BỘ GIÁO DỤC VÀO ĐÀO TẠO
VIỆN HÀN LÂM
KHOA HỌC VÀ CÔNG NGHỆ VN

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

Họ và tên tác giả luận án

NGUYỄN VĂN HUY

TÊN ĐỀ TÀI LUẬN ÁN

Nghiên cứu mô hình thanh điệu trong nhận dạng tiếng Việt từ vựng lớn phát âm liên tục

LUẬN ÁN TIẾN SĨ: TOÁN HỌC

HÀ NỘI – 2016

BỘ GIÁO DỤC VÀO ĐÀO TẠO **VIỆN HÀN LÂM KHOA HỌC VÀ CÔNG NGHỆ VN**

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

Họ và tên tác giả luận án

NGUYỄN VĂN HUY

TÊN ĐỀ TÀI LUẬN ÁN

Nghiên cứu mô hình thanh điệu trong nhận dạng tiếng Việt từ vựng lớn phát âm liên tục

Chuyên ngành: **Cơ sở toán học cho tin học**

Mã số: 62460110

LUẬN ÁN TIẾN SĨ: TOÁN HỌC

NGƯỜI HƯỚNG DẪN KHOA HỌC:

1. **PGS. TS. LƯƠNG CHI MAI**
2. **TS. VŨ TẤT THẮNG**

HÀ NỘI – 2016

Lời nói đầu

Các kỹ thuật nhận dạng tiếng nói đã đang rất phát triển, đặc biệt với một số ngôn ngữ phổ dụng như Anh, Pháp, Trung Quốc,… Những yếu tố chính ảnh hưởng đến chất lượng của một hệ thống nhận dạng tiếng nói như: Người nói, tốc độ nói, hoàn cảnh nói, nhiễu, kích thước từ điển, cách thức phát âm,… tuy nhiên hiện nay vẫn chưa có một giải pháp nào hoàn thiện giải quyết tất cả các yếu tố đó. Các phương pháp cơ bản thường được sử dụng cho nhận dạng tiếng nói là: Kỹ thuật so khớp mẫu, mạng nơron, phương pháp dựa trên tri thức và mô hình Markov ẩn. Trong đó phương pháp sử dụng mô hình Markov ẩn (Hidden Markov Model HMM) được sử dụng phổ biến nhất.

Đối với tiếng Việt hiện nay vẫn chưa có nhiều nghiên cứu về nhận dạng. Các công việc nghiên cứu mới đang ở những bài toán cơ bản. Tiếng Việt là một ngôn ngữ có thanh điệu, vì thế ngoài những khó khăn gặp phải tương tự như việc nhận dạng các ngôn ngữ không có thanh điệu khác (Anh, pháp,…), nhận dạng tiếng Việt còn phải nghiên cứu vấn đề nhận dạng thanh điệu. Tiếng Việt có sáu thanh điệu, một cách tổng quát có thể coi như mỗi âm tiết sẽ có thể có sáu ý nghĩa khác nhau khi ghép tương ứng với sáu thanh điệu đó. Việc nhận dạng thanh điệu là một công việc khó do thanh điệu chỉ tồn tại ở vùng âm hữu thanh. Vì thế đường đặc tính của nó không liên tục khi chuyển tiếp giữa hai vùng hữu thanh và vô thanh. Các đặc trưng được sử dụng phổ biến trong nhận dạng tiếng nói như MFCC (Mel Frequency Cepstral Coefficient) và PLP (Perceptual Linear Prediction) lại không mô tả được các đặc tính của thanh điệu, do vậy trước khi nhận dạng được thanh điệu ta phải áp dụng các kỹ thuật tính toán đặc trưng thanh điệu trong tín hiệu tiếng nói.

Các nghiên cứu hiện nay về nhận dạng thanh điệu tiếng Việt cũng mới chỉ ở những bước đầu tiên và chủ yếu áp dụng cho tiếng nói rời rạc, có lượng từ vựng nhỏ cỡ vài trăm từ. Các giải pháp chủ yếu là phát triển từ các nghiên cứu trên các ngôn ngữ có thanh điệu khác như Mandarin, Thái,…, vì vậy việc nghiên cứu một giải pháp nhận dạng tiếng Việt từ vựng lớn phát âm liên tục thực sự là một vấn đề cấp thiết cả về tính khoa học và kinh tế.

Từ các lý do cấp thiết này tôi đã chọn đề tài "**Nghiên cứu mô hình thanh điệu trong nhận dạng tiếng Việt từ vựng lớn phát âm liên tục**". Với mục tiêu chính là nghiên cứu các vấn đề trong nhận dạng tiếng Việt từ vựng lớn phát âm liên tục, và nghiên cứu các vấn đề về mô hình thanh điệu cho tiếng Việt.

Nội dung chính của luận án được trình bày thành 5 chương với nội dung như sau:

- Chương 1: Giới thiệu tổng quan về nhận dạng tiếng nói và ứng dụng. Cấu trúc tổng quan của một hệ thống nhận dạng tiếng nói cơ bản. Tình hình nghiên cứu tổng quan về nhận dạng tiếng nói chung và nhận dạng tiếng Việt nói riêng. Giới thiệu các mục tiêu và phạm vi nghiên cứu chính của luận án.
- Chương 2: Trình bày tổng quan về cấu trúc ngữ âm tiếng Việt. Mô hình nhận dạng tiếng Việt từ vựng lớn phát âm liên tục có thanh điệu. Dữ liệu và các công cụ sử dụng để cài đặt các thử nghiệm. Hệ thống nhận dạng cơ sở.
- Chương 3: Trình bày mô hình thanh điệu cho nhận dạng tiếng Việt từ vựng lớn phát âm liên tục sử dụng MSD-HMM. Bao gồm quy trình tính toán đặc trưng thanh điệu, cấu hình mô hình và huấn luyện.
- Chương 4: Trình bày phương pháp tăng cường đặc trưng ngữ âm sử dụng mạng nơron cho nhận dạng tiếng Việt, bao gồm quy trình gán nhãn, huấn luyện mạng, tối ưu mạng, trích chọn đặc trưng Bottleneck và cài đặt thử nghiệm.
- Chương 5: Trình bày phương pháp tăng cường đặc trưng thanh điệu với đặc trưng cải tiến Tonal-Bottleneck sử dụng mạng nơron. Bao gồm phương pháp gán nhãn thanh điệu, tối ưu mạng, tính toán đặc trưng và cài đặt thử nghiệm.

Tôi xin được gửi lời cảm ơn chân thành đến Bộ Giáo dục và Đào tạo, Viện Công nghệ Thông tin – Viện Hàn lâm Khoa học và Công nghệ Việt Nam, trường ĐH Kỹ thuật Công nghiệp Thái Nguyên – ĐH Thái Nguyên đã tạo điều kiện thuận lợi cho tôi hoàn thành đề tài nghiên cứu sinh này. Xin được gửi lời cảm ơn chân thành đến Viện công nghệ Karlshuhe – Đức, Viện Công nghệ Thông tin quốc gia Nhật Bản đã tạo điều kiện và hỗ trợ cả về mặt khoa học lẫn thiết bị cho tôi để thực hiện các thử nghiệm và các nghiên cứu trong quá trình thực tập sinh tại Đức và Nhật Bản.

Tôi xin được gửi lời cảm ơn đặc biệt đến PGS. TS. Lương Chi Mai, TS. Vũ Tất Thắng đã luôn chỉ bảo, định hướng, tạo điệu kiện thuận lợi nhất để tôi có thể hoàn thành luận án này.

Thái Nguyên, ngày 16 tháng 08 năm 2016

Nguyễn Văn Huy

Lời cam đoan

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của PGS.TS. Lương Chi Mai và TS. Vũ Tất Thắng. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa từng được công bố trước đây bởi người khác. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các thử nghiệm.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung luận án của mình.

Tác giả

Nguyễn Văn Huy

Mục Lục

Danh mục các từ viết tắt

TT	Viết tắt	Nghĩa
1	ACC	Accuracy
2	AMDF	Average Magnitude Difference Function
3	BN	Bottleneck
4	BNF	Bottleneck Feature
5	CV	Cross Validation Accuracy
6	DCT	Discrete cosine transform
7	DFT	Discrete Fourier transform
8	DNN	Deep Neural Network
9	F0	Fundamental Frequency
10	FST	Finite-State Transducer
11	G2P	Grapheme to Phoneme
12	GMM	Gaussian Mixture Model
13	GPU	Graphical processing unit
14	HMM	Hidden Markov Model
15	HTK	Hidden Markov Model Toolkit
16	HTS	HMM-based Speech Synthesis System
17	IDFT	Invert Discrete Fourier transform
18	IOIT2013	Institute Of Information and Technology 2013
19	IPA	International Phonetic Alphabet
20	LDA	Linear Discriminant Analysis
21	LM	Language Model
22	MFCC	Mel Frequency Cepstral Coefficients
23	MLLT	Maximum Likelihood Linear Transform
24	MLP	Multilayer Perceptron
25	MSD	Multispace Distribution
26	NCC	Normalized Cross-Correlation
27	NN	Neural Network
28	NoTone	No tone
29	P	Pitch
30	PLP	Perceptual Linear Prediction
31	T1	Tone 1
32	T2	Tone 2
33	T3	Tone 3
34	T4	Tone 4
35	T5	Tone 5
36	T6	Tone 6

37	TBNF	Tonal Bottleneck Feature
38	VN-G2P	Vietnamese Grapheme to Phoneme
39	VoiceTra	Voice Translation
40	VOV	Voice Of Vietnam
41	WER	Word Error Rate
42	Δ	Delta

Danh mục bảng biểu

Danh mục hình ảnh

Chương 1: Mở đầu

1.1. Tóm tắt chương

Giới thiệu tổng quan về nhận dạng tiếng nói và ứng dụng. Các vấn đề khó khăn cần giải quyết trong lĩnh vực nhận dạng tiếng nói. Giới thiệu về các thành phần cơ bản trong hệ thống nhận dạng tiếng nói từ vựng lớn. Giới thiệu tổng quan về tình hình nghiên cứu nhận dạng tiếng Việt trong và ngoài nước. Giới thiệu các nội dung nghiên cứu chính của luận án.

1.2. Tổng quan về nhận dạng tiếng nói

1.2.1. Nhận dạng tiếng nói

Nhận dạng tiếng nói là quá trình biến đổi tín hiệu âm thanh thu được của người nói thành một chuỗi các từ có nội dung tương ứng dưới dạng văn bản. Nếu gọi tín hiệu tiếng nói thu được trên miền thời gian là *s(t)* thì *s(t)* đầu tiên sẽ được rời rạc hóa để xử lý và trích chọn ra các thông tin quan trọng. Kết quả thu được là một chuỗi các vector đặc trưng tương ứng $X=\{x_1, x_2, x_3, \ldots, x_N\}$. Sau đó nhiệm vụ của hệ thống nhận dạng tiếng nói là tìm ra một chuỗi các từ, $\widehat{W}=\{w_1, w_2, w_3, \ldots, w_L\}$ có nội dung tương ứng với *X* về mặt ngữ nghĩa. Công thức (1.1) [Jurafsky 2008] mô tả mô hình toán học của một hệ thống nhận dạng tiếng nói theo nguyên lý xác suất của Bayes. Hầu hết các hệ thống nhận dạng tiếng nói thống kê ngày nay đều dựa trên mô hình này.

$$\widehat{W} = \underset{w}{\operatorname{argmax}} P(W|X) = \underset{w}{\operatorname{argmax}} \frac{P(X|W)P(W)}{P(X)} \quad (1.1)$$

Trong đó *P(W)* là xác suất của chuỗi *W*, giá trị này có thể được tính toán thông qua một mô hình ngôn ngữ n-gram và nó hoàn toàn độc lập với tín hiệu tiếng nói *X*. *P(X|W)* là xác suất để *X* là *W* được xác định thông qua mô hình âm học (acoustic model). Giá trị *P(X)* có thể được bỏ qua do giá trị của nó không thay đổi trong một bộ dữ liệu cụ thể với tất các chuỗi dự đoán *W*.

Các hệ thống nhận dạng tiếng nói hiện nay có thể được phân loại theo các cách như sau:

- Nhận dạng các từ phát âm rời rạc hoặc liên tục.
- Nhận dạng tiếng nói phụ thuộc hoặc không phụ thuộc người nói.
- Nhận dạng với hệ thống từ vựng nhỏ (vài trăm từ) hoặc từ vựng lớn (hàng nghìn từ).
- Nhận dạng tiếng nói trong môi trường nhiễu cao hoặc thấp.

1.2.2. Ứng dụng

Cùng với sự phát triển nhanh chóng của các thiết bị tính toán tốc độ cao như máy tính, điện thoại thông minh, vi xử lý- vi điều khiển, ngày nay nhận dạng tiếng nói được ứng dụng cho rất nhiều các lĩnh vực trong cuộc sống. Có thể kể đến một số ứng dụng trong một số lĩnh vực chính như sau:

- Trong ngành công nghiệp ôtô: Nhận dạng tiếng nói được ứng dụng để xây dựng các module tương tác giữa người lái với xe ôtô. Hãng xe Audi của Đức là một trong các hãng xe đã ứng dụng thành công và phổ biến công nghệ này. Người lái xe có thể tắt mở hệ thống âm thanh, điều chỉnh âm lượng, hoặc ra lệnh tìm đích đến cho hệ thống dẫn đường bằng giọng nói.
- Trong lĩnh vực y tế: Nhận dạng tiếng nói có thể được ứng dụng để tạo ra các hệ thống nhập hoặc tìm kiếm thông tin bệnh nhân tự động. Người bệnh có thể trả lời các câu hỏi trên một mẫu phiếu khai đã được tích hợp vào một hệ thống nhận dạng tiếng nói khi khám bệnh, hệ thống này sẽ nhận dạng tín hiệu tiếng nói của người bệnh và dịch nó sang dạng văn bản để điền tự động vào mẫu văn bản trên máy tính. Đối với các bệnh viện lớn, nhận dạng tiếng nói cũng có thể được ứng dụng để xây dựng các hệ thống tìm kiếm thông tin bệnh nhân đã có sẵn trong hồ sơ của bệnh viện. Nếu hồ sơ của bệnh nhân đã có trong bệnh viện, người bệnh chỉ cần nói một câu bất kỳ, hệ thống sẽ nhận dạng và tìm ra số hiệu của bệnh nhân đó thông qua giọng nói đặc trưng của họ, từ đó tự động tìm kiếm hồ sơ trong cơ sở dữ liệu. Nhận dạng tiếng nói còn được ứng dụng để ghi chép và tóm tắt tự động các đánh giá, nhận xét hoặc các lời khuyên của bác sỹ vào đơn thuốc của bệnh nhân.
- Trong quân đội:
    - Đối với các phi công lái máy chiến đấu, thông thường họ phải thực hiện nhiều thao tác trong quá trình điều khiển máy bay. Các thao tác này lại yêu cầu chính xác và nhanh. Nhận dạng tiếng nói có thể được ứng dụng để xây dựng các hệ thống tương tác bằng tiếng nói hỗ trợ phi công như: thiết lập tần số radio; chỉ huy hệ thống lái tự động; thiết lập tọa độ và thông số vũ khí; kiểm soát hiển thị chuyến bay. Các hệ thống này góp phần đáng kể trong việc giảm khối lượng công việc và nâng cao hiệu quả cũng như độ chính xác trong việc điều khiển máy bay cho các phi công. Trong thực tế các hệ thống như thế này đã được không quân Mỹ và Pháp ứng dụng cho các máy bay chiến đấu như F-16[1] và Mirage[2].

[1]http://www.f-16.net/f-16_versions_article19.html
[2]http://www.airforce-technology.com/projects/mirage/

- Đối với máy bay trực thăng: Việc trao đổi thông tin qua radio trên máy bay trực thăng gặp rất nhiều khó khăn và trở ngại do ảnh hưởng của tiếng ồn. Trong trường hợp này nhận dạng tiếng nói được ứng dụng để xây dựng các hệ thống hỗ trợ liên lạc, nó có chức năng xử lý và nhận dạng tiếng nói của phi công trong môi trường ồn nhằm nâng cao độ chính xác của thông tin trong trường hợp con người gặp khó khăn trong việc nghe thông tin từ phi công. Các hệ thống như thế này đã được nghiên cứu và ứng dụng trong thực tế trên các máy bay trực thăng của quân đội Hoa kỳ [Womak 1996] và Pháp[3].

- Trong viễn thông và giải trí: Đây là lĩnh vực mà nhận dạng tiếng nói được ứng dụng rộng rãi và đa dạng nhất. Trong viễn thông nhận dạng tiếng nói được áp dụng để xây dựng các tổng đài trả lời tự động bằng cách nhận dạng và phân loại câu hỏi của người gọi, hoặc các hệ thống dịch vụ tự động. Trong giải trí nhận dạng tiếng nói được áp dụng để tạo ra các thiết bị cho phép người điều khiển sử dụng tiếng nói để tương tác với thiết bị. Có thể kể đến rất nhiều các hệ thống cũng như các công ty lớn trên thế giới đã áp dụng công nghệ này như: Google, Microsoft Corporation (Microsoft Voice Command), Digital Syphon (Sonic Extractor), LumenVox, Nuance Communications (Nuance Voice Control), VoiceBox Technology,…
- Trong giáo dục: Các hệ thống nhận dạng tiếng nói có độ chính xác cao sẽ là rất hữu ích cho những người muốn học một ngôn ngữ thứ hai. Các hệ thống nhận dạng có thể được dùng để đánh giá độ phát âm chính xác của người học [Ambra 2003].
- Đối với người khuyết tật: Nhận dạng tiếng nói có thể giúp những người khuyết tật vận động vẫn có thể đi lại trên xe lăn hoặc sử dụng các thiết bị điện tử như máy tính, điện thoại hay ti vi bằng cách gửi lệnh điều khiển thông qua giọng nói.
- Trong giao tiếp: Với mục tiêu xóa bỏ rào cản ngôn ngữ, nhận dạng tiếng nói được ứng dụng để xây dựng các hệ thống dịch máy tự động nhằm giúp con người có thể nói chuyện với nhau bằng tiếng mẹ đẻ của mình ở bất kỳ đâu trên thế giới. Nhận dạng tiếng nói là một khâu trong hệ thống này, nó thu thập tín hiệu tiếng nói, nhận dạng và chuyển thành dạng văn bản. Sau đó phần dịch tự động sẽ chuyển nội dung văn bản này sang một văn bản khác ở một ngôn ngữ khác với cùng một nội dung. Hiện nay đã có một số phần mềm đã được đưa vào ứng dụng như: Phần mềm Siri chạy trên hệ điều hành IOS của công ty Apple. Phần mềm Voicetra chạy trên hệ điều hành IOS và Android của Viện công nghệ

[3]http://www.helis.com/database/model/84/

thông tin Nhật bản (NICT). Hệ thống dịch bài giảng tự động LectureTra của học viện Karlsuhe-Đức (KIT).

