số 4).

KẾT LUẬN

Luận án nghiên cứu hướng tiếp cận học mối quan hệ giữa tín hiệu tiếng nói với các tín hiệu khác cho bài toán nhận thức tiếng nói. Hướng tiếp cận nhằm mô phỏng cơ chế học ngôn ngữ ở người, tín hiệu tiếng nói được thu nhận bởi hệ thính giác đồng thời với việc thu nhận được các tín hiệu thông tin từ các giác quan khác như thị giác, xúc giác, khứu giác và vị giác. Trong khuôn khổ của luận án này, luận án mới mô phỏng việc học mối quan hệ giữa tín hiệu tiếng nói với một khái niệm cho trước và mô phỏng học mối quan hệ giữa tín hiệu tiếng nói với tín hiệu hình ảnh. Các kết quả chính của luận án như sau:

- Đề xuất sử dụng phương pháp trích chọn đặc trưng SIFT từ phổ tần số của tín hiệu tiếng nói dựa trên cơ chế thu nhận đặc trưng tiếng nói của hệ thính giác ở con người kết hợp với phương pháp phân lớp LNBNN cho bài toán nhận thức tiếng nói. Đề xuất mô hình nhận thức tiếng nói bằng mạng tích chập dựa trên phổ tần số của tín hiệu tiếng nói. So sánh kết quả thực nghiệm với mô hình LNBNN kết hợp với đặc trưng SIFT trích từ phổ tần số của tín hiệu tiếng nói.

- Đề xuất xây dựng mô hình nhận thức tiếng nói mô phỏng việc nhận thức của con người ở vùng não liên kết, xây dựng mô hình học mối quan hệ giữa tín hiệu tiếng nói với tín hiệu hình ảnh. Từ đó, đề xuất mô hình nhận thức tiếng nói thông qua học ánh xạ giữa tín hiệu tiếng nói với tín hiệu hình ảnh. Sau khi huấn luyện, mô hình sẽ trả về một hình ảnh phù hợp với tín hiệu tiếng nói đầu vào theo cách đã được huấn luyện.

- Đề xuất cải tiến hiệu năng của mô hình thông qua việc đề xuất phương pháp rút gọn dữ liệu bằng cách mã hóa đặc trưng SIFT từ một véc tơ 128 chiều với mỗi chiều có kích thước một byte dữ liệu thành một véc tơ SIFT nhị phân 128 chiều. Kết quả thực nghiệm trên các bộ dữ liệu huấn luyện cho thấy phương pháp rút gọn dữ liệu này vẫn giữ được độ chính xác của mô hình trong khi giảm kích thước lưu trữ 8 lần. Đề xuất cài đặt phương pháp phân lớp LNBNN trên nền Hadoop, một nền tảng cho bài toán xử lý dữ liệu lớn song song và phân tán. Nền tảng Hadoop, cho phép kết hợp nhiều máy tính có cấu hình thấp hơn để tạo thành một hệ thống xử lý song song, phân tán mạnh hơn, tận dụng được sức mạnh của các hệ thống máy tính hiện có.

Các kết quả trong mô hình nhận thức tiếng nói áp dụng cho bài toán nhận dạng tiếng nói rời rạc chưa thực sự cao so với các phương pháp hiện đại hiện nay, tuy nhiên đây cũng là một hướng nghiên cứu mới làm phong phú thêm các hướng tiếp cận cho bài toán nhận thức tiếng nói. Đặc biệt, mô hình nhận thức tiếng nói thông qua việc xây dựng mạng liên kết giữa các tín hiệu là một hướng tiếp cận hoàn toàn mới cho bài toán nhận thức tiếng nói. Mặc dù độ chính xác của mô hình mới đạt mức 87%, nguyên nhân chủ yếu là do dữ liệu huấn luyện còn ít, chưa đủ để minh họa cho bài toán, hướng tiếp cận này giúp việc huấn luyện người máy trở nên tự nhiên hơn như quá trình học ngôn ngữ ở người. Trong thời gian tới, nhóm tác giả sẽ tập trung nghiên cứu bổ sung thêm cho mô hình các tín hiệu khác mô phỏng cho các cơ quan cảm giác khác của con người, đồng thời, tìm kiếm và xây dựng bộ dữ liệu phù hợp hơn, đủ lớn để nâng cao độ chính xác của mô hình.