- Còn rất nhiều các ứng dụng khác có thể kể ra như công nghệ nhà thông minh, nhập dữ liệu bằng giọng nói, robot, …

Từ các ứng dụng tiêu biểu như trên cho thấy những ý nghĩa khoa học cũng như ý nghĩa về ứng dụng trong cuộc sống của nhận dạng tiếng nói là rất đa dạng và hữu ích. Nó khẳng định việc nghiên cứu và ứng dụng nhận dạng tiếng nói trong cuộc sống vẫn còn tiếp tục đặt ra những thách thức và nhiều bài toán khó cho các nhà khoa học.

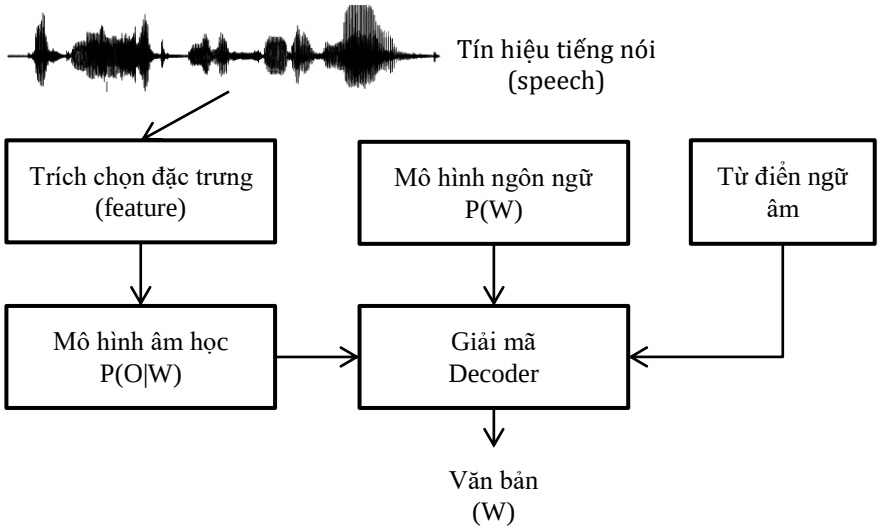
1.2.3. Các vấn đề trong nhận dạng tiếng nói

Nhận dạng tiếng nói là một dạng bài toán trong lĩnh vực nhận dạng mẫu, vì vậy cũng tồn tại những khó khăn tương tự như các bài toán nhận dạng khác. Ngoài ra còn tồn tại một số vấn đề khác do đặc tính biến đổi ngẫu nhiên của tín hiệu tiếng nói. Các vấn đề chính ảnh hưởng đến độ chính xác và hiệu suất làm việc của một hệ thống nhận dạng tiếng nói [Tebelskis 1995] [Đức 2003] [Jurafsky 2008] [Lei 2006] có thể kể đến như sau:

- 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. Xét theo đặc tính phụ thuộc người nói thì nhận dạng tiếng nói có thể phân chia làm hai loại. Một là nhận dạng tiếng nói phụ thuộc người nói, các hệ thống này được xây dựng chuyên biệt để chỉ làm việc với tiếng nói của một người hoặc vài người nhất định. Loại thứ hai là nhận dạng độc lập với người nói, tức là hệ thống nhận dạng được xây dựng để nhận dạng cho tiếng nói của bất kỳ người nào. Thông thường tỷ lệ lỗi nhận dạng tiếng nói của hệ thống độc lập với người nói thường cao hơn so với hệ thống nhận dạng tiếng nói phụ thuộc người nói.
- Vấn đề về tốc độ phát âm, hiện tượng đồng phát âm: 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.

- Vấn đề về kích thước của bộ từ vựng (từ điển): Kích thước từ điển là số lượng tất cả các từ khác nhau mà một hệ thống nhận dạng cụ thể có khả năng nhận dạng được. Kích thước bộ từ điển càng lớn thì độ phức tạp của hệ thống nhận dạng càng cao. Tỷ lệ lỗi của hệ thống nhận dạng luôn tỷ lệ thuận với kích thước của bộ từ điển.
- 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 gây ra những khó khăn đặc biệt 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 muốn nhận dạng thì chất lượng nhận dạng cho các trường hợp này vẫn còn rất thấp.
- Vấn đề về ngôn ngữ: Mỗi một ngôn ngữ lại có bộ ký tự, bộ âm vị mang đặc trưng riêng. Việc nghiên cứu và tìm ra được tập âm vị chuẩn cho một ngôn ngữ sẽ nâng cao độ chính xác nhận dạng. Đối với từng ngôn ngữ thì vấn đề ngữ pháp của phát âm cũng ảnh hưởng rất nhiều đến chất lượng nhận dạng. Các phát âm theo một cấu trúc cú pháp đầy đủ và rõ ràng sẽ được nhận dạng chính xác hơn là một phát âm tự do, tức là các từ trong phát âm không có ràng buộc cụ thể về ngữ pháp.

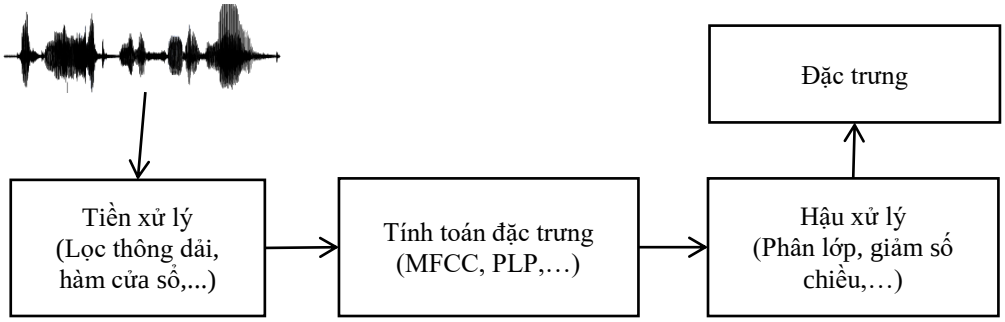
1.3. Các thành phần chính của một hệ thống nhận dạng tiếng nói



Hình 1-1: Sơ đồ khối tổng quan của một hệ thống nhận dạng tiếng nói

Cấu trúc tổng quát của một hệ thống nhận dạng tiếng nói được mô tả ở Hình 1-1.

1.3.1. Trích chọn đặc trưng



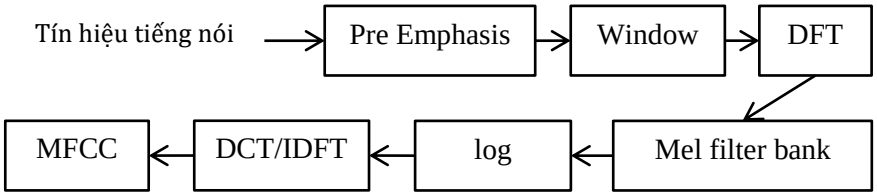
Hình 1-2: Sơ đồ các bước trích chọn đặc trưng

Khâu trích chọn đặc trưng áp dụng một số kỹ thuật nhằm làm giảm độ phức tạp của tín hiệu tiếng nói đầu vào, đồng thời rút trích các thông tin quan trọng và có ý nghĩa cho việc mô hình hóa và nhận dạng. Đầu ra thu được một chuỗi các vector đặc trưng (hay còn gọi là các quan sát) ký hiệu là *O*. Khâu này có thể chia ra làm ba giai đoạn gồm tiền xử lý, tính toán đặc trưng và hậu xử lý như mô tả ở Hình 1-2.

a) Khâu tiền xử lý: Có nhiệm vụ chính là lọc nhiễu, rút trích các tín hiệu nằm trong miền tần số mà tai người nghe được (0-10kHz), chia tín hiệu tiếng nói thành các khung có kích thước từ 10ms đến 30ms (còn gọi là hàm cửa sổ Window), độ lệch giữa hai khung liên tiếp thường nằm trong khoảng 10ms-20ms.
b) Khâu tính toán đặc trưng: Biến đổi tín hiệu sang miền tần số qua phép biến đổi Fourier rời rạc (DFT), thực hiện các tính toán để thu được đặc trưng. Hai loại đặc trưng được sử dụng phổ biến trong nhận dạng tiếng nói là các hệ số đường bao phổ của tần số mel (Mel Frequency Cepstral Coefficient - MFCC) và mã dự báo tuyến tính giác quan (Perceptual Linear Prediction - PLP).
c) Khâu hậu xử lý: Để nâng cao chất lượng đặc trưng và giảm kích thước vector đặc trưng trước khi đưa vào mô hình ngôn ngữ. Một trong các phương pháp phân lớp và giảm số chiều thường được áp dụng trong nhận dạng tiếng nói là phương pháp phân tích tuyến tính LDA.

1.3.1.1. Đặc trưng MFCC

Đây là một trong những loại đặc trưng được sử dụng phổ biến trong nhận dạng tiếng nói. Ý tưởng chính của MFCC tính toán các giá trị phổ của tín hiệu cho băng tần trên miền tần số mà tai người dễ cảm thụ nhất. Sơ đồ khối các bước để tính toán đặc trưng MFCC trên tín hiệu tiếng nói đầu vào được trình bày ở Hình 1-3 [Jurafsky 2008].



Hình 1-3: Sơ đồ khối các bước tính toán MFCC

Trong đó:

- **Pre Emphasis**: Do tai người chỉ nhạy cảm với các tần số thấp nên một hàm tăng cường tín hiệu theo công thức (1.2) cho các tần số cao được áp dụng trước khi tín hiệu được đưa vào tính toán ở các bước sau.

$$s(n) = x(n) - a * x(n-1) \quad (1.2)$$

Trong đó *x(n)* là tín hiệu vào, *a* là hệ số (trong luận án này *a*=0.95)

- **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 (Window) 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. Hình 1-4 minh họa quá trình phân chia cửa sổ cho một tín hiệu tiếng nói với kích thước cửa sổ là 25ms và khoảng cách giữa hai khung (độ dịch khung) là 10ms. Hàm cửa sổ áp lên mỗi khung thường là hàm Hamming với công thức sau:

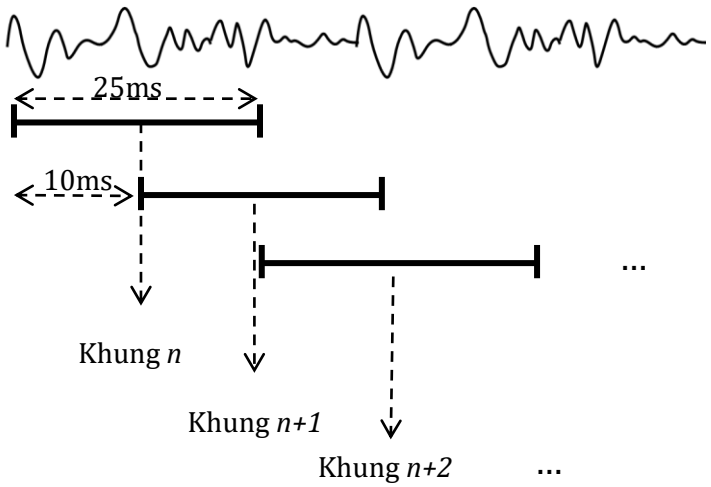
$$W(n) = \left\{0.54 - 0.46\cos\left(\frac{2\pi n}{L}\right)\right\} \quad (1.3)$$

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 của tín hiệu đầu vào. 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 (1.4).

$$X(k) = \sum_{n=0}^{L-1} y[n] e^{-j2\frac{\pi}{L}kn} \qquad (1.4)$$



Hình 1-4: Tạo khung trên tín hiệu tiếng nói

Trong đó: *L* là kích thước của cửa sổ, *w[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 âm thanh thường dao động trong khoảng dưới 10kHz, tuy nhiên tai người chỉ nhạy cảm hay nghe rõ nhất trong khoảng 1kHz. Các hệ thống nhận dạng cố 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 Hz sang miền tần số mà con người dễ nghe nhất. Miền tần số này gọi là Mel (được đặt đề xuất bởi Steven and Volkmann, 1940). Công thức biến đổi được mô tả ở công thức (1.5).

$$mel(f) = 2595 log_{10}(1 + \frac{f}{700}) \qquad (1.5)$$

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

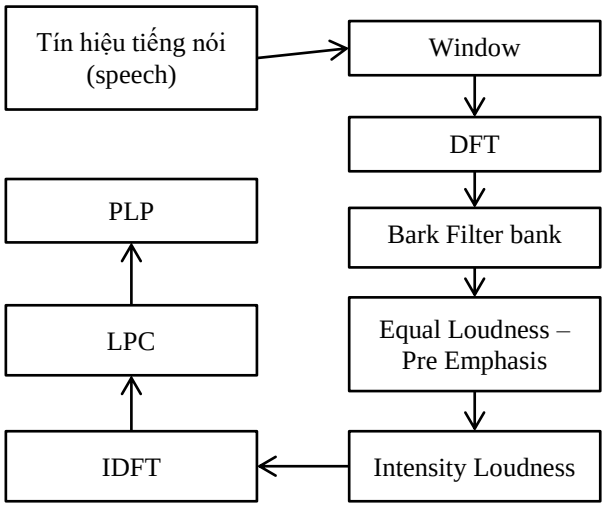
- **Logarithm (log) và biến đổi Cosine rời rạc (DCT)**: Hàm logarithm được áp dụng trên các giá trị DFT đo độ thính của tai người theo hàm logarithm, vì vậy việc áp dụng hàm log để đưa đặc trưng tính toán được gần giống với tín hiệu

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 (1.6) để thu được các giá trị MFCC.

$$C[k] = \sum_{n=0}^{L-1} \log(|X[k]|) e^{j\frac{2\pi}{L}kn} \qquad (1.6)$$

1.3.1.2. Đặc trưng PLP

Phương pháp tính toán đặc trưng PLP dựa trên cơ sở phương pháp mã dự báo tuyến tính LPC (Linear Prediction Coding). Đặc trưng này được tạo ra dựa trên đặc tính vật lý của tai người khi nghe [H. Hermansky 1990]. Hình 1-5 miêu tả các bước xử lý tính toán PLP.



Hình 1-5: Sơ đồ khối các bước tính toán PLP

Trong đó:

- **Windows và DFT**: Là khâu lấy cửa sổ và biến đổi Fourier rời rạc. Bước này thực hiện tương tự như ở MFCC.
- **Bark Filter bank**: Tín hiệu tiếng nói sau bước DFT được lọc theo thang tần phi tuyến Bark theo công thức (1.7).

$$Bark(f) = 6ln\left\{\frac{f}{1200} + \left[\left(\frac{f}{1200}\right)^2 + 1\right]^{0.5}\right\} \qquad (1.7)$$

- **Equal Loudness – Pre Emphasis**: Tăng cường tín hiệu sử dụng hàm Equal Loudness như công thức (1.8).

$$E(\omega) = \frac{(\omega^2 + 56.8 * 10^6)\omega^4}{(\omega^2 + 6.3 * 10^6)(\omega^2 + 0.38 * 10^9)(\omega^6 + 9.58 * 10^{26})} \qquad (1.8)$$

- **Intensity Loudness**: Dùng một phép ánh xạ phi tuyến để làm tăng đặc tính năng lượng của tín hiệu tương đồng với cách thức mà tai nghe âm thanh. Phép ánh xạ này mô tả ở công thức (1.9).

$$\Phi(f) = \Psi(f)^{0.33} \qquad (1.9)$$

- **IDFT**: Biến đổi Fourier ngược tượng tự như công thức (1.6)
- **LPC**: Thuật toán tính toán các hệ số dự báo tuyến tính theo thuật toán Levinson-Durbin [Levinson 1947].

1.3.2. Mô hình âm học

Nhận dạng tiếng nói từ vựng lớn phát âm liên tục thường sử dụng mô hình xác suất để mô hình hóa các đơn vị nhận dạng của hệ thống. Mỗi mô hình âm học có thể coi như một hàm xác suất *P(O|W)* để đi xác định xác suất để một vector đặc trưng đầu vào *O* là đầu ra *W*. Các tham số của hàm *P(O|W)*được xác định thông qua quá trình huấn luyện trên một tập mẫu có trước. Dữ liệu huấn luyện ảnh hưởng trực tiếp đến độ chính xác của mô hình âm học. Trong thực tế với các hệ thống nhận dạng cho tập từ vựng lớn thì mô hình âm học thường được áp dụng để mô hình các âm vị độc lập ngữ cảnh (mono-phone) hay phụ thuộc ngữ cảnh (tri-phone). Khi đó tất cả các từ sẽ được phân tách ra thành các đơn vị cơ bản gọi là âm vị. Việc phân tách này làm giảm các đơn vị nhận dạng trong hệ thống. Ví dụ một hệ thống nhận dạng cho tập từ vựng khoảng 100.000 từ, nếu mỗi từ là một đơn vị thì sẽ có 100.000 mô hình *P(O|W)* với hệ thống độc lập ngữ cảnh. Nếu phân tích các từ thành âm vị trong một tập gồm 54 âm vị thì tổng số mô hình độc lập ngữ cảnh sẽ chỉ còn 54. Như vậy vừa làm giảm kích thước của hệ thống đồng thời tăng số lượng mẫu huấn luyện cho mỗi âm vị do các từ khác nhau có thể sử dụng chung một âm vị. Nếu cần bổ sung thêm từ vựng cho hệ thống cũng không cần thiết phải bổ sung dữ liệu huấn luyện vì mọi từ trong một ngôn ngữ đều được tổng hợp từ tập âm vị đã có. Như vậy về mặt lý thuyết hệ thống không bị giới hạn về số từ vựng.

Một trong những mô hình xác suất được sử dụng phổ biến cho mô hình âm học là mô hình Markov ẩn HMM (Hidden Markov Model).

❖ Tổng quan về mô hình HMM:

a) Định nghĩa HMM

HMM là mô hình xác suất dựa trên lý thuyết về chuỗi Markov [Rabiner 1989] bao gồm các đặc trưng sau:

- $O = \{o_1, o_2, \dots, o_T\}$ là tập các vector quan sát.
- $S = \{s_1, s_2, \dots, s_N\}$ là tập hữu hạn các trạng thái *s* gồm *N* phần tử.
- $A = \{a_{11}, a_{12}, \dots, a_{NN}\}$ 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 $\sum_{j=k} a_{ij} = 1, \forall i$.
- $B = \{b_{2t}, b_{it}, \dots, b_{(N-1)t}\}$ là tập các hàm xác suất phát tán 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) có dạng như công thức (1.10), trong trường hợp này ta gọi là mô hình kết hợp Hidden Markov Model và Gaussian Mixtrue Model (HMM-GMM).

$$\mathrm{b_i(o_t)} = \sum_{\mathrm{k=1}}^{\mathrm{M}} \mathrm{c_{ik}} \mathcal{N}(\mathrm{o_t}; \mu_{\mathrm{ik}}, \Sigma_{\mathrm{ik}}) \qquad (1.10)$$

Trong đó: o_t là vector quan sát tại thời điểm *t*, *M* là số thành phần trộn của hàm Gaussian, $c_{ik}, \mu_{ik}, \Sigma_{ik}$ theo thứ tự là trọng số, vector trung bình và ma trận phương sai (covariance matrix) của thành phần trộn thứ *k* của trạng thái s_i.

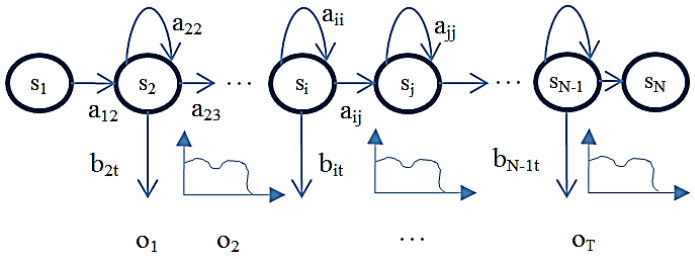
- $\Pi = \{\pi_i\}$ là tập xác suất trạng thái đầu, với $\pi_i = P(q_1 = s_i)$với *i=1..N* là xác suất để trạng thái s_i là trạng thái đầu q_1.

Như vậy một cách tổng quát một mô hình HMM λ có thể được biểu diễn bởi λ=(*A,B,*Π). Trong lĩnh vực nhận dạng thì mô hình HMM 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_i*}* có thể được xác định thông qua xác suất của từng quan sát o_i như sau: $P(O) = \prod_{i=t}^{T} P(o_i)$.

- 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 1-6 minh họa một mô hình HMM-GMM có cấu trúc dạng Left-Right liên kết không đầy đủ.



Hình 1-6: Mô hình HMM-GMM Left-Right với N trạng thái

b) Áp dụng mô hình HMM trong nhận dạng tiếng nói

Trong nhận dạng tiếng nói, mô hình HMM-GMM có thể được sử dụng để mô hình hoá 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 Hình 1-6. Quá trình xây dựng một hệ thống nhận dạng tiếng nói sử dụng mô hình HMM-GMM thông thường có hai bước như sau:

c) 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 $q_t \in S$ 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 (1.11), $P(Q|O,\lambda)$ được gọi là xác suất mô hình âm học (acoustic model).

$$P(Q|O,\lambda) = \sum_{q_t}^{Q} \pi_{tk} a_{t_{k-1}t_k} b_{tk}(o_t), \quad k = 1..N \qquad (1.11)$$

d) 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:

$$P(O,Q|\lambda) = max(P(q_1,q_2,..,q_t=i,o_1,o_2,..,o_t|\lambda))$$

1.3.3. Mô hình ngôn ngữ

Mô hình ngôn ngữ (Language Model - LM) 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. Và đây cũng là loại mô hình được sử dụng trong các thử nghiệm của luận án.

❖ **Tổng quan về mô hình n-gram:**

a) Định nghĩa

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

$$P(w_1\ w_2\ w_3 \ldots\ w_L)=P(w_1)*P(w_2|w_1)*P(w_3|w_1w_2)*\ldots*P(w_L|w_1w_2\ldots w_{L-1}) \quad (1.12)$$

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 (1.12) được viết lại như công thức (1.13). Mô hình ngôn ngữ như này gọi là mô hình ngôn ngữ n-gram.

$$P(w_1\ w_2\ w_3 \ldots\ w_L)=P(w_1)*P(w_2|w_1)*P(w_3|w_1w_2)*\ldots*P(w_L|w_{L-n+0}w_{L-n+1}\ldots w_{L-1}) \quad (1.13)$$

Công thức (1.14) được sử dụng để tính xác suất của từ w_i theo sau cụm từ w_{i-1}:

$$P(w_i|w_{i-1}) = \frac{C(w_{i-1}w_i)}{C(w_{i-1})} \qquad (1.14)$$

Trong đó:

- w_{i-1} có thể là một cụm từ có kích thước là n
- $C(w_{i-1}w_i)$, $C(w_{i-1})$ là số lần cụm "$w_{i-1}w_i$" và "w_{i-1}" xuất hiện.

b) 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 $C(w_{i-1}w_i)$, $C(w_{i-1})$ trong công thức (1.14) được xác định hoàn toàn dựa vào tập văn bản này. Như vậy việc một từ w_i hoặc cụm "$w_{i-1}w_i$" 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 và 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-1}" 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 (1.14) 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.

c) Một số phương pháp làm trơn 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:

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

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:

$$P(w_i|w_{i-n}w_{i-n+1}\dots w_{i-1}) = \begin{cases} P(w_i|w_{i-n}w_{i-n+1}\dots w_{i-1}), nếu\ C(w_{i-n}w_{i-n+1}\dots w_{i-1}w_i) > 0 \\ \alpha * P(w_{i-n}w_{i-n+1}\dots w_{i-1}), nếu\ C(w_{i-n}w_{i-n+1}\dots w_{i-1}w_i) = 0 \end{cases}$$

Trong đó α là hệ số.

- **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 có mặt w_i. Công thức tổng quát như sau:

$$P(w_i|w_{i-1}) = \varphi_1 P(w_i|w_{i-n+1}w_{i-n+2}\dots w_{i-1}) + \varphi_2 P(w_i|w_{i-n+2}w_{i-n+3}\dots w_{i-1}) + \dots + \varphi_m P(w_i|w_{i-1})$$

Trong đó $\sum_{i=1}^{m} \varphi_i = 1$.

1.3.4. Từ điển ngữ âm

Từ điển ngữ âm là tập các từ vựng, trong đó mỗi từ được phiên âm thành các âm vị cấu tạo nên từ đó. Trong các hệ thống nhận dạng tiếng nói từ vựng lớn thì mô hình âm học thường là mô hình hóa của các âm vị trong một ngôn ngữ cụ thể. Việc sử dụng từ điển âm học giúp làm giảm số lượng mô hình âm học đồng thời vẫn đảm bảo cho hệ thống có thể nhận dạng được tất cả các từ có thể có của một ngôn ngữ.

1.4. Đánh giá chất lượng hệ thống nhận dạng tiếng nói

Các hệ thống nhận dạng tiếng nói có thể được đánh giá thông qua hai tham số độ chính xác theo từ (Word Error Rate, hoặc Word Accuracy), và độ chính xác theo câu (Centence Error Rate) [[Jurafsky 2008]]. Trong đó tham số độ chính các theo từ được sử dụng phổ biến hơn cả. Trong phạm vị luận án này tham số độ chính xác theo từ (ACC) được sử dụng để đánh giá chất lượng nhận dạng cho các mô hình. Các tính ACC được mô tả ở công thức (1.15).

$$ACC = 100 - 100 \times \frac{I+S+D}{N}(\%) \quad (1.15)$$

Trong đó: I là số từ phải thêm vào; S dố từ phải thay thế; D số tử phải xóa bỏ; để tạo xâu có kích thước tương ứng với xâu tham chiếu.

1.5. Tình hình nghiên cứu hiện nay về nhận dạng tiếng nói

Trải qua nhiều giai đoạn cùng với sự phát triển của toán ứng dụng và công nghệ máy tính, nhận dạng tiếng nói nói chung đã đạt được nhiều kết quả vượt bậc trên một số các ngôn ngữ phổ biến. Hiện nay việc ứng dụng công nghệ nhận dạng tiếng nói đã được áp dụng trên nhiều lĩnh vực của cuộc sống như đã trình bày ở mục 1.2.2. Tuy nhiên do đặc tính ngẫu nhiên và bị ảnh hưởng bởi nhiều yếu tố nên hiện nay vẫn chưa thực sự có một hệ thống nhận dạng tốt như con người. Hầu hết các hệ thống được đánh giá có độ chính xác cao đều đi kèm một số điều kiện như: chỉ làm việc trong một chủ đề cụ thể; giới hạn tập từ vựng; không có nhiễu;… Ngoài ra mới chỉ các ngôn ngữ phổ biến như Anh, Pháp, Đức, Trung mới có các hệ thống có độ chính xác cao, do thu hút được nhiều tổ chức nghiên cứu trên ngôn ngữ đó và cũng do nhu cầu sử dụng các ngôn ngữ đó trên thế giới chiếm đa số. Hiện nay có rất nhiều các nhà khoa học đang đi theo cũng như đã công bố rất nhiều các nghiên cứu trên nhiều khía cạnh khác nhau để góp phần nâng cao chất lượng nhận dạng tiếng nói, để đưa ra một cái nhìn tổng quan về tình hình nghiên cứu hiện nay luận án sẽ đưa ra một số nghiên cứu mới hiện nay dựa trên bốn thành phần chính của một hệ thống nhận dạng là:

1- Trích chọn đặc trưng.
2- Mô hình âm học.
3- Mô hình ngôn ngữ.
4- Bộ giải mã.

1) Về trích chọn đặc trưng

Hai loại đặc trưng được sử dụng phổ biến trong các hệ thống nhận dạng tiếng nói phát âm liên tục từ vựng lớn hiện là Mel-frequency cepstral coefficients (MFCC) và Perceptual Linear Prediction (PLP) [Muda 2010] [Florian 2005]. Các nghiên cứu về nâng cao chất lượng đặc trưng thường là các cải tiến dựa trên hai loại đặc trưng cơ sở này. Các kỹ thuật nói chung là đi tìm một mô hình biến đổi hoặc mô hình phân lớp để chuyển hai loại đặc trưng này sang một miền không gian mới làm tăng sự khác biệt giữa hai mẫu trong hai lớp khác nhau. Một số kỹ thuật phổ biến có thể kể đến như sau:

- Phương pháp Linear Discriminant Analysis (LDA) [Haeb-Umbach 1992] [Sakai 2007] (phân tích phân biệt tuyến tính). Kỹ thuật này đi tìm một ma trận biến đổi đặc trưng đầu vào sang một đặc trưng đầu ra sao cho làm tăng mối quan hệ tuyến tính giữa các mẫu trong cùng một lớp. LDA được áp dụng phổ biến như một bước tiền xử lý đặc trưng nhằm nâng cao chất lượng và giảm số chiều cho đặc trưng đầu vào như MFCC hay PLP.
- Phương pháp Maximum Likelihood Linear Transform (MLLT) [Psutka 2007]. Phương pháp này thường được sử dụng cùng với LDA. MLLT cũng đi tìm một ma trận biến đổi đặc trưng đầu vào sang một miền không gian mới sao cho các mẫu trong cùng một lớp sẽ được mô hình hóa tốt hơn bởi các mô hình Gaussian. Tối đa hàm tương quan (Maximum Likelihood) là tham số để phân tách các lớp trong quá trình tìm ma trận chuyển đổi.
- Phương pháp tính toán đặc trưng phụ thuộc người nói (Speaker Adaptation). Kỹ thuật này nói chung là đi tìm một mô hình biến đổi riêng biệt cho từng người nói. Khi đó vector đặc trưng tương ứng với mỗi người nói sẽ được biến đổi sang một không gian mới thông qua mô hình biến đổi của người đó để tách lọc và mang nhiều thông tin hơn của người nói đó. Trong thực tế kỹ thuật này nâng cao đáng kể chất lượng nhận dạng cho hệ thống. Tuy nhiên nhược điểm của nó là chỉ làm việc tốt với những người nói đã có mô hình biến đổi, việc nhận dạng cho một người nói mới cần có dữ liệu mới để huấn luyện lại hệ thống. Trong nghiên cứu [Anastasakos 1997] tác giả đề xuất phương pháp huấn luyện để tìm các ma trận biến đổi phụ thuộc người nói cho các đặc trưng đầu vào trước khi đưa vào hàm phân bố xác suất phát tán của mô hình Markov ẩn, mặc dù phương pháp này được đề xuất khá lâu (năm 1997) tuy nhiên đến nay vẫn nhiều hệ thống áp dụng hoặc sử dụng các kỹ thuật dựa trên phương pháp này. Trong nghiên cứu [Martin 2011] nhóm tác giả đề xuất sử dụng vector đặc trưng mô tả

người nói i-vector để huấn luyện mô hình âm học, kỹ thuật này làm tăng thêm khoảng 0.8% tuyệt đối chất lượng nhận dạng.

- Một trong các phương pháp nổi lên hiện nay đó là sử dụng mạng nơron để trích chọn đặc trưng. Đây là một phương pháp mới và các kết quả nghiên cứu cho thấy nó có thể nâng cao chất lượng hệ thống. Thông thường mạng nơron được áp dụng trong bài toán phân lớp. Khi đó giá trị tại lớp đầu ra của mạng có thể chỉ ra lớp mà đặc trưng đầu vào thuộc về, hoặc xác suất mà đặc trưng đầu vào có thể thuộc về các lớp của hệ thống. Tuy nhiên phương pháp tiếp cận mới này lại sử dụng giá trị của hàm kích hoạt của một lớp ẩn trong mạng như là một giá trị đặc trưng đầu vào trực tiếp cho mô hình Markov ẩn. Cùng với sự quay trở lại của mạng nơron trong những năm gần đây, đặc biệt là kỹ thuật mạng học sâu (Deep Learning) cũng với sự phát triển mạnh mẽ của công nghệ tính toán song song dựa trên GPU (Graphical Processing Unit) đã thúc đẩy các nghiên cứu này đạt nhiều kết quả. Mốt số nghiên cứu đã công bố gần đây như [Gehring 2013] [Tuerxun 2014] [Ravanelli 2014] [K. a. Kevin 2014]. Trong các nghiên cứu này các tác giả đã sử dụng một mạng nơron nhiều lớp ẩn với các tham số được khởi tạo bằng phương pháp huấn luyện không giám sát (unsupervise traning) để tính toán đặc trưng gọi là Bottleneck (đặc trưng dạng cổ chai). Loại đặc trưng này trung bình nâng cao chất lượng với tỷ lệ khoảng 10%.

Từ các nghiên cứu gần đây cho thấy hầu hết các loại đặc trưng được sử dụng là đặc trưng ngữ âm (acoustic feature). Loại đặc trưng này thường được tính toán dựa trên phổ tín hiệu đầu vào để biểu diễn đặc tính của các âm vị trong một ngôn ngữ. Đặc trưng này rất hiệu quả với các ngôn ngữ không có thanh điệu như tiếng Anh, Đức,… Đối với các ngôn ngữ có thanh điệu, tức là thanh điệu kết hợp với các âm vị cũng tạo nên ngữ nghĩa của từ thì đặc trưng ngữ âm chưa thể hiện hết được thông tin thanh điệu này. Thanh điệu (Pitch) được tạo ra do dao động của dây thanh trong quá trình phát âm, nó thường tồn tại trong suốt khoảng thời gian phát âm của một âm tiết. Các phương pháp tính toán đặc trưng thanh điệu thường dựa trên tần số cơ bản F0 của tín hiệu tiếng nói đầu vào. Đặc trưng thanh điệu được sử dụng khá phổ biến trong tổng hợp tiếng nói, nhưng lại chưa được sử dụng phổ biến trong nhận dạng tiếng nói. Một trong những lý do đó là đặc trưng thanh điệu cần thêm một số kỹ thuật tiền xử lý trước khi được sử dụng, do thanh điệu không tồn tại trong vùng vô thanh của một phát âm. Một số nghiên cứu gần đây như [Shen 2014] [K. a. Kevin 2014] đã cho thấy việc tích hợp thêm đặc trưng thanh điệu với đặc trưng ngữ âm làm tăng chất lượng nhận dạng lên khoảng 2% tuyệt đối. Từ đó cho thấy việc nghiên cứu áp dụng đặc trưng thanh

điệu đặc biệt là cho các ngôn ngữ có thanh điệu như tiếng Việt là một hướng nghiên cứu cần thiết để nâng cao chất lượng cho hệ thống nhận dạng.

2) Về mô hình âm học (acoustic model)

Hai loại mô hình thống kê được sử dụng phổ biến trong nhận dạng tiếng nói hiện nay là: 1 – Mô hình Markov ẩn kết hợp với mô hình Gaussian (HMM-GMM); 2 – Mô hình mạng nơron (NN). Các nghiên cứu hiện nay chủ yếu thực hiện trên hai loại mô hình này hoặc lai ghép cả hai loại trong một. Các phương pháp chủ yếu tập trung vào việc tối ưu hóa quá trình ước lượng tham số cho mô hình trên một tập mẫu huấn luyện cụ thể. Có rất nhiều kỹ thuật cải tiến đã được đề xuất tập trung vào các hướng chính như ước lượng tham số phụ thuộc người nói (speaker adaptive training), ước lượng tham số để tối ưu giá trị tự tương quan giữa các mẫu trong cùng một lớp (Maximum Likelihood), tối ưu hóa tham số dựa trên đặc trưng phụ thuộc người nói (feature space adaptive training). Mô hình đa đầu vào (Multistream model, Subspace model). Một số phương pháp được sử dụng phổ biến có thể kể đến như sau:

- Trong nghiên cứu [Anastasakos 1997] tác giả đề xuất phương pháp huấn luyện các mô hình âm học mà các tham số được ước lượng tối ưu theo người nói (Speaker adaptive training-SAT). Phương pháp này dựa trên mô hình HMM-GMM. Một ma trận biến đổi (transform matrix) được tìm ra dựa trên dữ liệu và thông tin về người nói đầu vào. Sau đó đặc trưng đầu vào sẽ được biến đổi sang không gian mới thông qua ma trận này trước khi đưa vào mô hình GMM. Các đặc trưng trong miền không gian mới đã được phân lớp lại dựa trên việc tối đa mối quan hệ giữa các vector thuộc về một người nói cụ thể. Mặc dù phương pháp này đã được đề xuất từ năm 1997 nhưng cho đến nay nó vẫn được sử dụng một cách rộng rãi. Hầu hết các hệ thống nhận dạng tiên tiến trên nhiều ngôn ngữ hiện nay vẫn áp dụng phương pháp này [Shen 2014] [K. a. Kevin 2014] và thực tế cho thấy nó giúp nâng cao đáng kể chất lượng nhận dạng của hệ thống.
- Nghiên cứu [Ochiai 2014] các tác giả đã đề xuất một phương pháp mới sử dụng mô hình mạng nơron học sâu làm mô hình âm học (Deep Neural Network Speaker Adaptation), tuy nhiên lớp ẩn ở giữa của mô hình này được huấn luyện lại cho từng người nói. Sau đó với mỗi người nói cụ thể mô hình phụ thuộc người nói sẽ là các lớp khác của mạng kết hợp với lớp ẩn ở giữa đã được huấn luyện cho người này. Kết quả cho thấy mô hình mới tăng với tỷ lệ khoảng 8.4% so với mô hình độc lập người nói.
- Nghiên cứu [P. a. Daniel 2011] đề xuất một phương pháp huấn luyện mô hình âm học trong trường hợp dữ liệu huấn luyện bị hạn chế. Đối với các mô hình

xác suất thì dữ liệu là một nhân tố quan trọng trong việc ước lượng tham số mô hình âm học trong quá trình huấn luyện, việc thiếu dữ liệu có thể dẫn đến mô hình chỉ nhận được các tham số khởi tạo ngẫu nhiên hoặc không mô tả được tất cả các trường hợp có thể có của mẫu đầu vào. Trong thực tế đối với một số ngôn ngữ mới được bắt đầu nghiên cứu thì thường rất hạn chế về dữ liệu, ngay cả với các ngôn ngữ đã được nghiên cứu nhiều năm thì cũng xảy ra các trường hợp đặc biệt mà có ít dữ liệu như xuất hiện người nói mới cho hệ thống, hoặc hệ thống phải làm việc với một ngữ cảnh mới, môi trường mới. Mô hình mà nghiên cứu này đề xuất có thể giải quyết được vấn đề này. Ý tưởng chính của phương pháp là tất cả các mô hình Gaussian của các đơn vị nhận dạng trong hệ thống sẽ cùng chia sẻ một mô hình Gaussian khác, mô hình này gọi là mô hình Gaussian con (Subspace Gaussian - SGMM) trong đó các tham số của nó được xác định thông qua tất cả các tham số của các mô hình của các đơn vị nhận dạng trong hệ thống. Các thử nghiệm của tác giả đã cho thấy trung bình nó nâng chất lượng nhận dạng lên với tỷ lệ 9.7%.