Các kết quả nghiên cứu của luận án sẽ là những đóng góp mới về mặt lý thuyết cho lĩnh vực nhận thức tiếng nói, đồng thời có thể ứng dụng trong lĩnh vực giao tiếp người máy, chế tạo người máy. Đây cũng là bước tiền đề để phát triển mô hình nhận thức cho người máy hoàn thiện hơn, gần với quá trình nhận thức của con người thông qua trang bị các bộ cảm biến mô phỏng các cơ quan giác quan của con người, giúp nâng cao thông tin cho hệ thống người máy.

DANH MỤC CÁC CÔNG TRÌNH KHOA HỌC ĐÃ CÔNG BỐ

- [1] Quang Trung, Nguyễn; Thê Duy, Bùi; Thị Châu, Ma; 2015, *An Image based approach for speech perception*, (2015) 2nd National Foundation for Science and Technology Development Conference on Information and Computer Science, Springer, 208 – 213.
- [2] Quang Trung, Nguyen; The Duy, Bui; (2016), *Speech classification using SIFT features on spectrogram images*, Vietnam Journal of Computer Science, 3(4), 247-257.
- [3] The Duy, Bui; Quang Trung, Nguyen; *Speech classification by using binary quantized SIFT features of signal spectrogram images*, (2016), 3rd National Foundation for Science and Technology Development Conference on Information and Computer Science, IEEE.
- [4] Quang Trung, Nguyen; The Duy, Bui; (2016), *MapReduce based for speech classification* SoICT '16: Proceedings of the Seventh Symposium on Information and Communication Technology, ACM.
- [5] The Duy, Bui; Quang Trung, Nguyen; (2016), *Learning relationship between speech and image*, The 8th International Conference on Knowledge and Systems Engineering (KSE) 2016, IEEE, 103-108.
- [6] Quang Trung, Nguyen; The Duy, Bui; (2018), *Speech perception based on mapping speech to image by using convolution neural network*, The 5th NAFOSTED Conference on Information and Computer Science, NICS 2018, IEEE.

TÀI LIỆU THAM KHẢO

Tiếng Việt

1. Đức, Đ., & Mai, L. (2004). Tăng cường độ chính xác của hệ thống mạng nơ-ron nhận dạng tiếng Việt. *Tạp chí Bưu chính viễn thông*, số 11.
2. Dũng, N. M. (2010). Nghiên cứu kỹ thuật nhận dạng người nói dựa trên từ khoá tiếng Việt. Trong *LATS Kỹ thuật*.
3. Hoan, N. (1996). Ôn định mạng nơ-ron Hopfield và khả năng ứng dụng trong điều khiển Robot. *Luận án Tiến sĩ*.
4. Huy, N., Mai, L., Trung, B., Mai, N., Bảng, V., & Hà, V. (2003). Thiết kế các hệ thống nhận dạng Tiếng Việt trong thời gian thực. *Kỷ yếu hội thảo Fair*.
5. Phúc, N. (2000). Một số phương pháp nhận dạng lời Việt: Áp dụng phương pháp kết hợp mạng nơ-ron với mô hình Markov ẩn cho các hệ thống nhận dạng lời Việt. *Luận án tiến sĩ kỹ thuật, Đại học Bách khoa Hà Nội*.

Tiếng Anh

6. Abdel-Hamid, O., Mohamed, A., Jiang, H., Deng, L., Penn, G., & Yu, D. (2014, Oct). Convolutional Neural Networks for Speech Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(10), 1533-1545.
7. Aida-zade, K., Xocayev, A., & Rustamov, S. (2016). Speech recognition using Support Vector Machines. *IEEE 10th International Conference on Application of Information and Communication Technologies (AICT)*, (pp. 1-4). Baku.
8. Alexandre, L. (2016). 3d object recognition using convolutional neural networks with transfer learning between input channels. *Intelligent Autonomous Systems*, Springer, 13, 889-898.
9. Allen, J., & Miller, J. (2004). Listener sensitivity to individual talker differences in voice-onset-time. *Journal of the Acoustical Society of America*, 116, 3171-3183.