- Nghiên cứu [Tokuda 1999] đề xuất một loại mô hình Markov ẩn mới có khả năng mô hình hóa loại đặc trưng chứa cả số và ký hiệu. Mô hình này được đặt tên là mô hình Markov ẩn phân bố xác suất đa không gian (Multi-space Probability Distribution Hidden Markov Model MSD-HMM), ngay khi ra đời tác giả đã áp dụng nó cho tổng hợp tiếng nói. Tác giả sử dụng mô hình này để mô hình hóa một dạng đặc trưng với 2 luồng riêng biệt, một là đặc trưng ngữ âm chứa giá trị số thực, luồng còn lại chứa thông tin về ngữ điệu (Pitch). Điều đặc biệt là đặc trưng ngữ điệu có thể chứa cả số thực và ký hiệu. Phương pháp này sau đó được áp dụng chủ yếu trong lĩnh vực tổng hợp tiếng nói [Yu 2010] [Kunikoshi 2011] và nhận dạng người nói [Miyajima 2001]. Mặc dù đây có thể tạm coi là một giải pháp khả thi đối với các ngôn ngữ có thanh điệu vì mô hình này có khả năng mô hình chính xác đặc tính gián đoạn của đặc trưng thanh điệu, nhưng tính đến nay có rất ít nghiên cứu áp dụng mô hình cho nhận dạng tiếng nói. MSD-HMM mới chỉ được áp dụng cho tiếng Quan thoại của Trung quốc [Y. a. Qian 2009] [Chong-Jia 2011].

Nhìn qua một số kết quả nghiên cứu gần đây cho thấy hầu hết các nghiên cứu mới chỉ tập trung vào một số ngôn ngữ phổ biến. Hầu hết các ngôn ngữ này là ngôn ngữ không có thanh điệu, vì thế đặc trưng thanh điệu hoặc là bị bỏ qua hoặc là chỉ được sử dụng như một yếu tố làm gia tăng chất lượng nhận dạng. Các đoạn đứt gãy của đặc trưng thanh điệu được bù bởi một giá trị ngẫu nhiên thông qua các thuật toán làm trơn hoặc tương quan chéo. Duy nhất có nghiên cứu của tác giả Tokuda [Tokuda

1999] là đề cập đến việc mô hình hoá đặc tính đứt gãy này. Tuy nhiên mô hình này chưa được nghiên cứu một cách rộng rãi trong nhận dạng tiếng nói cho các ngôn ngữ khác.

3) Về mô hình ngôn ngữ

Hiện nay các phương pháp xây dựng mô hình ngôn ngữ (Language Model) thường dựa trên 2 kỹ thuật chính là mô hình n-gram và mạng nơron. Các phương pháp dựa trên n-gram đã được phát triển từ rất sớm và ngày nay vẫn được áp dụng phổ biến do tính đơn giản của mô hình. Nhược điểm chính của mô hình là không xác định được xác suất của các chuỗi từ hoặc các từ mà nó không xuất hiện trong dữ liệu. Đã có rất nhiều các nghiên cứu [R. K. Ney 1995] [Stolcke 1998] [Katz 1987] [Frederick 1980] [Good 1953] nhằm khắc phục nhược điểm này gọi chung là phương làm trơn mô hình (Smoothing). Một số phương pháp được sử dụng phổ biến như:

- Phương pháp cộng thêm 1 (add-one smoothing).
- Phương pháp lùi (back-off smoothing).
- Phương pháp nội suy (interpolation smoothing).
- Phương pháp Kneser-Ney (Kneser-Ney smoothing).

Loại mô hình ngôn ngữ thứ hai dựa trên mô hình mạng nơron. Loại mô hình này thường tốt hơn mô hình n-gram vì tận dụng được khả năng phân lớp của mạng. Tuy nhiên thông thường để huấn luyện loại mô hình này cần nhiều dữ liệu và tốn bộ nhớ hơn. Trong những năm gần đây loại mô hình này được nhiều tác giả nghiên cứu phát triển với nhiều cải tiến mới. Như nghiên cứu [Bengio 2003] [Schwenk 2007] trình bày phương pháp sử dụng mạng học sâu (Deep Learning) để làm mô hình ngôn ngữ. Trong nghiên cứu này nhóm tác giả đã làm nhiều thử nghiệm cho thấy mô hình ngôn ngữ sử dụng mạng nơron học sâu cho kết quả tốt hơn mô hình n-gram trung bình với tỷ lệ khoảng 1%.

4) Về bộ giải mã

Các bộ giải mã trong các hệ thống nhận dạng tiếng nói hiện nay chủ yếu dựa trên thuật toán tìm kiếm Viterbi, bản chất là đi tìm một đường dẫn tối ưu từ một đồ thị mà các đỉnh là đơn vị nhận dạng của hệ thống và trọng số đường đi hay xác suất chuyển giữa các đỉnh tính toán được từ mô hình ngôn ngữ và mô hình âm học. Một số nghiên cứu gần đây chỉ đưa ra các kỹ thuật mới để tăng tốc độ tìm kiếm hay là giảm dung lượng bộ nhớ. Một phương pháp tiêu biểu có thể chỉ ra đó là phương pháp sử

dụng bộ biến đổi trạng thái hữu hạn (Finite-State Transducer - FST) [Dixon 2012]. Ý tưởng của phương pháp là tích hợp và biểu diễn mô hình ngôn ngữ, mô hình âm học, từ điển vào một mô hình biến đổi trạng thái duy nhất. Như vậy khi giải mã từ một đầu vào thông qua mô hình FST ta có thể tìm ra đường đi tốt nhất mà không cần phải tính toán lại trên mô hình ngôn ngữ, mô hình âm học. Phương pháp này làm giảm tối thiểu thời gian giải mã cho hệ thống nhận dạng, rất hiệu quả cho các hệ thống nhận dạng online.

1.6. Nhận dạng tiếng Việt và các nghiên cứu hiện nay

Nhìn chung tính đến hiện nay các nghiên cứu về nhận dạng tiếng Việt vẫn còn rất hạn chế. Phần lớn các nghiên cứu mới chỉ dừng lại ở nhận dạng số hoặc nhận dạng các từ phát âm rời rạc. Tiếng Việt là một ngôn ngữ có thanh điệu, như vậy một hệ thống nhận dạng đầy đủ sẽ phải bao gồm 2 thành phần là nhận dạng âm vị và nhận dạng thanh điệu. Đã có một số nghiên cứu về nhận dạng thanh điệu cho tiếng Việt, tuy nhiên các nghiên cứu này mới chủ yếu tập trung vào việc phân tích đặc tính và tìm ra mô hình phù hợp trong việc mô hình hóa và nhận dạng thanh điệu đơn lẻ. Hầu hết chưa tích hợp việc nhận dạng thanh điệu với nhận dạng âm vị để tạo thành một hệ thống hoàn chỉnh.

- ***Một số nghiên cứu về nhận dạng tiếng nói cho chữ số và các từ phát âm rời rạc***: Các nghiên cứu chỉ thực hiện trên tiếng nói phát âm rời rạc, tức khoảng trễ giữa hai từ liền nhau lớn. Số từ vựng chỉ là 10 trong trường hợp nhận dạng số, hoặc nhỏ hơn 200.
  - Nghiên cứu của tác giả Đặng Ngọc Đức [Đức 2003] đã đề xuất một số phương pháp gán nhãn cho dữ liệu tiếng Việt phát âm liên tục. Đồng thời đã đề xuất sử dụng mô hình lai ghép giữa mạng nơron và mô hình Markov ẩn cho nhận dạng 10 chữ số tiếng Việt trên dữ liệu thu âm qua điện thoại với chất lượng nhận dạng đạt 97.46% mức từ. Trong nghiên cứu này tác giả đã sử dụng tiếng nói phát âm liên tục để thử nghiệm. Tuy nhiên tác giả mới chỉ tập trung vào giải quyết vấn đề gán nhãn dữ liệu tự động và sử dụng mô hình lai ghép để mô hình hóa bộ đơn vị trong bài toán nhận dạng số. Kết quả nghiên cứu cho thấy việc sử dụng mạng nơron lai ghép với mô hình HMM cho kết quả tốt hơn mô hình HMM truyền thống.
  - Nghiên cứu của nhóm tác giả Bạch Hưng Khang trong đề tài cấp nhà nước năm 2004 [Khang 2004]. Trong nghiên cứu này các tác giả đã nghiên cứu và phân tích chi tiết về đặc điểm và đặc tính của tiếng Việt như các đặc trưng âm vị và âm học, thanh điệu. Nghiên cứu cũng trình bày các phương pháp trích chọn đặc

trưng, phân tích ảnh hưởng của nhiễu. Hai loại mô hình được sử dụng và so sánh là mạng nơron và Markov ẩn. Phạm vi của nghiên cứu mới chỉ áp dụng cho tiếng nói rời rạc với 193 âm tiết. Các câu phát âm có nội dung hạn chế cho bài toán điều khiển một số chức năng của một số thiết bị điện tử, tin học.

- ***Một số nghiên cứu về nhận dạng thanh điệu***: Các nghiên cứu này chỉ tập trung vào việc nhận dạng thanh điệu trong mỗi từ phát âm, tức đầu ra của hệ thống nhận dạng là một trong sáu thanh điệu của tiếng Việt. Các nghiên cứu về vấn đề này mặc dù đã áp dụng trên tiếng nói liên tục nhưng vẫn sử dụng các mô hình truyền thống như HMM hoặc NN và đặc trưng thanh điệu được bổ sung các giá trị "nhân tạo" tại các vùng vô thanh nơi mà nó không tồn tại.
  - Nghiên cứu của tác giả Nguyễn Quốc Cường [Quoc Cuong 2001] đã trình bày việc sử dụng tần số cơ bản F0 để làm đặc trưng cho thanh điệu tiếng Việt, sau đó mô hình hóa bởi mô hình Markov ẩn để nhận dạng thanh điệu. Một dạng vector đặc trưng cho thanh điệu dựa trên tổng và hiệu của F0 và giá trị năng lượng giữa hai khung tín hiệu liền kề được đề xuất. Từ kết quả đó tác giả đã xây dựng một hệ thống nhận dạng tiếng Việt có tích hợp nhận dạng thanh điệu cho các từ phát âm rời rạc với độ chính xác khoảng 94%.
  - Nghiên cứu của nhóm tác giả Nguyễn Hồng Quang [Hong Quang 2008] được thực hiện ở Pháp. Đây cũng là một nghiên cứu về nhận dạng thanh điệu tiếng Việt nhưng theo hướng tiếp cận trên tiếng nói phát âm liên tục. Trong nghiên của này tác giả đã đề xuất một loại đặc trưng cùng với phương pháp chuẩn hóa nó dựa trên tần số cơ bản F0 và giá trị năng lượng của tín hiệu tiếng nói. Các kết quả nghiên cứu đã được thử nghiệm trên một tập dữ liệu phát âm liên tục có kích thước trung bình. Kết quả nhận dạng thanh điệu đạt 81.02%.
  - Nghiên cứu của tác giả Vũ Tất Thắng [Thang 2008]. Trong nghiên cứu này nhóm tác giả đã đề xuất phương pháp nhận dạng thanh điệu cho tiếng Việt sử dụng mạng nơron. Nghiên cứu cũng đề xuất loại đặc trưng và phương pháp chuẩn hóa phù hợp cho mô hình nhận dạng. Kết quả nghiên cứu được thử nghiệm trên bộ dữ liệu thu âm từ chương trình phát thanh của Việt Nam. Chất lượng nhận dạng 6 thanh điệu trung bình là 83.83% phụ thuộc người nói, cao hơn khoảng 2% so với hệ thống sử dụng mô hình Markov ẩn.
- ***Một số nghiên cứu gần đây về nhận dạng tiếng Việt phát âm liên tục từ vựng lớn:*** Các nghiên cứu này các tác giả đã đề xuất các mô hình cho nhận dạng tiếng Việt phát âm liên tục từ vựng lớn. Tuy nhiên tất cả các nghiên cứu mới chỉ áp dụng mô hình truyền thống HMM và NN trên đặc trưng thanh điệu đã chỉnh sửa.
  - Một trong những nghiên cứu đầu tiên về nhận dạng tiếng Việt từ vựng lớn phát âm liên tục là của nhóm tác giả Vũ Tất Thắng [T. T. Vu 2005] được thực hiện

tại Nhật bản. Trong nghiên cứu này tác giả trình bày cấu trúc cơ bản của tiếng việt và đề xuất thử nghiệm một số tập âm vị có chứa và không chứa thanh điệu. Các thử nghiệm được thực hiện trên dữ liệu thu âm từ đài phát thanh Việt Nam sử dụng 2 loại đặc trưng MFCC và PLP, mô hình nhận dạng là Markov ẩn. Kết quả nhận dạng đạt 82.97%. Mặc dù trong nghiên cứu này tác giả chưa sử dụng đặc trưng thanh điệu, nhưng bằng việc mô hình hóa thanh điệu sử dụng bộ âm vị có thanh điệu đã cho kết quả tối ưu hơn mô hình âm vị không có thanh điệu. Từ kết quả này đã cho thấy thanh điệu là một nhân tố góp phần làm tăng chất lượng nhận dạng tiếng Việt cũng tương tự như tiếng Mandarin, Cantonese.

- Nghiên cứu của nhóm tác giả Vũ Ngọc Thắng [N. T. Vu 2009] được thực hiện tại Đức. Đây là một trong số nghiên cứu đầu tiên về nhận dạng tiếng Việt phát âm liên tục từ vựng lớn có tích hợp cả mô hình thanh điệu. Tác giả đã trình bày một cách tiếp cận mới để khởi tạo việc huấn luyện các mô hình âm học cho tiếng Việt bằng cách kế thừa mô hình âm học từ các âm vị tương đương của các ngôn ngữ khác. Tác giả đề xuất việc mô hình hóa thanh điệu tương tự như nghiên cứu [T. T. Vu 2005] tức là bổ sung thêm các ký hiệu thanh điệu vào các ký hiệu âm vị trong tập âm vị của hệ thống. Trong nghiên cứu này tác giả cũng đã đưa ra cách tiếp cận tổng hợp 2 loại dữ liệu ngữ âm (acoustic) và dữ liệu thanh điệu (pitch) vào một để làm đầu vào cho mô hình HMM. Trong nghiên cứu này tác giả còn đề xuất phương pháp cải tiến mô hình ngôn ngữ bằng việc thu thập thêm dữ liệu văn bản từ các website tiếng Việt. Kết quả thử nghiệm đạt sai số nhận dạng theo từ là 11%.
- Nghiên cứu của nhóm tác giả Nguyễn Tuân [Tuan 2009] đề xuất bộ âm vị kết hợp giữa các âm vị đơn, nguyên âm để huấn luyện mô hình âm học cho nhận dạng tiếng Việt liên tục từ vựng lớn. Kết quả thử nghiệm đạt độ chính xác 86.06% trên bộ dữ liệu kích thước 27 giờ cho huấn luyện và 1 giờ cho thử nghiệm. Nghiên cứu nhóm tác giả tập trung vào vấn đề tối ưu bộ âm vị dựa trên việc ghép nối các đơn vị ngữ âm cơ bản trong âm tiết tiếng Việt. Đặc trưng và mô hình vẫn là MFCC và HMM truyền thống.
- Các nghiên cứu của Viện nghiên cứu quốc tế MICA thuộc Đại học Bách khoa Hà nội đã đề xuất giải pháp kế thừa các mô hình âm vị của các ngôn ngữ khác như tiếng Anh, Pháp để huấn luyện các mô hình âm vị cho nhận dạng tiếng Việt [Sethserey 2010], đề xuất các thư viện để xây dựng các hệ thống nhận dạng tiếng nói cho tiếng Việt dựa trên công cụ YAST [Ferreira 2012]. Ở nghiên cứu này nhóm tác giả kế thừa hoàn toàn các mô hình của các âm vị tương đương đã được huấn luyện trong các ngôn ngữ Anh, Pháp để xây dựng bộ mô hình âm vị cho tiếng Việt. Cách tiếp cận này loại bỏ được khó khăn về việc xây dựng bộ cơ

sở dữ liệu huấn luyện đủ tốt, tuy nhiên do sử dụng các mô hình âm vị tương đồng từ các ngôn ngữ Anh, Pháp là các ngôn ngữ không có thanh điệu dẫn đến mô hình âm học trong trường hợp này sẽ không có khả năng nhận dạng thanh điệu. Việc nhận dạng thanh điệu sẽ phụ thuộc hoàn toàn vào mô hình ngôn ngữ.

- Nghiên cứu gần đây trong luận án tiến sĩ tại Đức của tác giác Vũ Ngọc Thắng [Thắng 2014] trình bày một hướng tiếp cận để xây dựng các hệ thống nhận dạng cho những ngôn ngữ hạn chế về dữ liệu huấn luyện. Tiếng Việt là một trong các ngôn ngữ được thử nghiệm trong nghiên cứu này. Tác giả đề xuất việc sử dụng chung một tập âm vị cho các ngôn ngữ thử nghiệm. Bằng việc kế thừa dữ liệu hoặc mô hình đã huấn luyện cho các âm vị này để khởi tạo mô hình cho một ngôn ngữ mới. Trong nghiên cứu này tác giả cũng đề xuất sử dụng mạng nơron học sâu (deep learning) là công nghệ tiên tiến đang được nhiều nghiên cứu áp dụng hiện nay để trích chọn đặc trưng.
- Nghiên cứu của tác giả Nguyen Thien Chuong [Chuong 2014] trong luận án tiến sỹ tại Cộng hòa Czech nghiên cứu về việc tối ưu tập âm vị cho nhận dạng tiếng Việt. Tác giả đề xuất và thử nghiệm các bộ âm vị khác nhau thông qua việc kết hợp âm đầu, âm cuối, âm đơn, âm đôi với các tổ hợp khác nhau để tìm ra bộ âm vị cho kết quả tốt nhất trên tập dữ liệu thử nghiệm.

Như vậy một các tổng thể có thể thấy, hầu hết các nghiên cứu nhận dạng tiếng Việt mới tập trung vào việc nhận dạng chữ số và các từ phát âm rời rạc. Các nghiên cứu về tiếng nói phát âm liên tục trên bộ từ vựng lớn còn rất hạn chế. Hầu hết các nghiên cứu cho nhận dạng tiếng Việt đã công bố cho đến nay mới chỉ sử dụng mô hình HMM, DNN hoặc mô hình lai ghép. Các mô hình này sử dụng các đặc trưng đầu vào là đặc trưng ngữ âm hoặc đặc trưng thanh điệu ở dạng liên tục. Chưa có một công bố nào đề cập việc mô hình hóa thanh điệu tiếng việt cũng với sự không liên tục của loại tín hiệu này.

1.7. Một số nghiên cứu gần đây trên các ngôn ngữ có thanh điệu

- Tiếng Mandarin (tiếng Quan thoại) và Cantonese (tiếng Quảng Đông) của Trung Quốc: Đây là 2 ngôn ngữ có thanh điệu được sử dụng phổ biến trên thế giới. Trong đó tiếng Mandarin có 5 thanh điệu bao gồm cả thanh bằng, tiếng Cantonese tổng quát có 6 thanh điệu (nếu xét cả đến sự biến thiên của 3 thanh cao, thanh bằng và thanh thấp trong các âm tiết chứa các phụ âm dừng thì Cantonese có 9 thanh điệu). Các nghiên cứu về nhận dạng tiếng nói có thanh điệu trên hai ngôn ngữ này đã được nhiều tác giả người bản địa thực hiện với

nhiều cách tiếp cận khác nhau. Một số nghiên cứu tiêu biểu có thể chỉ ra như sau:

- Trong nghiên cứu [Chen 2001] nhóm tác giả đề xuất mô hình thanh điệu cho các ngôn ngữ như Mandarin, Cantonese bằng cách kết hợp thông tin thanh điệu với phần nguyên âm chính và sử dụng nó như một âm vị có thanh điệu. Nhóm tác giả có đưa ra các cách kết hợp khác nhau giữa thông tin thanh điệu với các thành phần cấu tạo nên một âm tiết để thu được các bộ âm vị khác nhau. Qua các thử nghiệm và phân tích nhóm tác giả chỉ ra rằng cách kết hợp thanh điệu với âm chính vừa làm giảm kích thước tập âm vị vừa thu được chất lượng nhận dạng tốt hơn so với các phương pháp khác. Từ kết quả nghiên cứu này cho thấy việc bổ sung thông tin thanh điệu cho tập âm vị là một phương pháp quan trọng trong việc tối ưu mô hình nhận dạng cho tiếng Mandarin và Cantonese.
- Trong các nghiên cứu [Wang 2006] [Y. a. Qian 2009] nhóm tác giả đề xuất sử dụng mô hình phân bố đa không gian MSD-HMM áp dụng cho nhận dạng tiếng Mandarin. Mô hình này được sử dụng để mô hình hóa đặc trưng đầu vào là tổ hợp của MFCC và Pitch. Đây là nghiên cứu đầu tiên áp dụng mô hình MSD-HMM cho nhận dạng tiếng nói. Trong nghiên cứu này các âm vị của tiếng Mandarin được mô hình hóa bởi mô hình MSD-HMM, đây là loại mô hình có khả năng mô hình hóa đặc tính đứt gẫy của đường đặc trưng thanh điệu. Kết quả thử nghiệm cho thấy mô hình MSD-HMM cho kết quả tốt hơn mô hình HMM truyền thống, và việc sử dụng đặc trưng thanh điệu đứt gãy theo đúng bản chất vật lý đã cho kết quả tốt hơn đặc trưng thanh điệu đã chỉnh sửa trên ngôn ngữ Mandarin.

- Tiếng Thái Lan (5 thanh điệu): Các nghiên cứu về nhận dạng tiếng nói cho tiếng Thái cũng tương tự như tiếng Việt còn rất hạn chế, hầu hết các nghiên cứu tập trung vào vấn đề phân đoạn từ trong các câu phát âm tiếng Thái, do các từ trong tiếng Thái có thể được viết liền nhau. Có rất ít các nghiên cứu về nhận dạng tiếng Thái có thanh điệu, một nghiên cứu gần đây có thể kể đến như sau:
  - Trong nghiên cứu [Sinaporn 2005] nhóm tác giả trình bày quy trình xây dựng hệ thống nhận dạng tiếng Thái sử dụng mô hình HMM. Trong nghiên cứu này nhóm tác giả đã xây dựng mô hình thanh điệu bằng cách sử dụng đặc trưng thanh điệu kết hợp với đặc trưng MFCC làm đặc trưng đầu vào. Thông tin về thanh điệu không được tích hợp vào bộ âm vị mà được sử dụng làm thông tin phân lớp của hệ thống. Kết quả thử nghiệm trên bộ dữ liệu có kích thước khoảng 26 giờ đạt độ chính xác theo từ là

khoảng 84%. Cách tiếp cận này có ưu điểm là giảm kích thước bộ âm vị nhưng vẫn tạo ra sai số gán thanh điệu vào âm vị trong quá trình phân lớp. Tuy nhiên, từ kết quả này cho thấy việc bổ sung thông tin thanh điệu cũng như tiếng Mandarin hay Cantonese đối với tiếng Thái đã mang lại kết quả tối ưu hơn cho mô hình nhận dạng.

1.8. Kết luận, các nội dung và phạm vi nghiên cứu chính của luận án

Qua các phân tích tổng quan về tình hình nghiên cứu ở trên cho thấy các nghiên cứu trên các ngôn ngữ có thanh điệu như tiếng Việt vẫn còn hạn chế. Một số vấn đề cấp thiết đối với nhận dạng tiếng Việt có thể chỉ ra như sau:

1) Các nghiên cứu về tiếng Việt với tập từ vựng lớn phát âm liên tục còn rất hạn chế. Chưa có một nghiên cứu nào tập trung vào việc mô hình hóa, phân tích và đánh giá ảnh hưởng của thanh điệu trong hệ thống nhận dạng tiếng Việt từ vựng lớn phát âm liên tục. Từ các kết quả nghiên cứu đã công bố trên các ngôn ngữ Mandarin, Cantonese, Thái cho thấy việc mô hình hóa thanh điệu hoặc sử dụng thông tin thanh điệu để xây dựng hệ thống nhận dạng đều đã làm tăng chất lượng của hệ thống. Tuy nhiên với tiếng Việt các nghiên cứu mới chỉ dừng lại ở việc sử dụng các mô hình truyền thống như HMM hay NN với đặc trưng thanh điệu đã được chỉnh sửa làm đầu vào. Các tiếp cận này mặc dù đã sử dụng đến thông tin thanh điệu nhưng mới ở mức đơn giản đó là sử dụng bộ âm vị có thanh điệu, hoặc sử dụng đặc trưng thanh diệu đã chỉnh sửa. Lý do là các nghiên cứu đã tập trung vào giải quyết các vấn đề khác như tính toán đặc trưng, xây dựng dữ liệu, kế thừa tài nguyên từ các ngôn ngữ khác, xây dựng mô hình ngôn ngữ, áp dụng mô hình HMM, NN hoặc mô hình lai ghép HMM-NN,…
2) Chưa có một nghiên cứu nào tập trung vào việc nghiên cứu phương pháp mô hình hóa đúng bản chất đứt gãy của đặc trưng thanh điệu cho tiếng Việt. Trong khi vấn đề này đã được nghiên cứu thành công cho tiếng Mandarin bằng cách sử dụng mô hình MSD-HMM. Đặc trưng thanh điệu trong các nghiên cứu đã công bố cho tiếng Việt thường được áp dụng các kỹ thuật làm trơn để bổ sung các giá trị "nhân tạo" cho những đoạn bị đứt gãy trên miền vô thanh và sau đó được mô hình hóa bằng các mô hình HMM hoặc NN như một loại đặc trưng liên tục kết hợp với đặc trưng ngữ âm. Như vậy cần có nghiên cứu để đánh giá và so sánh phương pháp sử dụng đặc trưng thanh điệu đã làm trơn và đặc trưng thanh điệu thô theo đúng bản chất của nó. Chưa có một nghiên cứu nào đưa ra các phương pháp tăng cường chất lượng cho cả đặc trưng ngữ âm và đặc trưng thanh điệu dựa theo đặc tính của tiếng Việt.

3) Tính đến hiện nay mới chỉ có mô hình MSD-HMM là mô hình hóa đặc trưng thanh điệu đúng theo bản chất vật lý của nó và hiện trong lĩnh vực nhận dạng tiếng nói MSD-HMM mới chỉ được nghiên cứu áp dụng thành công cho tiếng Mandarin, vẫn chưa có nghiên cứu nào nghiên cứu áp dụng mô hình này cho tiếng Việt.
4) Một trong những xu thế về học máy gần đây đó là việc ứng dụng mạng nơron, đặc biệt là mạng nơron học sâu (deep learning). Mạng nơron có thể được sử dụng làm mô hình âm học, mô hình ngôn ngữ, hoặc kết hợp với HMM làm mô hình lai ghép. Tuy nhiên có một cách tiếp cận mới đã được nghiên cứu áp dụng thành công cho tiếng Anh, Đức,.. là sử dụng mạng NN làm bộ biến đổi tăng cường chất lượng đặc trưng đầu vào. Vì thế cần có thêm các nghiên cứu với cách tiếp cận này để đánh giá hiệu quả của NN trong việc tăng cường đặc trưng cho tiếng Việt.

❖ ***Từ các vấn đề thực tế trên dẫn đến luận án sẽ tập trung nghiên cứu một số nội dung chính như sau:***

1) Nghiên cứu mô hình thanh điệu cho nhận dạng tiếng Việt phát âm liên tục từ vựng lớn dựa trên bộ âm vị có thông tin thanh điệu.
2) Nghiên cứu áp dụng mô hình MSD-HMM cho nhận dạng tiếng Việt phát âm liên tục từ vựng lớn. Thanh điệu trong tiếng Việt ảnh hưởng đến ngữ nghĩa của từng phát âm. Vì thế mô hình hóa thanh điệu cần mô tả đúng đặc tính vật lý của nó đó là đặc trưng thanh điệu không liên tục theo thời gian. Luận án sẽ tập trung vào việc nghiên cứu để có thể áp dụng và tìm ra mô hình MSD-HMM phù hợp cho tiếng Việt. Tìm ra phương pháp trích chọn đặc trưng thanh điệu tương thích với mô hình này. So sánh và đánh giá chất lượng của mô hình MSD-HMM với mô hình HMM.
3) Nghiên cứu áp dụng mạng nơron vào việc trích chọn và nâng cao chất lượng đặc trưng đầu vào cho nhận dạng tiếng Việt. Ứng dụng truyền thông của mạng nơron là dùng để phân lớp. Trong nhận dạng tiếng nói nó được sử dụng như là mô hình âm học hoặc có thể kết hợp với HMM để tạo ra mô hình lai ghép. Hiện nay có một cách tiếp cận mới là sử dụng mạng nơron để tính toán đặc trưng. Các nghiên cứu đã công bố cho tiếng Anh, Đức cho thấy loại đặc trưng này làm tăng chất lượng nhận dạng của hệ thống. Luận án sẽ tiến hành nghiên cứu và áp dụng phương pháp này để tăng cường đặc trưng ngữ âm và đặc trưng thanh điệu cho tiếng Việt.
4) Nghiên cứu mô hình tích hợp đặc trưng đã được tăng cường bằng mạng nơron và MSD-HMM cho nhận dạng tiếng Việt.

❖ ***Phạm vi nghiên cứu của luận án***

1) Đối tượng nghiên cứu của luận án là tiếng nói phát âm liên tục, tức là tiếng nói được phát âm một cách tự nhiên và không có bất cứ điều kiện nào về khoảng cách giữa hai âm tiết liên tục.

2) Kích thước từ vựng là không giới hạn (từ vựng lớn), nghĩa là hệ thống nhận dạng dựa trên các mô hình của luận án có khả năng nhận dạng tất cả các từ có thể có tiếng Việt.
3) Do hạn chế khách quan là dữ liệu huấn luyện và thử nghiệm được thu bởi các giọng miền Bắc vì thế các thử nghiệm và kết quả trong luận án mới chỉ thực hiện cho giọng Bắc.

Chương 2: Mô hình thanh điệu cho nhận dạng tiếng Việt từ vựng lớn phát âm liên tục

2.1. Tóm tắt chương

Trong điều kiện từ vựng nhỏ thì các hệ thống nhận dạng có thể xây dựng bộ đơn vị nhận dạng một cách tùy biến như: chọn âm tiết làm vị, chọn âm đầu và phần vần làm âm vị,… Tuy nhiên cách chọn này hoặc do số từ vựng nhỏ nên bộ đơn vị này có thể không đủ để cấu thành nên tất cả các âm tiết của ngôn ngữ nhận dạng. Chính vì vậy các hệ thống này thường là không có khả năng nhận dạng một từ nếu nó không có trong dữ liệu huấn luyện. Trong trường hợp đó hệ thống phải được huấn luyện lại. Để một hệ thống nhận dạng có khả năng nhận dạng tất cả các từ có thể có của một ngôn ngữ (từ vựng lớn) thì bộ đơn vị của hệ thống phải đủ để có thể cấu thành nên tất cả các âm tiết có thể có của ngôn ngữ đó. Với mục tiêu xây dựng mô hình nhận dạng cho tiếng Việt từ vựng lớn phát âm liên tục thì trong chương này luận án sẽ trình bày đề xuất mô hình có khả năng nhận dạng tất cả các âm tiết của tiếng Việt bằng cách sử dụng bộ âm vị của tiếng Việt như là bộ đơn vị nhận dạng, trong điều kiện tiếng nói đầu vào là tiếng nói phát âm liên tục một cách tự nhiên.

Nội dung chính của chương bao gồm: Trình bày phương pháp xây dựng mô hình nhận dạng tiếng Việt phát âm liên tục từ vựng lớn có thanh điệu; Trình bày đề xuất về thuật toán tạo từ điển ngữ âm tự động cho tiếng Việt; Giới thiệu tổng quan về dữ liệu thử nghiệm và các công cụ được sử dụng trong luận án; Kết quả thử nghiệm và các so sánh đánh giá giữa mô hình có thanh điệu và không có thanh điệu cho tiếng Việt.

2.2. Tổng quan về tiếng Việt

Tiếng Việt là một ngôn ngữ đơn âm tiết (Monosyllable), nghĩa là mỗi một âm tiết được thể hiện bởi một từ và cũng là đơn vị cơ bản trong phát âm. Các đặc tính chính của âm tiết tiếng Việt [Chừ 1997] như sau:

a) Âm tiết tiếng Việt có tính độc lập cao

Âm tiết là đơn vị cơ bản trong hệ thống các đơn vị ngôn ngữ. Mỗi âm tiết đều có khả năng được thể hiện bởi một từ không biến hình, hay nói cách khác một âm tiết cũng đồng thời là một hình vị. Về mặt ý nghĩa và ngữ pháp trong tiếng Việt được thể hiện chủ yếu bằng trật tự giữa các từ. Như vậy tiếng Việt khác với một số ngôn ngữ khác như tiếng Anh, Pháp,… là luôn có ranh giới rõ ràng giữa hai âm tiết.

b) Âm tiết tiếng Việt có khả năng biểu hiện ý nghĩa

Hầu hết các âm tiết tiếng Việt khi đứng một mình đều có khả năng biểu hiện một ý nghĩa xác định. Như vậy âm tiết tiếng Việt ngoài vai trò là một đơn vị ngữ âm nó còn có vai trò về từ vựng và ngữ pháp.

c) Âm tiết tiếng Việt có cấu trúc chặt chẽ

Tất cả các âm tiết tiếng Việt đều có thể phân tích thành một cấu trúc gồm năm thành phần như Bảng 2-1:

Bảng 2-1: Cấu trúc âm tiết tiếng Việt

Thanh điệu			
Âm đầu	Vần		
	Âm đệm	Âm chính	Âm cuối

Ví dụ cấu trúc của âm tiết (từ) "chuyển" có thể được phân tích thành 5 thành phần như sau:

Bảng 2-2: Ví dụ cấu trúc ngữ âm của âm tiết "chuyển"

Thanh điệu (Thanh hỏi)			
Âm đầu (Ch)	Vần (uyên)		
	Âm đệm (u)	Âm chính (yê)	Âm cuối (n)

2.2.1. Âm vị tiếng Việt

Bảng 2-3: Tập âm vị ngữ âm tiếng Việt

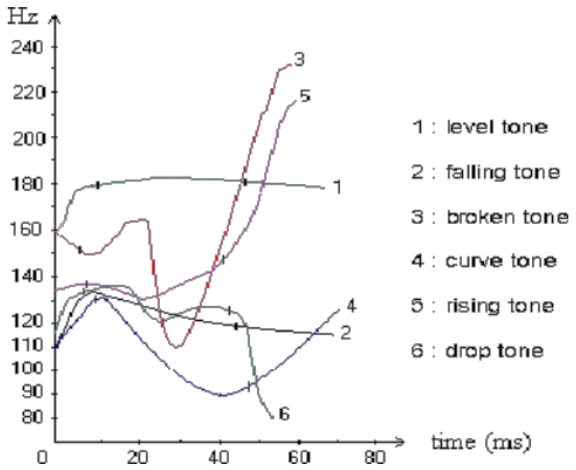
Âm đầu				**Âm đệm**		**Âm chính**				**Âm cuối**	
IPA	Cách Viết	IPA	Cách Viết	IPA	Cách Viết	IPA	Cách Viết	IPA	Cách Viết	IPA	Cách Viết
/b/	b	/ʂ/	s	/w/	o, u	/i/	i, y	/u/	u	/-p/	p
/m/	m	/c/	ch			/e/	ê	/o/	ô, ôô	/-t/	t
/f/	ph	/ʈ/	tr			/ɛ/	e	/ɔ/	o, oo	/-k/	c, ch
/v/	v	/ɲ/	nh			/ɛ˘/	a (khi đứng trước /-k/, /-ŋ/)	/ɔ˘/	o (khi đứng trước /-k/, /-ŋ/)	/-m/	m
/t/	t	/l/	l			/i̯e/	iê, ia, yê, ya	/u̯o/	uô, ua	/-n/	n
/t'/	th	/k/	c, k, q			/ ɯ /	ư			/-ŋ/	ng, nh
/d/	đ	/χ/	kh			/ɤ/	ơ			/-u̯/	u, o
/n/	n	/ŋ/	ng, ngh			/a/	a			/-ˌi/	i, y
/z/	d, gi	/ɣ/	g, gh			/ɤ˘/	â				
/ʐ/	r	/h/	h			ă	ă, a (khi đứng trươc /-u̯/, /-ˌi/)				
/s/	x					/ɯ̯ɤ/	ươ, ưa				

Âm vị (phoneme) là đơn vị trừu tượng nhỏ nhất của một ngôn ngữ. Mọi âm tiết trong một ngôn ngữ đều được tạo ra bằng tổ hợp của các âm vị. Trong nhận dạng tiếng nói từ vựng lớn thì âm vị thường được chọn như là đơn vị của hệ thống nhận dạng với mục đích giảm số lượng đơn vị trong hệ thống. Trong một khoảng thời gian ngắn từ 10ms đến 40ms có thể coi tín hiệu âm thanh là ổn định và đây cũng là khoảng thời gian phổ biến cho một âm vị. Vì vậy chọn âm vị là đơn vị nhận dạng còn là để giảm ảnh hưởng của sự biến đổi của tín hiệu tiếng nói.