10. Angelis, V., Felici, G., & Mancinelli, G. (2006). Feature Selection for Data Mining. In *Data Mining and Knowledge Discovery Approaches Based on Rule Induction Techniques*, 6, 227–252.
11. Bagul, S., & Shastri, R. (2013). Text independent speaker recognition system using GMM. *International Conference on Human Computer Interactions (ICHCI)*, (pp. 1-5). Chennai.
12. Balakrishnama, S., & Ganapathiraju, A. (1999). Linear Discriminant Analysis - a Brief Tutorial. *Compute*, 11, 1–9.
13. Baum, L., & Petrie, T. (1966). Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *The Annals of Mathematical Statistics*(37 (6)), 1554–1563.
14. Bever, T., Lackner, J., & Kirk, R. (1969). The underlying structure sentence is the primary unit of immediate speech processing. *Percep. Psychophys*, (pp. 225–234).
15. Boiman O., Shechtman E., and Iran M. (2008). In Defense of Nearest-Neighbor Based Image Classification. In *CVPR*.
16. Broadbent, D., & Ladefoged, P. (1957). On the fusion of sounds reaching different sense organs. *Journal of the Acoustical Society of America*, 29, 708-710.
17. Chen, X., Ragni, A., Liu, X., & Gales, M. (2017). Investigating Bidirectional Recurrent Neural Network Language Models for Speech Recognition. *International Speech Communication Association (ISCA)*.
18. Christian, S., Wei, L., Yangqing, J., Pierre, S., Scott, R., Dragomir, A., Andrew, R. (2015). Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
19. Davis, S., & Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions*, 28, 357-366.
20. Dominique, F., Odile, M., & Irina, I. (2017). New Paradigm in Speech Recognition: Deep Neural Net-works. *IEEE International Conference on Information Systems and Economic Intelligence*.

21. Fanty, R. C. (1994). ISOLET (Isolated Letter Speech Recognition). *Department of Computer Science and Engineering*, September 12.
22. Fowler, C. (1995). *Speech production - Handbook of Perception and Cognition*. Speech, Language, and Communication. San Diego: Academic Press.
23. Francois, D., Rossi, F., Wertz, V., & Verleysen, M. (2007). Resampling methods for parameter-free and robust feature selection with mutual information. *Neurocomputing*(70(7–9)), 1276–1288.
24. Gheyas, I., & Smith, L. (2010). Feature subset selection in large dimensionality domains. *Pattern Recognition*, 43(1), 5–13.
25. Goldinger, S. (1998). Echoes of echoes? An episodic theory of lexical access. *Psychological Review*, 105, 251–279.
26. Graves, A., Mohamed, A., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, (pp. 6645-6649).
27. Gregory, R. (1987). In *Perception* (pp. 598–601). Gregory, Zangwill.
28. Guenter, E. (1978). Stiffness gradient along the basilar membrane as a way for spatial frequency analysis within the cochlea. *Acoust Soc Am*, 64 (6).
29. Guo, S., Chen, S., & Li, Y. (2017). Face recognition based on convolutional neural network and support vector machine[C]. *IEEE International Conference on Information and Automation*.
30. Halle, M., & Stevens, K. (1962). Speech recognition: A model and a program for research. *IRE Transactions on Information Theory*, (pp. 155-159).
31. Hang, L. (2018, January). Deep learning for natural language processing: advantages and challenges. *National Science Review*, 5(1), 24–26.
32. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Identity Mappings in Deep Residual Networks. *Computer Vision – ECCV 2016*.

33. Hermansky, H. (1990). Perceptual linear predictive (PLP) analysis of speech. *Acoustical Society of America Journal*, 1738–1752.
34. Hickok, G., & Poeppel, D. (2000). Towards a functional neuroanatomy of speech perception. *Trends in Cognitive Science*, 4, 131–138.
35. Hickok, G., & Poeppel, D. (2007). The cortical organization of speech processing. *Nature Reviews Neuroscience*, 8, 393-402.
36. Hillenbrand, J., Clark, M., & Nearey, T. (2001). Effects of consonant environment on vowel formant patterns. *Journal of the Acoustical Society of America*, 109 (2), 748–763.
37. Hillenbrand, J., Getty, L., Clark, M., & Wheeler, K. (1995). Acoustic characteristics of American English vowels. *Journal of the Acoustical Society of America*, 3099–3111.
38. Hong Quang, N., Nocera, P., Castelli, E., & Van Loan, T. (2008). Tone recognition of Vietnamese continuous speech using hidden Markov mode. *Communications and Electronics - ICCE, IEEE*, (pp. 235-239). Hoi an, Viet Nam.
39. Jin, Z., Yang, J., Hu, Z., & Lou, Z. (2001). Face recognition based on the uncorrelated discriminant transformation. *Pattern Recognition*(34(7)), 1405–1416.
40. Johnson, K. (1997). The auditory/perceptual basis for speech segmentation. *Ohio State University Working Papers in Linguistics*, 101-113.
41. Juang, B., & Rabiner, L. (1991). Hidden Markov Models for Speech Recognition. *TECHNOMETRICS*, 33(3).
42. Kadir, A. (2011). Binary SIFT: Fast Image Retrieval Using Binary Quantized SIFT Features. *CBMI*.
43. Kaiming, H., Xiangyu, Z., Shaoqing, R., & Jian, S. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA.
44. Karami, E., Prasad, S., & Shehata, M. (November, 2015). Image Matching Using SIFT, SURF, BRIEF, and ORB: Performance