Dựa theo cấu trúc của âm tiết tiếng Việt thì hệ thống âm vị của tiếng Việt bao gồm 21 âm đầu, 1 âm đệm, 16 âm chính và 8 âm cuối [Chừ 1997]. Các âm vị tiếng Việt theo bản âm vị quốc tế (IPA) và cách thể hiện bằng chữ viết được trình bày ở Bảng 2-3.

2.2.2. Thanh điệu tiếng Việt

Về mặt hình thức nếu không xét đến sự biến đổi thanh điệu trên các phụ âm dừng ở cuối âm tiết thì tiếng Việt có 6 thanh điệu [Chừ 1997]. Bao gồm thanh huyền, ngã, hỏi, sắc, nặng và thanh bằng (không có thanh điệu, thể hiện trong chữ viết là không dấu).



Hình 2-1: Các đường đặc tính của 6 thanh điệu tiếng Việt

(*nguồn* [Hong Quang 2008])

a) **Thanh bằng:**(T1) là thanh điệu cao, có đường đặc tính bằng phẳng như đường số 1 trong Hình 2-1;

b) **Thanh huyền**:(T2) Là thanh điệu thấp, đường đặc tính có dạng bằng phẳng tương tự thanh bằng nhưng phần cuối có phần đi xuống thấp hơn như thể hiện ở đường số 2 trong Hình 2-1.

c) **Thanh ngã**:(T3) Đường số 3 Hình 2-1, đường đặc tính của thanh ngã biến đổi từ ngang, thấp rồi cao.

d) **Thanh hỏi**: (T4) Đường số 4 Hình 2-1, là thanh thấp và có đường đặc tính gãy ở giữa.

e) **Thanh sắc:**(T5) Đường số 5 Hình 2-1, đường đặc tính của thanh sắc có hướng đi lên.

f) **Thanh nặng**: (T6) Đường số 6 Hình 2-1, là thanh thấp và có đường đặc tính đi xuống.

Đối với các âm tiết kết thúc bởi các các phụ âm đóng "p, k, t" thì các âm tiết này có xu thế kết thúc nhanh hơn so với các âm tiết khác, chính vì thế hai thanh sắc và thanh nặng (trong tiếng Việt chỉ có hai thanh này tồn tại với các âm tiết kết thúc bằng các phụ âm đóng "p, k, t") cũng có xu hướng kết thúc nhanh hơn khi đi cùng với các âm tiết khác. Trong trường hợp này có thể coi tiếng Việt có 8 thanh điệu [Hong Quang 2008].

2.3. Mô hình cho hệ thống nhận dạng tiếng Việt từ vựng lớn

Xét một hệ thống nhận dạng có bộ từ vựng $W=\{W_i\}$, $i=(1,..,N)$ kích thước N. Xét một ngôn ngữ L có tập từ vựng là W^*. Mục tiêu của luận án là xây dựng mô hình hệ thống để có thể nhận dạng mọi từ trong L (trong phạm vi luận án này L là tiếng Việt). Khi đó ta sẽ có $W^* \equiv W$. Nếu mô hình hóa mỗi một từ W_i bởi một mô hình λ_i thì kích thước của hệ thống sẽ là N. Trong thực tế thì N thường là rất lớn, và rất khó để có thể liệt kê hết tất cả các từ của W^*. Nguyên nhân là do hạn chế của người liệt kê, tính chất vùng miền, tính chất thế hệ hoặc theo sự phát triển của văn hóa, công nghệ thì vẫn luôn có những từ mới được bổ sung vào W^*. Như vậy nhược điểm của phương pháp này là kích thước hệ thống rất lớn và hệ thống không có khả năng nhận dạng W_j nếu $W_j \in W^*$ nhưng $W_j \notin W$. Để giải quyết nhược điểm này thì W_i sẽ được phân tích thành một chuỗi các âm vị $W_i=\{\beta_{i,j}\}$, $j=1,..,M$ với M là số âm vị tạo ra W_i, trong đó $\beta_i \in \beta, i = 1,..,K$. Trong đó K là kích thước tập âm vị β. β là bộ âm vị theo cấu trúc ngữ âm của ngôn ngữ L. Cụ thể với tiếng Việt thì β chính là bộ 45 âm vị ($N=45$) trong Bảng 2-3. Với cách tiếp cận này thì các âm vị β_i sẽ được chọn làm đơn vị nhận dạng của hệ thống. Như vậy một cách tổng quát thì kích thước của hệ thống nhận dạng luôn là 45 và không phụ thuộc vào kích thước của W^*. Đồng thời do β_i được chọn từ β là tập tất cả các âm vị của L nên mọi từ có trong L đều có thể nhận dạng được bằng cách nhận dạng các âm vị cấu tạo nên nó. Mô hình xác suất để đoán nhận vector đặc trưng

đầu vào tại thời điểm k, x_k (hoặc một chuỗi vector x_k) là $\beta_{i,k}$ được xác định theo công thức (2.1).

$$P(\beta_i|x_i) = argmax_{\beta_i \in \beta} \sum_{j=1}^{M-1} P(\beta_{i-j,j}|x_{i-j}) * P(\beta_{i,k}|x_k) \quad (2.1)$$

Trong phạm vi luận án này bộ âm vị β được sử dụng làm bộ đơn vị nhận dạng của hệ thống gồm có 45 âm vị (không tính âm câm) như đã liệt kê ở Bảng 2-3.

2.4. Mô hình cho hệ thống nhận dạng tiếng Việt từ vựng lớn có thanh điệu

Theo cách biểu diễn sử dụng bộ âm vị β với 45 âm vị tiếng Việt (Bảng 2-3) thì về mặt mô hình không thể hiện sự khác biệt giữa hai âm tiết khác thanh điệu. Xét ví dụ hai âm tiết "ma" và "má". Nếu biểu diễn theo β thì ta có:

- Phiên âm của âm tiết "ma":

  g(W"ma") = Âmđầu("ma") + ÂmĐệm("ma") + ÂmChính("ma") + ÂmCuối("ma")
  = "m" + "null" + "a" + "null" = $\beta_1 + \beta_2$

  Trong đó:

  - *g()*: là hàm phân tích âm vị cho một âm tiết đầu vào.
  - β_1 = *"m"*, β_2 = *"a"*.
  - Phép "+" ở đây là phép ghép xâu.
  - *"null"* = ký tự rỗng.

- Phiên âm của âm tiết "má":

  g(W"má") = Âmđầu("má") + ÂmĐệm("má") + ÂmChính("má") + ÂmCuối("má")
  = "m" + "null" + "a" + "null" = $\beta_1 + \beta_2$

  Trong đó:

  - *g()*: là hàm phân tích âm vị cho một âm tiết đầu vào.
  - β_1 = *"m"*, β_2 = *"a"*.
  - Phép "+" ở đây là phép ghép xâu.
  - *"null"* = ký tự rỗng.

Như vậy mặc dù về mặt ngữ nghĩa và hình thức thì "ma" khác hoàn toàn "má", tuy nhiên theo cách biểu diễn ở trên thì *g(W"ma")*= *g(W"má")*= $\beta_1 + \beta_2$. Theo công thức (1.1) thì mô hình Bayes để tính toán xác suất đoán nhận hai từ đâu ra "ma" và "má" với vector đặc trưng đầu vào cho trước *X* được viết lại như sau:

$$P(W_{ma}|X) = \frac{P(X|W_{ma})P(W_{ma})}{P(X)} = \frac{P(X|\beta_1\beta_2)P(W_{ma})}{P(X)} \quad (2.2)$$

$$P(W_{má}|X) = \frac{P(X|W_{má})P(W_{má})}{P(X)} = \frac{P(X|\beta_1\beta_2)P(W_{má})}{P(X)} \quad (2.3)$$

Rõ ràng là giá trị tạo nên sự khác biệt cho $P(W_{ma}|X)$ và $P(W_{má}|X)$ chỉ còn phụ thuộc vào mô hình ngôn ngữ $P(W_{ma})$ và $P(W_{má})$. Hay nói cách khác là mô hình âm học không phân biệt được thanh điệu giữa các âm tiết có thanh điệu khác nhau. Việc nhận dạng thanh điệu hoàn toàn phụ thuộc vào mô hình ngôn ngữ.

Qua các nghiên cứu đã công bố cho tiếng Việt như [T. T. Vu 2005] [Thắng 2014] và kết quả nghiên cứu của Nghiên Cứu Sinh cùng nhóm nghiên cứu ở [Jonas 2013], cùng với các nghiên cứu trên các ngôn ngữ khác như Mandarin, Cantonese [Chen 2001] [Chong-Jia 2011] cho thấy việc chỉ sử dụng mô hình ngôn ngữ để nhận dạng thanh điệu không làm tối ưu chất lượng của hệ thống. Để khắc phục vấn đề này luận án sử dụng phương pháp tích hợp thêm thông tin thanh điệu vào bộ âm vị và sử dụng bộ âm vị có thanh điệu này (β^*) làm đơn vị nhận dạng của hệ thống. β^* được xây dựng từ β bằng cách thêm thông tin thanh điệu của âm tiết vào âm chính như sau:

$$\beta = \{\{\text{âm đầu}\}, \{\text{âm đệm}\}, \{\text{âm } chính\}, \{\text{âm } cuối\}\} \quad (2.4)$$

$$\beta^* = \left\{\{\text{âm đầu}\}, \{\text{âm đệm}\}, \{\text{Âm } chính\}\_\{Thanh \text{ điệu}\}, \{\text{âm } cuối\}\right\} \quad (2.5)$$

Trong đó tập {thanh điệu}={T1, T2, T3, T4, T5, T6} như đã trình bày ở mục 2.2.2. Theo phân tích của các nghiên cứu ngữ âm tiếng Việt [Chừ 1997] [Khang 2004] thì thanh điệu trong tiếng Việt thường tồn tại trong suốt phần vần bao gồm Âm đệm, Âm chính và Âm cuối. Tuy nhiên nếu bổ sung thêm thanh điệu vào cả 3 thành phần này thì số lượng đơn vị của β^* sẽ rất lớn, dẫn đến tăng độ phức tạp cho mô hình nhận dạng. Thực tế đã có các nghiên cứu công bố về việc tối ưu bộ âm vị, tối ưu vị trí đặt thanh điệu vào các âm vị trong âm tiết [T. T. Vu 2005] [Chuong 2014], tuy nhiên trong phạm vị của luận án này chỉ tập trung vào việc nghiên cứu phương pháp mô hình hóa và phương pháp tăng cường đặc trưng thanh điệu tiếng Việt theo đúng bản chất đứt gãy của nó trong điều kiện phát âm liên tục từ vựng lớn. Nên ở đây luận án sẽ chỉ sử dụng bộ âm vị β^* theo công thức (2.5). Sau khi bổ sung thanh điệu vào tất cả các âm chính và loại bỏ đi các âm vị không tồn tại trong thực tế thì luận án thu được β^* với 154 âm vị. Bảng 2-4 minh họa một số ví dụ khác về việc phân tích các âm tiết thành các âm vị có và không có thông tin thanh điệu.

Bảng 2-4: Một số ví dụ phiên âm sử dụng tập âm vị có thanh điệu

Âm tiết	Tiếng Anh	Âm vị không thanh điệu β	Thanh điệu	Âm vị có thanh điệu β^*
Không	Zero	/χ/ /o/ /-ŋ/	T1	/χ/ /o_T1/ /-ŋ/
Thuyền	Boat	/th/ /w/ /i̯e/ /-n/	T2	/th/ /w/ /i̯e_T2/ /-n/
Diễn	Act	/z/ /i̯e//-n/	T3	/z/ /i̯e_T3//-n/
Bẩy	Seven	/ʂ//ɤ̆//-ˀi/	T4	/ʂ//ɤ̆_T4//-ˀi/
Bốn	Four	/b/ /o/ /-n/	T5	/b/ /o_T5/ /-n/
Mụn	Spot	/m/ /u/ /-n/	T6	/m/ /u_T6/ /-n/

2.5. Thuật toán tạo từ điển ngữ âm tự động có thanh điệu cho tiếng Việt (VN-G2P)

Một trong các bước trước khi tiến hành phát triển một hệ thống nhận dạng tiếng nói là tạo từ điển ngữ âm. Tức là phiên âm các từ (word) thành chuỗi các âm vị cấu tạo nên từ đó (phoneme-based pronuncation). Đối với những hệ thống điều khiển hoặc ra lệnh bằng giọng nói, lượng từ vựng chỉ khoảng vài từ đến vài trăm từ thì công việc tạo từ điển có thể làm bằng tay. Tuy nhiên với các hệ thống nhận dạng từ vựng lớn cỡ vài nghìn từ trở lên thì việc tạo từ điển bằng tay là một công việc khó khăn và đòi hỏi kiến thức ngữ âm về ngôn ngữ đó. Với các ngôn ngữ có cấu trúc ngữ âm biến đổi, hình thái từ biến đổi như tiếng Anh thì công việc này lại càng khó. Để giải quyết công việc này ngay cả đối với những nhà nghiên cứu bản địa thì thông thường người ta sử dụng một mô hình xác suất gọi là mô hình chuyển đổi từ hình vị sang âm vị (grapheme-to-phoneme/G2P). Mô hình này trước tiên được huấn luyện để học các phiên âm từ một từ điển ngữ âm đã có sẵn. Sau đó mô hình này có thể được dùng để tạo từ điển âm vị cho bất kỳ từ mới nào. Tuy nhiên vì là mô hình xác suất nên độ chính xác của nó phụ thuộc vào dữ liệu học và thường là không thể đạt độ chính xác 100%. Chính vì vậy mà từ điển tạo ra cũng không thể chính xác 100% theo như cách mà con người phát âm. Ưu điểm của phương pháp G2P là nó có thể được dùng để tạo từ điển phát âm cho một ngôn ngữ mà người thực hiện không cần biết hoặc hiểu về ngôn ngữ đó. Điều này rất có ý nghĩa với những nhà nghiên cứu khi làm việc với các ngôn ngữ không phải là ngôn ngữ bản địa.

Tiếng Việt là một trong các ngôn ngữ mới được nghiên cứu cho các hệ thống nhận dạng và dịch tiếng nói. Hầu hết các nghiên cứu mới chỉ được thực hiện bởi các nhóm nghiên cứu người Việt hoặc có thành viên là người Việt. Do tiếng Việt là một ngôn ngữ khó và có thanh điệu. Chưa có nhiều các nguồn từ điển phát âm cho tiếng Việt được công bố trên thế giới. Việc này làm hạn chế số lượng các nhà nghiên cứu quốc tế có thể tham gia nghiên cứu trên tiếng Việt vì họ chưa biết tiếng Việt. Một mô hình xác suất G2P có thể được áp dụng để tạo từ điển phát âm cho tiếng Việt nếu họ đã

có một từ điển phát âm tiếng Việt với lượng từ vựng đủ lớn. Tuy nhiên tiếng Việt chuẩn lại là một ngôn ngữ có cấu trúc ngữ âm và quy tắc phát âm nhất quán. Hay nói cách khác là có khả năng đánh vần được. Dựa trên cấu trúc và tập âm vị của âm tiết tiếng Việt đã trình bày ở mục 2.2 luận án đề xuất một thuật toán tạo từ điển phát âm (VN-G2P) cho mọi từ tiếng Việt đầu vào. Đầu ra của thuật toán là từ điển phát âm được tạo nên từ tập các âm vị có thông tin thanh điệu β^* như đã trình bày ở mục trên.

***Tên thuật toán**: VN-G2P.*

Thuật toán được thực hiện với các giải thiết sau:

- $W = \{W_i\}, \mathrm{i} = 1, .., \mathrm{M}$ là tập từ vựng; M là kích thước bộ từ vựng;
- *Vocal(T)* là hàm trả về tập từ vựng có trong một văn bản đầu vào *T*. Khí đó ta có: *W ={Wi} =Vocal(T);*
- $I = \{I_i\},\ i = 1, .., 25$ là tập các biểu diễn bằng chữ viết của âm đầu được liệt kê trong cột "Âm đầu" Bảng 2-3;
- $O=\{O_i\}$, *i=1,2* là tập các biểu diễn bằng chữ viết của âm đệm được liệt kê trong cột "Âm đệm" Bảng 2-3;
- $C=\{C_i\}$, *i=1,..,12* là tập các biểu diễn bằng chữ viết của các âm cuối được liệt kế trong cột "Âm cuối" Bảng 2-3;
- $N=\{N_i\}$, *i=1, 22* là tập các biểu diễn bằng chữ viết của các âm chính được liệt kế trong cột "Âm chính" Bảng 2-3;
- *Del_L(X,Y):* Nếu xâu *'X'* chứa xâu *'Y'* và *'Y'* là các ký tự đầu tiên từ bên trái của *'X'* thí xóa *Y* khỏi *X*;
- *Del_R(X,Y):* Nếu xâu *'X'* chứa xâu *'Y'* và *'Y'* là các ký tự cuối cùng từ bên phải của *'X'* thí xóa *Y* khỏi *X*;
- *Find_L(X,Y):* Trả về giá trị "True" nếu xâu *'Y'* là các ký tự đầu tiên của *'X'* từ bên trái, trái lại trả về giá trị "False";
- *Find_R(X,Y):* Trả về giá trị "True" nếu xâu *'Y'* là các ký tự cuối cùng của *'X'* từ bên phải, trái lại trả về giá trị "False";
- *Len(X):* trả về độ dài của xâu *'X'*;
- *Tone(X):* trả về thanh điệu của có trong xâu *"X"* bằng cách so sánh "*X*" với các ký tự trong bảng mã Unicode của tiếng Việt.
- *Get_W(X):* Trả về xâu có nội dung tương tự "*X*" nhưng không có thanh điệu (Thanh bằng).

Thuật toán:

***Đầu vào**: T (Văn bản tiếng Việt).*

***Đầu ra**: Từ điển ngữ âm của bộ từ vựng trích ra từ văn bản đầu vào.*

***Chi tiết**:*

Bước 1: *W=Vocal(T)*

Bước 2: $W_j \leftarrow W$, *W'*=W_j

Bước 3:

- $I' = \begin{bmatrix} I_i: Find_{L(W_j,I_i)} = True, I_i \in I \\ \text{"Ký tự rỗng"} : Find\_L(W_j, I_i) = False \end{bmatrix}$
- W_j=*Del_L(W_j,I')*

Bước 4:

- $C' = \begin{bmatrix} C_i : Find\_R(W_j, C_i) = True \text{ và } C_i \notin \{o,u,i,y\}, C_i \in C \\ C_i: Len(W_j) > 1 \text{ và } C_i \in \{o,u,i,y\}, C_i \in C \\ \textit{"Ký tự rỗng"}: \textit{Trường hợp khác} \end{bmatrix}$
- W_j=*Del_R(W_j,C')*

Bước 5:

- $O' = \begin{bmatrix} \text{"o"}: Find\_L(W_j, \text{"o"}) \text{ và } Len(W_j) > 1 \\ \text{"u"}: Find\_L(Del(W_j, \text{"u"}), \text{"ô"}) = False \text{ và } Find\_L(Del(W_j, \text{"u"}), \text{"a"}) = False \text{ và } Len(W_j) > 1 \\ \text{"Kỹ tự rỗng"}: Len(W_j) \leq 1 \end{bmatrix}$

Bước 6:

- *T=Tone(N')*
- *N''=Get_W(N')*
- Nếu $N'' \notin N$ thì loại từ *W'* khỏi đầu ra

Bước 7: Xuất ra *(W' I' O' N'_T C'),* quay lại Bước 2 nếu $j \leq M$

Bước 8: Kết thúc thuật toán.