Comparison for Distorted Images. *Proceedings of the 2015 Newfoundland Electrical and Computer Engineering Conference*. St. John's, Canada.

45. Kim, K., Hong, S., Roh, B., Cheon, Y., & Park, M. (2016). PVANET: Deep but Lightweight Neural Networks for Real-time Object Detection. *arXiv preprint arXiv:1608.08021*.
46. Kimura, D. (1961a). Some effects of temporal-lobe damage on auditory perception. *Canadian Journal of Psychology*, 15, 156-165.
47. Kimura, D. (1961b). Cerebral dominance and the perception of verbal stimuli. *Canadian Journal of Psychology*, 15, 166-171.
48. Kinsner, W., & Peters, D. (1988). A speech recognition system using linear predictive coding and dynamic time warping. *Engineering in Medicine and Biology Society, IEEE*.
49. Klatt, D. (1979). Speech perception: A model of acoustic-phonetic analysis and lexical access. *Journal of Phonetics*, 7, 279–312.
50. Kleesiek, J., Urban, G., Hubert, A., Schwarz, D., Maier-Hein, K., Bendszus, M., & Biller, A. (2016). Deep MRI brain extraction: A 3D convolutional neural network for skull stripping. *Neuroimage*, 129, 460-469.
51. Kohonen, T. (1982). Self-Organized Formation of Topologically Correct Feature Maps. *Biological Cybernetics*(43 (1)), 59–69.
52. Kosko, B. (1987). Adaptive Bidirectional Associative Memories. *Applied Optics*, 23(26), 4947-4960.
53. Kosko, B. (1988). Bidirectional Associative Memory. *IEEE Transaction on Systems, Man, and Cyber*, (pp. 49–60).
54. Krisztina, Z., Jeannette, M., Ton, G., & Louis, C. (2005). Cross-linguistic Comparison of Two-year-old Children's Acoustic Vowel Spaces: Contrasting Hungarian with Dutch. *INTERSPEECH*, (pp. 1173-1176).
55. Krizhevsky, A., Sutskever, I., & Geoffrey, E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *NIPS'12 Proceedings of the 25th International Conference on Neural*

Information Processing Systems - Volume 1, (pp. 1097-1105). Lake Tahoe, Nevada.

56. Kröger, B., Kannampuzha, J., & Neuschaefer-Rube, C. (2009, September). Towards a neurocomputational model of speech production and perception. *Speech Communication*, 51(9), 793-809.
57. Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, (pp. 2278 - 2324).
58. Leibe, B., & Schiele, B. (2004). Scale-invariant object categorization using a scale-adaptive mean-shift search. *Lecture Notes in Computer Science*.
59. Lengeris, A., & Nicolaidis, K. (2014). English consonant confusions by Greek listeners in quiet and noise and the role of phonological short-term memory. *INTERSPEECH*, (pp. 534-538).
60. Leuba, G., & Kraftsik, R. (1994). Changes in volume, surface estimate, three-dimensional shape and total number of neurons of the human primary visual cortex from midgestation until old age. *Anat Embryol*, 190, 351-366.
61. Li, S., Jiang, H., & Pang, W. (2016). Joint multiple fully connected convolutional neural network with extreme learning machine for hepatocellular carcinoma nuclei grading. *Comput. Biol. Med*; vol. 84, (pp. 156-167).
62. Liberman, A., Cooper, F., Shankweiler, D., & Studdert-Kennedy, M. (1967). Perception of the speech code. *Psychological Review*, 74, 431-461.
63. Lowe, D. (1999). Object recognition from local scale-invariant features. *Proceedings of the International Conference on Computer Vision 2*, (pp. 1150–1157).
64. Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *IJCV*.
65. Luce, P., & Pisoni, D. (1998). Recognizing spoken words: the neighborhood activation model. *Ear Hear*, 19, 1–36.