2.6. Dữ liệu thử nghiệm

2.6.1. Dữ liệu huấn luyện (Training)

Các thử nghiệm của nghiên cứu được thực hiện trên 3 bộ dữ liệu tiếng nói phát âm liên tục (tiếng nói được phát âm tự nhiên, không có điều kiện về khoảng cách giữa hai âm tiết cạnh nhau) với các thông số được trình bày ở Bảng 2-5.

Trong đó:

- IOIT2013 và VOV: Là dữ liệu được phát triển bởi Viện công nghệ thông tin – Viện Hàn lâm Khoa học và Công nghệ Việt Nam. IOIT2013 là dữ liệu tiếng nói đọc (read speech) được thu trong phòng cách âm cho cả 3 giọng Bắc, Trung và Nam. VOV là dữ liệu thu từ các mục bản tin, đọc truyện, phỏng vấn,… từ đài tiếng nói Việt Nam.
- GlobalPhone: Là dữ liệu được phát triển bởi Đại học Carnegie Mellon – USA[4].

[4]http://www.cs.cmu.edu/~tanja/GlobalPhone/index-e.html

Bảng 2-5: Dữ liệu huấn luyện

Tên	Kích thước theo giờ	Số người nói	Số lượng câu	Từ vựng	Chủ đề
VOV	17	30	20750	4908	Truyện, tin tức, phỏng vấn
IOIT2013	170	206	86000	5378	Nhiều chủ đề
GlobalPhone	19.7	129	19000	4200	Nhiều chủ đề

2.6.2. Dữ liệu thử nghiệm (Testing)

Nghiên cứu này sử dụng hai bộ dữ liệu đánh giá là VOV-Test và VoiceTra-Test. Trong đó VOV-Test có kích thước 2 giờ gồm 2688 câu có cùng chủ đề với tập huấn luyện VOV. VoiceTra-Test có kích thước 39 phút với 803 câu là dữ liệu được thu âm qua phần mềm dịch tiếng nói tự động được phát triển bởi Viện Công nghệ Thông tin và Truyền thông – Nhật Bản[5].

Bảng 2-6: Dữ liệu thử nghiệm

Tên	Kích thước theo giờ	Số người nói	Số lượng câu	Chủ đề
VOV-test	2	13	2688	Truyện, tin tức, phỏng vấn
VoiceTra-test	0.65	200	803	Nhiều chủ đề

Tất cả các bộ dữ liệu được lưu trữ dưới dạng file wav, tần số lấy mẫu là 16kHz và độ phân giải 16bit.

2.6.3. Đánh giá kích thước dữ liệu

Xét một số hệ thống nhận dạng được phát triển bởi các Viện hoặc Trường Đại học tiên tiến trên thế giới tính đến năm 2014 tại Califonia-Mỹ trong cuộc thi về xây dựng hệ thống dịch máy tiếng nói tự động thường niên "International Workshop on Spoken Language Translation (IWSLT)" như sau:

- Hệ thống nhận dạng tiếng Đức được phát triển bởi Viện công nghệ Karsuhe Đức [K. a. Kevin 2014] được xây dựng trên tập dữ liệu huấn luyện có kích thước 370 giờ và bộ từ điển kích thước 100.000 từ. Như vậy về mặt thống kê ta có trung bình 370*60/100000=0.222 phút mẫu thu âm cho 1 từ.

[5]http://www.nict.go.jp/en/

- Hệ thống nhận dạng tiếng Anh được phát triển bởi Viện Công nghệ thông tin quốc gia Nhật bản [Shen 2014]được xây dựng trên tập dữ liệu huấn luyện có kích thước 310 giờ và bộ từ điển kích thước 123.000 từ. Như vậy trung bình có 310*60/123000 ≈ 0.15 phút mẫu thu âm cho 1 từ.

Bộ dữ liệu huấn luyện thử nghiệm được sử dụng chính trong luận án là VOV có kích thước 17 giờ và bộ từ điển có kích thước là 5000 từ. Như vậy trung bình ta có 17*60/5000 ≈ 0.204 phút mẫu thu âm cho 1 từ. Như vậy tập VOV đủ độ lớn để có thể tiến hành các thử nghiệm. Và trong thực tế tập dữ liệu này cũng đã được dùng thử nghiệm trong một số các nghiên cứu trước đây như: Nghiên cứu của nhóm tác giả Vũ Tất Thắng [T. T. Vu 2005] thực hiện tại Nhật Bản. Nghiên cứu của nhóm tác giả Vũ Ngọc Thắng [N. T. Vu 2009] thực hiện tại Đức.

2.7. Tổng quan về công cụ HTK& HTS cho nhận dạng tiếng nói

2.7.1. Tổng quan về HTK

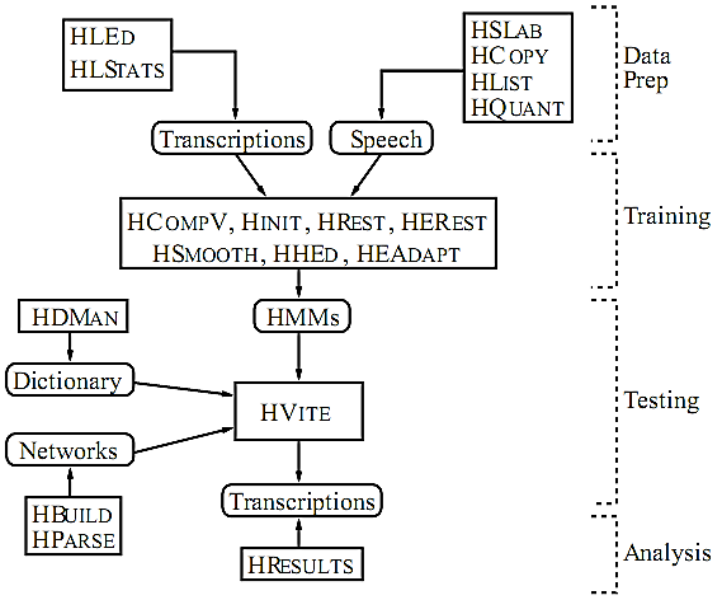
HTK (Hidden Markov Model Toolkit) là một bộ công cụ phát triển để xây dựng các mô hình Markov ẩn cho nhiều bài toán khác nhau, tuy nhiên HTK được thiết kế cho mục đích chính là phát triển các hệ thống nhận dạng tiếng nói. HTK là một bộ thư viện được viết trên ngôn ngữ C cung cấp các hàm liên quan đến trích chọn đặc trưng, xây dựng và huấn luyện mô hình HMM, bộ giải mã, huấn luyện thích nghi,… HTK được xây dựng đầu tiên bởi một nhóm nghiên cứu về học máy thuộc trường đại học Cambridge. Chức năng chính của HTK là dùng để huấn luyện các mô hình HMM dựa trên một tập các mẫu đã được gán nhãn trước. Sau đó HTK có thể sử dụng các mô hình HMM đã được huấn luyện để đoán nhận nhãn cho một tập mẫu khác [Young 2009].

Một cách tổng quát các công cụ của HTK có thể chia ra làm bốn nhóm dựa theo quy trình để xây dựng một hệ thống nhận dạng tiếng nói như Hình 2-2.

Trong đó:

- **Data preparing**: Bước chuẩn bị cơ sở dữ liệu. Tại bước này HTK hỗ trợ việc ghi, soạn các file âm thanh thông qua hàm HSLab. Tính toán đặc trưng thông qua hàm Hcopy. Hcopy hỗ trợ tính toán các loại đặc trưng như MFCC, PLP, Fillter bank,…Soạn và tạo các phiên âm (transription) bằng hàm HLed.

- **Training**: Đầu tiên các mô hình HMM sẽ được khởi tạo các tham số ngẫu nhiên ban đầu theo cấu hình đã chọn bằng hàm HInit. Sau đó các mô hình này được huấn luyện ở mức đơn âm (monophone) bằng hàm HRest. Các mô hình cho các âm buộc hay còn gọi là âm phụ thuộc ngữ cảnh (triphone) được tạo ra bằng hàm HHed dựa trên tập các mô hình đơn âm đã có, sau đó các mô hình này được huấn luyện lại bằng công cụ HERest.
- **Testing**: HTK cung cấp hai bộ nhận dạng là HVite và HDecode. HVite được sử dụng cho các hệ thống nhận dạng sử dụng mô hình ngôn ngữ ở mức 2-gram hoặc grammar. HDecode được sử dụng cho các hệ thống nhận dạng từ vựng lớn và sử dụng mô hình ngôn ngữ từ 3-gram trở lên.
- **Analysis**: Để đánh giá chất lượng nhận dạng của mô hình trên một tập mẫu đầu vào HTK cung cấp hàm HResults để tính toán các tham số độ chính xác theo từ (Word Accuracy - ACC) và độ chính xác theo câu (Sentence Accuracy).



Hình 2-2: Quy trình xây dựng một hệ thống nhận dạng tiếng nói trên HTK
(*hình ảnh được trích dẫn từ* **[Young 2009])**

2.7.2. Tổng quan về HTS

HTS (HMM-based Speech Synthesis System) [Oura 2011]là một công cụ để xây dựng các hệ thống tổng hợp tiếng nói. HTS được xây dựng dựa trên HTK, vì thế có thể coi HTS là một phiên bản chỉnh sửa của HTK dành cho tổng hợp tiếng nói. Tất cả các bước huấn luyện mô hình HMM của HTS tương tự như HTK. Điều đặc biệt trong HTS là nó hỗ trợ mô hình phân bố đa không gian MSD-HMM (Mutli-space

Distribution HMM). Đây là loại mô hình có khả năng mô hình hóa dữ liệu đầu vào chứa cả dữ liệu liên tục và dữ liệu rời rạc. MSD-HMM được thiết kế chính cho các mô hình tổng hợp tiếng nói nhưng trong nghiên cứu này nó sẽ được đề xuất áp dụng cho nhận dạng tiếng Việt.

Mô hình MSD-HMM trong HTS được khai báo như sau:

```
globalOpts=option { option }
option= <HmmSetId> string
<StreamInfo> short { short }
<MSDInfo> short { short }
<VecSize> short
<ProjSize> short
<InputXform> inputXform
<ParentXform> ~a macro
covkind
durkind
parmkind
```

Các trường khai báo khác tương tự HTK, HTS bổ sung thêm một số trường để hỗ trợ MSD gồm: <MSDInfo>để khai báo số luồng (Stream) của dữ liệu đầu vào, nếu luồng tương ứng được đánh dấu là 1 thì nó sẽ được áp dụng mô hình MSD, trái lại không áp dụng. Ví dụ một khai báo như "<MSDInfo> 0 0 0 1" sẽ chỉ ra rằng dữ liệu đầu vào có 4 luồng độc lập và MSD sẽ được áp dụng cho luồng thứ 4. Để tương thích với mô hình MSD-HMM đa đầu vào HTS còn hỗ trợ khả năng khai báo từng thành phần trộn Gaussian (Mixtures) cho từng luồng độc lập như ví dụ sau:

```
################################
~o <VecSize> 45 <USER><DIAGC><MSDInfo> 4  0  1  1  1 <StreamInfo> 4 42 1 1
1
<BeginHMM>
<NumStates> 5
<State> 2
<SWeights> 4 1.0 1.0 1.0 1.0
<Stream> 1
<Mean> 42
      0.0 … 0.0
<Variance> 42
      1.0 … 1.0
<Stream> 2
<NumMixes> 2
<Mixture> 1 0.5000
<Mean> 1
      0.0
```

```
<Variance> 1
      1.0
<Mixture> 2 0.5000
<Mean> 0
<Variance> 0
<Stream> 3
…
#################################
```

Ở khai báo trên định nghĩa một mô hình MSD-HMM với 4 luồng độc lập trong đó luồng thứ nhất có số chiều là 42 và không áp dụng MSD. 3 luồng còn lại có số chiều là 1 và để áp dụng MSD. Số Mixture của luồng 1 là 1, số Mixture của luồng 2, 3, 4 đều là 2 và được định nghĩa sau thẻ <Mixture>.HTS cũng cải tiến một số hàm khác của HTS để có thể làm việc với loại mô hình MSD mới, các hàm chính đã được cải tiến gồm: HHed, HRest, HERest, HAdapt, HMap, HGen, HCompv, HVite,…

2.8. Thử nghiệm mô hình không có thanh điệu (Hệ thống nhận dạng cơ sở Baseline)

Các nghiên cứu đã có về nhận dạng tiếng Việt hiện mới chỉ áp dụng mô hình HMM trên các loại đặc trưng phổ biến là MFCC hoặc PLP trên bộ từ vựng kích thước nhỏ cỡ vài trăm từ hoặc trên tiếng nói phát âm rời rạc. Tính đến hiện nay cũng chưa có một công bố nào về bộ dữ liệu chuẩn cho huấn luyện và đánh giá chất lượng hệ thống chung cho cộng đồng nghiên cứu nhận dạng tiếng Việt. Nghĩa là các thử nghiệm của các nghiên cứu đã khó có thể so sánh với nhau do không cùng các điều kiện tiêu chuẩn như dữ liệu thử nghiệm, đầu ra của hệ thống nhận dạng. Vì thế để có thể so sánh và đánh giá chất lượng của các phương pháp mới trong luận án thì một hệ thống nhận dạng cơ sở (Baseline) ban đầu cần được xây dựng. Hệ thống cơ sở này được xây dựng dựa trên mô hình không có thanh điệu. Tức là bộ âm vị không có thanh điệu sẽ được sử dụng làm đơn vị nhận dạng và được mô hình hóa bởi mô hình HMM truyền thống với các tham số chính như sau:

Đặc trưng đầu vào: MFCC/PLP. Trong đó kích thước của mỗi vector đặc trưng MFCC/PLP là 39 bao gồm 13 thành phần MFCC/PLP, 13 thành phần Delta và 13 thành phần Acceleration của MFCC/PLP.

- Mô hình HMM: Được huấn luyện ở mức triphone với 2179 âm buộc (tied-states).
- Từ điển: Từ điển sử dụng tập âm vị không có thông tin thanh điệu có 45 âm vị.
- Mô hình ngôn ngữ: 2-gram được xây dựng từ dữ liệu phiên âm của VOV.

- Kết quả đánh giá theo tham số độ chính xác theo từ ACC (Word Accuracy) đạt 77.70% với đặc trưng MFCC.

Các bước xây dựng hệ thống cơ sở như sau:

2.8.1. Dữ liệu

- Dữ liệu huấn luyện: VOV
- Dữ liệu thử nghiệm: VOV-test

2.8.2. Chuẩn hoá dữ liệu

Tất các các file phiên âm (transcription) của các câu phát âm đã được đưa về cũng định dạng chữ in thường, tất cả các chữ số được chuyển thành chữ tương ứng với phát âm của nó. Loại bỏ tất cả các ký tự đặc biệt như: “.”, “,”, “&”,…

2.8.3. Trích chọn đặc trưng

Hai loại đặc trưng cơ sở được sử dụng trong nghiên cứu là MFCC và PLP. Hai đặc trưng này được tạo ra từ các file wav trong bộ dữ liệu VOV và VOVTest sử dụng hàm HCopy của HTK.

2.8.4. Từ điển

Từ điển ngữ âm được tạo ra thông qua một thuật toán VN-G2P. Từ điển này sử dụng các phiên âm trên tập âm vị không có thông tin thanh điệu, tổng số âm vị là 45. Số từ vựng của hệ thống là 4908 từ có dấu, được trích trọn từ toàn bộ dữ liệu huấn luyện. Một số ví dụ về các phiên âm trong từ điển trình bày trong Bảng 2-7.

Bảng 2-7: Ví dụ một số phiên âm trong từ điển

Từ tiếng Việt	Phiên âm
anh	ea ngz
ánh	ea ngz
còi	k ow iz
cói	k ow iz

2.8.5. Mô hình âm học

Mô hình âm học được huấn luyện ở mức Tri-phone với 2179 âm buộc, sử dụng 16 thành phần trộn (Gaussian mixture).

2.8.6. Mô hình ngôn ngữ

Do tất cả các thí nghiệm kiểm thử trong nghiên cứu này sử dụng hàm Hvite của HTK nên mô hình ngôn ngữ được sử dụng là mô hình Bi-gram được xây dựng từ tất cả các phiên âm (transcript) của dữ liệu huấn luyện VOV (VOV-BiGgram-LM). Công cụ để tạo mô hình ngôn ngữ này hàm LGPrep của HTK.

2.8.7. Thử nghiệm (Testing)

Kết quả nhận dạng trên dữ liệu kiểm thử VOV-test được đánh giá theo tham số độ chính xác theo từ ACC (word accuracy) trên hai loại đặc trưng PLP và MFCC được trình bày ở Bảng 2-8.

Bảng 2-8: Kết quả nhận dạng của hệ thống cơ sở

Hệ thống	Đặc trưng	ACC(%)
Sys1(**Baseline**)	MFCC	**77.70**
Sys2	PLP	76.77

Như vậy đặc trưng MFCC cho chất lượng nhận dạng tốt hơn PLP là 0.93% theo ACC. Hệ thống sử dụng MFCC sẽ được tham chiếu như một hệ thống cơ sở (Baseline) ban đầu để đánh giá các đề xuất và cải tiến của luận án sau này.

2.9. Thử nghiệm mô hình có thanh điệu

Hai thử nghiệm tiếp theo được thực hiện để đánh giá hiệu quả của hệ thống sử dụng mô hình có thanh điệu. Trong các hệ thống này thì bộ âm vị có thông tin thanh điệu sẽ được sử dụng thay vì bộ âm vị không có thanh điệu như trong hệ thống cơ sở. Các âm vị này vẫn được mô hình hóa bằng mô hình HMM. Thử nghiệm thứ nhất được thực hiện trên bộ công cụ HTK với tập dữ liệu VOV. Thử nghiệm thứ hai được thực hiện trên bộ công cụ Kaldi sử dụng các tập dữ liệu huấn luyện VOV, IOIT2013, Globalphone, VoiceTra với mục tiêu đánh giá khách quan từ điển này trên một môi trường phát triển khác và với tập dữ liệu kích thước lớn hơn, đồng thời kết quả thử nghiệm này của luận án cũng được áp dụng để xây dựng phần mềm dịch tiếng nói tự động quốc tế VoiceTra [Matsuda 2013] cho dự án liên kết giữa IOIT-Việt Nam và NICT-Nhật Bản. Đóng góp này của luận án được công bố ở [Van Huy 2015].