66. Majeed, S., Husain, H., Samad, S., & Idbeaa, T. (2015). Mel frequency cepstral coefficients (mfcc) feature extraction enhancement in the application of speech recognition: a comparison study. *Journal of Theoretical and Applied Information Technology*, 79(1).
67. Marslen-Wilson, W. (1975). Sentence perception as an interactive parallel process. *Science*, 189, 226-228.
68. Marslen-Wilson, W. (1987). Functional parallelism in spoken word recognition. *Cognition*, 25, 71-102.
69. Massaro, D. (1998). *Perceiving Talking Faces: From Speech Perception to a Behavioral Principle*, Cambridge. MA / London, MIT Press.
70. McClelland, J., & Elman, J. (1986). The TRACE Model of Speech Perception. *Cognitive Psychology*, (pp. 1-86).
71. Menezes, P., Oliveira, B., & Morais, S. (2004). Resonance: a study of the outer ear. *NCBI*, 16(3).
72. Miller, G. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63, 81-97.
73. Miller, G. (1962). Decision units in the perception of speech. *IRE Transactions on Information Theory*, (pp. 81–83).
74. Milner, A., & Goodale, M. (1995). *The visual brain in action*. Oxford University Press.
75. Norouzi, M., Punjani, A., & Fleet, D. (2012). Fast Search in Hamming Space with Multi-Index Hashing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
76. Park, C., & Lee, M. (2008). On applying linear discriminant analysis for multilabeled problems. *Pattern Recognition Letters*(29(7)), 878–887.
77. Pickles, C., & James, O. (2012). *An Introduction to the Physiology of Hearing* (4th ed.). Bingley: UK: Emerald Group Publishing Limited.

78. Pisoni, D. (1973). Auditory and phonetic memory codes in the discrimination of consonants and vowels. *Perception and Psychophysics*, 13, 253-260.
79. Purves, D., Augustine, G., & Fitzpatrick, D. (2001). *Neuroscience* (2nd edition ed.). Sunderland (MA): Sinauer Associates.
80. Purves, D., Augustine, G., & Fitzpatrick, D. (2001). *Chapter 13, The Auditory System*. Sunderland (MA): Sinauer Associates.
81. Raul, R. (1996). *Neural Networks*. Springer.
82. Reinhard, S., Andreas, A., & Gerhard, W. (2016). Landmark-based audio fingerprinting for DJ mix monitoring. *International Society for Music Information Retrieval Conference (ISMIR)*.
83. Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*.
84. Rosen, S., & Howell, P. (2011). *Signals and Systems for Speech and Hearing* (2nd ed.). Emerald.
85. Rosenblum, L. D. (n.d.). Primacy of Multimodal Speech Perception. In David Pisoni, Robert Remez. *The Handbook of Speech Perception*, (p. 51).
86. Sak, S. B. (2014). LSTM Recurrent Neural Network architectures for large scale acoustic modeling. *Interspeech*.
87. Sancho, M., & David, G. (2012). Local Naive Bayes Nearest Neighbor for Image Classification. In *CVPR*.
88. Schacter, & Daniel . (2011). *Psychology*. Worth Publishers.
89. Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556* .
90. Smith, R. (2004). *The Role of Fine Phonetic Detail in Word Segmentation*. PhD Dissertation, Department of Linguistics, Cambridge University.
91. Soliz, P., Russell, S., Abramoff, M., Murillo, S., Pattichis, M., & Davis, H. (2008). Independent Component Analysis for Vision-inspired Classification of Retinal Images with Age-related Macular

- Degeneration. *2008 IEEE Southwest Symposium on Image Analysis and Interpretation*, 65–68.
92. Soltau, S. S. (2014). Joint Training of Convolutional and Non-Convolutional Neural Networks. *ICASSP*.
 93. Stevens, K. (1972). The quantal nature of speech: Evidence from articulatory-acoustic data. In *Human Communication: A Unified View* (pp. 51-66). New York: McGraw-Hill.
 94. Stevens, K. (1989). On the quantal nature of speech. *Journal of Phonetics*, 17, pp. 3-45.
 95. Stevens, K. (1998). *Acoustic Phonetics*. Cambridge, MA: The MIT Press.
 96. Sumby, W., & Pollack, I. (1954). Visual contribution to speech intelligibility in noise. *Journal of the Acoustical Society of America*, 26, 212-215.
 97. Sun, Y. (2007). Iterative RELIEF for feature weighting: Algorithms, theories, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*(29(6)), 1035–1051.
 98. Syaffeza, A., Khalil-Hani, M., & Liew, S. (2014). Convolutional neural network for face recognition with pose and Illumination Variation [J]. *International Journal of Engineering & Technology*, 6, 44-57.
 99. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *Computer Vision and Pattern Recognition 2016*.
 100. Thang, V., Tang, K., Son, L., & Chi Mai, L. (2008). Vietnamese tone recognition based on multi-layer perceptron network. *Conference of Oriental Chapter of the International Coordinating Committee on Speech Database and Speech I/O System*, (pp. 253-256). Kyoto.
 101. Tsenov, G., & Mladenov, V. (2010). Speech recognition using neural networks. *10th Symposium on Neural Network Applications in Electrical Engineering*, (pp. 181-186). Belgrade.

102. Tuan, N., & Hai Quan, V. (2009). Advances in Acoustic Modeling for Vietnamese LVCSR. *Asian Language Processing*, (pp. 280-284). Singapore.
103. Van Huy, N., Chi Mai, L., & Tat Thang, V. (2015). Tonal phoneme based model for Vietnamese LVCSR. *Conference of the Oriental chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (OCOCOSDA)*. Shanghai.
104. Vu Ngoc, T., & Schultz, T. (2009). Vietnamese Large Vocabulary Continuous Speech Recognition. *Automatic Speech Recognition & Understanding-ASRU*, (pp. 333 - 338). Merano.
105. Vu Thang, T., Nguyen Dung, T., Chi Mai, L., & Hosom John, P. (2005). Vietnamese large vocabulary continuous speech recognition. *INTERSPEECH*, (p. 1172). Lisbon.
106. Wahab, N., Khan, A., & Lee, Y. (April 2017). Two-phase deep convolutional neural network for reducing class skewness in histopathological images based breast cancer detection. *Comput. Biol. Med.*; vol. 85;, (pp. 86-97).
107. Wanda, G. (2017). *Neurology for the Speech-Language Pathologist*. (S. Edition, Ed.) Webb PhD.
108. Wang, H. (2006). A Multi-Space Distribution (MSD) Approach to speech recognition of tonal languages. *INTERSPEECH*. Pittsburgh, USA: IEEE.
109. Wiener, E., Pedersen, J., & Weigend, A. (1995). A neural network approach to topic spotting. *Proceedings of SDAIR95 4th Annual Symposium on Document Analysis and Information Retrieval*, (pp. 317–332).
110. Wróblewska, A., & Sydow, M. (December 4-7, 2012). DEBORA: dependency-based method for extracting entity-relationship triples from open-domain texts in Polish. *In Foundations of Intelligent Systems -20th International Symposium (ISMIS) 2012*, (pp. 155–161). China.

111. Xiaofan, X., Alireza, D., David, C., Sam, C., & David, M. (2016). Convolutional Neural Network for 3D object recognition using volumetric representation. *Sensing, Processing and Learning for Intelligent Machines (SPLINE), 2016 First International Workshop on*.
112. Xu-Yao, Z., Yoshua, B., & Cheng, L. (2017, January). Online and offline handwritten Chinese character recognition: A comprehensive study and new benchmark. *Pattern Recognition*, 61, 348-360.
113. Yang, C., Wang, L., & Feng, J. (2009). A novel margin based algorithm for feature extraction. *New Generation Computing*(27(4)), 285–305.
114. Yang, J., Frangi, A., Yang, J., Zhang, D., & Jin, Z. (2005). KPCA plus LDA: A complete kernel fisher discriminant framework for feature extraction and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*(27(2)), 230–244.
115. Yin, W., Kann, K., Yu, M., & Schütze, H. (2017). Comparative Study of CNN and RNN for Natural Language Processing. *arXiv:1702.01923*.
116. Yuen, P., & Lai, J. (2002). Face representation using independent component analysis. *Pattern Recognition*(35(6)), 1247–1257.
117. Zeiler, M., & Fergus, R. (2014). Visualizing and Understanding Convolutional Networks. *Computer Vision – ECCV 2014*.
118. Zhang, M., Peña, J., & Robles, V. (2009). Feature selection for multi-label naive Bayes classification. *Information Sciences*(179(19)), 3218–3229.
119. Zhang, X., Zhu, B., Li, L., & et al. (2015, February). SIFT-based local spectrogram image descriptor: a novel feature for robust music identification. *EURASIP Journal on Audio, Speech, and Music Processing*, 6.