2.9.1. Thử nghiệm với HTK

1) Dữ liệu:

- Dữ liệu huấn luyện: VOV.
- Dữ liệu kiểm thử: VOV-test.

2) Trích chọn đặc trưng

Hai loại đặc trưng được trích chọn từ dữ liệu huấn luyện là MFCC và PLP sử dụng hàm HCopy của HTK với các tham số tương tự như hệ thống cơ sở ở Mục 2.8.

3) Từ điển

Bộ từ điển sử dụng tập âm vị có thông tin thanh điệu được tạo ra bằng cách áp dụng thuật toán VN-G2P trên dữ liệu phiên âm của tập VOV. Từ điển thu được có 4908 từ và 154 âm vị. Từ điển này được gọi là Tonal-Dict. Để đánh giá hiệu quả của tập âm vị có thông tin thanh điệu thì một loại từ điển thứ hai được tạo ra bằng cách xoá bỏ tất cả các ký hiệu thanh điệu trong các âm vị của từ điển Tonal-Dict thu được từ điển NonTonal-Dict với tập âm vị chỉ còn 45 âm vị và không chứa thông tin thanh điệu. NonTonal-Dict đã được sử dụng để xây dựng hệ thống cơ sở (Baseline system) ở mục 2.8.

4) Huấn luyện mô hình âm học.

Hai hệ thống được huấn luyện tương ứng với hai loại đặc trưng MFCC và PLP sử dụng bộ từ điển Tonal-Dict được ký hiệu lần lượt là HMM-1 và HMM-2. Các bước huấn luyện được tiến hành tương tự như hệ thống Baseline sử dụng công cụ HTK.HMM-1 và HMM-2 được huấn luyện ở mức tri-phone với 2179 âm buộc, mỗi state sử dụng 16 thành phần trộn Gaussian.

5) Mô hình ngôn ngữ:

Mô hình ngôn ngữ VOV-BiGram-LM của hệ thống Baseline được sử dụng lại cho các thí nghiệm ở đây.

6) Kết quả thử nghiệm

Kết quả nhận dạng của HMM-1 và HMM-2 theo tham số độ chính xác (Accuracy - ACC) trên tập thử nghiệm VOV-Test được trình bày ở Bảng 2-9.

Bảng 2-9: Kết quả thử nghiệm mô hình thanh điệu

TT	Hệ thống	Đặc trưng	Từ điển	ACC (%)
1	Baseline	MFCC	NonTonal-Dict	77.70
2	HMM-1	PLP	Tonal-Dict	77.58
3	HMM-2	MFCC		**78.31(+0.61)**

Từ kết quả thử nghiệm cho thấy mô hình có thanh điệu cho kết quả nhận dạng tốt hơn mô hình không có thanh điệu trên cả hai loại đặc trưng MFCC và PLP. Cụ thể hệ thống sử dụng MFCC tốt hơn 0.61% tuyệt đối so với hệ thống cơ sở.

2.9.2. Thử nghiệm với công cụ Kaldi sử dụng cơ sở dữ liệu lớn

Kaldi [P. a. Daniel 2011] là một trong các công cụ mã nguồn mở để phát triển các hệ thống nhận dạng tiếng Nói được sử dụng phổ biến nhất hiện nay. Rất nhiều các tổ chức nghiên cứu về nhận dạng tiếng nói uy tín lâu năm hiện cũng đang sử dụng công cụ này. Ưu điểm của Kaldi là đã tích hợp rất nhiều các kỹ thuật mới hiện nay như mạng nơron học sâu (Deep learning), đặc trưng phụ thuộc người nói i-vector, đặc trưng thanh điệu Pitch, các kỹ thuật huấn luyện phụ thuộc người nói (speaker adptive training), … Để đánh giá khách quan hơn về hiệu quả của mô hình có thanh điệu luận án tiến hành xây dựng hai hệ thống trên hai loại từ điển Tonal_Dict (sử dụng bộ âm vị có thanh điệu) và NonTonal-Dict (sử dụng bộ âm vị không có thanh điệu). Chi tiết các bước thử nghiệm được tiến hành như sau:

1) Dữ liệu

- Dữ liệu huấn luyện mô hình âm học: VOV+IOIT2013+GlobalPhone, kích thước khoảng 210 giờ.
- Dữ liệu liệu thử nghiệm: VoiceTra-test
- Dữ liệu mô hình ngôn ngữ: Toàn bộ phần phiên âm của dữ liệu âm thanh với khoảng 128000 câu phát âm.

2) Trích chọn đặc trưng

Đặc trưng sử dụng cho thử nghiệm này là đặc trưng kết hợp cả MFCC và Pitch (MFCC+P). Mỗi vector đặc trưng gồm 42 thành phần gồm 13 thành phần MFCC, 13 thành phần delta, 13 thành phần double delta, 1 thành phần là giá trị đặc trưng Pitch NCC, 1 thành phần là delta của NCC và 1 thành phần cuối cùng là giá trị xác suất của

khung hiện thời là voice/unvoiced. MFCC được trích chọn với cửa sổ 25ms, độ lệch giữa các khung là 10ms trong dải băng tần từ 20Hz-7000Hz, dải tần để tính toán Pitch từ 50Hz đến 400Hz.

3) Từ điển

Tương tự như thử nghiệm trên công cụ HTK hai từ điển Tonal-Dict và NonTonal-Dict được tạo ra sử dụng thuật toán VN-G2P, trong đó NonTonal-Dict thu được bằng cách xóa bỏ tất cả các ký hiệu thanh điệu ra khỏi các âm vị. Từ vựng của hai từ điển có kích thước 5378 từ, bộ âm vị của Tonal-Dict là 154, của NonTonal-Dict là 45.

4) Mô hình âm học

Hai hệ thống Kaldi-HMM-1 và Kaldi-HMM-2 được huấn luyện trên hai bộ từ điển Tonal-Dict và NonTonal-Dict. Các mô hình âm học của hai hệ thống được huấn luyện ở mức tri-phone với 3459 âm buộc, mỗi state sử dụng 16 thành phần trộn Gaussian. Các bước huấn luyện bao gồm các bước cơ bản sau:

1. Khởi tạo các mô hình HMM với tham số ngẫu nhiên cho các âm vị trong từ điển (mono-phone training)
2. Huấn luyện lại các mô hình mono-phone trên toàn bộ dữ liệu huấn luyện vòng lặp. Tại mỗi vòng lặp thứ *i* mô hình đã thu được ở vòng thứ *i-1* được sử dụng để gán nhãn lại dữ liệu huấn luyện và mô hình sẽ được ước lượng lại tham số trên dữ liệu gán nhãn mới này.
3. Huấn luyện các mô hình âm vị phụ thuộc ngữ cảnh mức tri-phone với 4000 âm buộc, mỗi state sử dụng 18 thành phần trộn Gaussian. Các mô hình cũng được huấn luyện với 40 vòng lặp, tại mỗi vòng lặp dữ liệu cũng được gán nhãn lại như bước huấn luyện mono-phone.

5) Mô hình ngôn ngữ

Mô hình ngôn ngữ được huấn luyện ở mức 3-gram với tập từ vựng lấy từ từ điển ngữ âm. Dữ liệu huấn luyện là toàn bộ phần phiên âm của dữ liệu âm thanh. Công cụ huấn luyện là Srilm [SRI 2011]. Mô hình ngôn ngữ thu được có giá trị OOV (Out Of Vocabulary) và PPL (Perplexity) trên tập dữ liệu thử nghiệm lần lượt là 31 và 141.

6) Kết quả thử nghiệm

Kết quả nhận dạng trên bộ dữ liệu thử nghiệm VoicTra-test được trình bày ở Bảng 2-10.

Bảng 2-10: Kết quả thử nghiệm mô hình thanh điệu với Kaldi

TT	Hệ thống	Đặc trưng	Từ điển	ACC (%)
1	Kaldi-HMM-1	MFCC+P	NonTonal-Dict	45.63
2	Kaldi-HMM-2	MFCC+P	Tonal-Dict	**47.17** (+1.54)

Như vậy trong thử nghiệm trên công cụ Kaldi với tập dữ liệu lớn, mô hình có thanh điệu đã giúp tăng độ chính xác lên 1.54% tuyệt đối so với mô hình không có thanh điệu. Chất lượng nhận trên thử nghiệm này kém hơn thử nghiệm trên công cụ HTK do sử dụng bộ dữ liệu thử nghiệm khó hơn. Bộ dữ liệu thử nghiệm này được phát triển bởi Viện công nghệ Thông tin Nhật bản (NICT) trong dự án VoiceTra[6]. Đây là tập dữ liệu thu âm ở môi trường bên ngoài tại các nhà ga, vỉa hè,… nơi có rất nhiều tập âm và nhiễu. Đồng thời nội dung là các câu hỏi về các địa điểm, tên đường với nhiều từ tiếng nước ngoài. Chính các lý do này dẫn đến kết quả kém hơn so với thử nghiệm trên HTK. Bộ dữ liệu thử nghiệm trên HTK là dữ liệu thu trong phòng thu, và có nội dung tương đồng với dữ liệu huấn luyện.

2.10. Kết luận chương

Trong chương này luận án đã trình bày phương pháp xây dựng mô hình có thanh điệu và không có thanh điệu cho hệ thống nhận dạng tiếng Việt từ vựng lớn phát âm liên tục sử dụng mô hình HMM. Qua kết quả thử nghiệm cho thấy mô hình có thanh điệu cho kết quả tốt hơn khoảng 3% tương đối so với mô hình không có thanh điệu. Kết quả này cũng tương đồng với các kết quả trên các ngôn ngữ khác như Mandarin, Cantonese khi áp dụng mô hình thanh điệu. Như vậy trong điều kiện từ vựng lớn và tiếng nói phát âm liên tục thì thanh điệu vẫn là một yếu tố quan trọng góp phần tối ưu chất lượng hệ thống nhận dạng cho tiếng Việt. Với mô hình này nghiên cứu sinh cũng đạt được kết quả tăng chất lượng tương tự trên các bộ dữ liệu trong các điều kiện khác nhau như dữ liệu thu âm qua điện thoại [Jonas 2013] với khoảng 5% tuyệt đối, dữ liệu lớn với nhiều chủ đề [Van Huy 2015] với 1.54% tuyệt đối.

Trong chương này luận án cũng đã đề xuất thuật toán tạo từ điển ngữ âm tự động VN-G2P sử dụng bộ âm vị có thanh điệu. Với thuật toán này người sử dụng đặc biệt là những người sử dụng là người nước ngoài không có hiểu biết về tiếng Việt vẫn

[6] http://voicetra.nict.go.jp/en/

có thể dễ ràng tạo ra từ điển ngữ âm tiếng Việt cho các nhiệm vụ nhận dạng hoặc tổng hợp tiếng Việt. Thuật toán này đã được công bố ở nghiên cứu [Van Huy 2015].

2.11. Các bài báo đã công bố liên quan đến nội dung của chương

1. **Van Huy Nguyen**, Chi Mai Luong, Tat Thang Vu, *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-China, Oct-2015.
2. Jonas Gehring, Kevin Kilgour, Quoc Bao Nguyen, **Van Huy Nguyen**, Florian Metze, Zaid A. W. Sheikh, Alex Waibel, *Models of tone for tonal and non-tonal languages*, Automatic Speech Recognition and Understanding (ASRU), Czech Republic, Dec-2013.

Chương 3: Mô hình thanh điệu sử dụng MSD cho nhận dạng tiếng Việt từ vựng lớn phát âm liên tục

3.1. Tóm tắt chương

Các nghiên cứu đã công bố cho nhận dạng tiếng Việt mới chỉ áp dụng mô hình HMM truyền thống. Mặc dù đặc trưng thanh điệu đã được sử dụng trong một số nghiên cứu nhưng các đặc trưng thanh điệu này đã được bổ sung các giá trị "nhân tạo" vào các vùng vô thanh và sau đó được mô hình hóa bởi HMM. Chương này của luận án sẽ trình bày một phương pháp mới để mô hình hóa đặc trưng thanh điệu ngay cả khi nó bị đứt gãy bằng mô hình phân bố đa không gian MSD-HMM. Mô hình này được áp dụng khá phổ biến cho tổng hợp tiếng nói nhưng mới được nghiên cứu áp dụng thành công duy nhất cho tiếng Mandarin.

Nội dung chính của chương bao gồm: Tổng quan về mô hình phân bố đa không gian MSD-HMM và đề xuất phương pháp áp dụng mô hình này cho nhận dạng tiếng Việt; Phương pháp tính toán và chuẩn hóa đặc trưng thanh điệu tương thích với mô hình MSD-HMM.

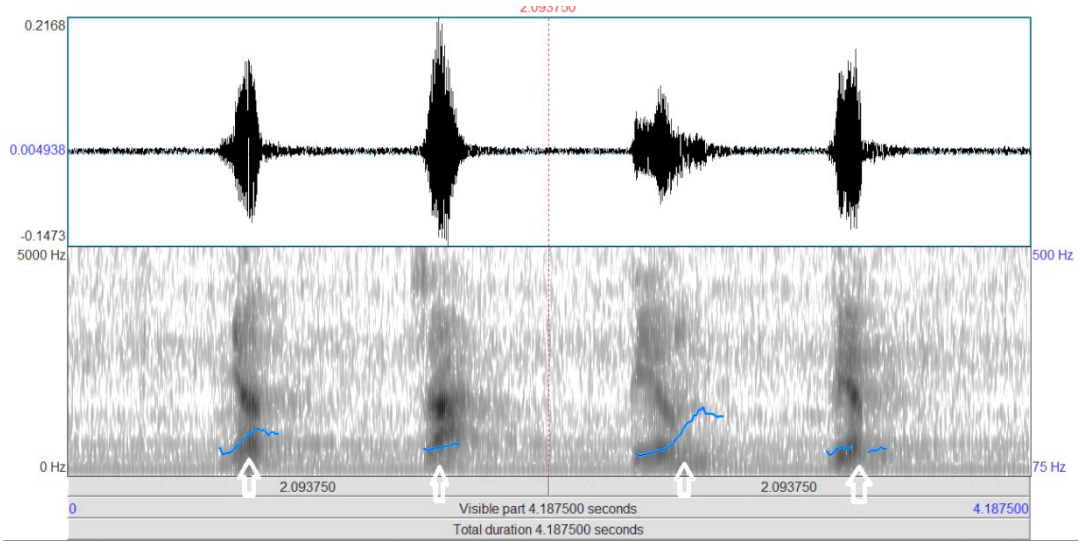
3.2. Vai trò của đặc trưng thanh điệu

Thanh điệu được tạo ra do dao động của dây thanh trong quá trình phát âm của bộ máy tạo âm. Trong lĩnh vực nhận dạng tiếng nói thì tần số cơ bản F0 thường được sử dụng để biểu diễn đặc trưng của thanh điệu [Hong Quang 2008] [Jurafsky 2008]. F0 không thường được tính toán trong miền tần số 0Hz-250Hz và nó có thể tồn tại trong suốt khoảng thời gian phát âm của một âm tiết (thường ≥200ms). Trong khi đó các âm vị cấu tạo nên các âm tiết được tạo ra do dao động của thanh quản, cấu hình của khoang miệng và lưu lượng khí thoát ra từ phổi. Đặc trưng biểu diễn cho các âm vị này được gọi là đặc trưng ngữ âm với các phương pháp phổ biến là MFCC, PLP được tính toán trong miền tần số 300Hz-8000Hz. MFCC và PLP thường được tính toán trong một đoạn thời gian đủ nhỏ để coi tín hiệu tiếng nói là ổn định, khoảng thời gian này thường từ 25ms-40ms. Như vậy xét cả về nguồn gốc, khoảng thời gian tồn tại và phương pháp tính toán thì đặc trưng thanh điệu (Pitch) và đặc trưng ngữ âm (MFCC/PLP) đều khác nhau. Đối với tiếng Việt thì thanh điệu ảnh hưởng đến ngữ nghĩa của âm tiết đi cùng nó. Một cách tổng quát thì mỗi âm tiết có thể có sáu ngữ nghĩa khác nhau khi kết hợp với sáu thanh điệu tiếng Việt. Ở Chương 2 luận án đã trình bày mô hình thanh điệu cho tiếng Việt. Theo đó thì hai âm tiết khác thanh điệu sẽ có mô hình khác nhau, vì vậy rõ ràng là cần thiết phải bổ sung thêm đặc trưng thanh

điệu để tăng thêm đặc tính khác biệt cho hai mô hình của hai âm tiết chỉ khác nhau phần thanh điệu.

3.3. Đặc trưng thanh điệu và vấn đề không liên tục

Thanh điệu được tạo ra do dao động của dây thanh. Tuy nhiên dây thanh chỉ dao động đối với các âm hữu thanh vì vậy mà trong vùng âm vô thanh không tồn tại thanh điệu. Nếu xét trong cả một câu phát âm thì đường đặc trưng của thanh điệu sẽ bị đứt gãy tại các vùng vô thanh. Hình 3-1 mô tả đường đặc trưng thanh điệu không liên tục của câu nói "nhận dạng tiếng Việt". Để có thể mô hình hoá đặc trưng thanh điệu sử dụng mô hình HMM hoặc mạng nơron thì đặc trưng này cần phải được áp dụng một kỹ thuật tiền xử lý trước để bổ sung các giá trị cho các vùng đứt gãy. Biện pháp đơn giản nhất là thay thế các vùng đứt gãy bằng giá trị 0. Hoặc có thể áp dụng một số kỹ thuật làm trơn khác. Tuy nhiên việc áp dụng các kỹ thuật khác để bổ sung giá trị vào vùng mà thanh điệu không tồn tại sẽ làm biến đổi đặc trưng này và đặc trưng mới không còn thể hiện đúng đắn đặc tính đó. Đối với các ngôn ngữ không có thanh điệu như tiếng Anh, Pháp đặc trưng thanh điệu chỉ làm tăng các thông tin về ngữ điệu, người nói, giới tính,… do nó không làm thay đổi ngữ nghĩa của âm tiết. Vì thế việc thay đổi đặc trưng thanh điệu bằng việc bổ sung các giá trị "nhân tạo" cũng có thể chấp nhận được hoặc thậm chí có thể bỏ qua đặc trưng này khi xây dựng các hệ thống nhận dạng tiếng nói. Đối với tiếng Việt do thanh điệu còn ảnh hưởng trực tiếp đến ngữ nghĩa của từ, vì vậy việc thay đổi nó có thể làm giảm chất lượng nhận dạng. Như vậy cần thiết phải có một phương pháp mô hình hoá sao cho có thể mô hình hoá được đặc tính thanh điệu bị đứt gãy để mô tả đúng nhất đặc tính của nó trong việc góp phần thay đổi ngữ nghĩa trong tiếng Việt.



Hình 3-1: Đường pitch của câu nói "Nhận dạng tiếng Việt"

Tính đến hiện nay có rất nhiều cách nghiên cứu đề xuất các kỹ thuật để trích chọn đặc trưng thanh điệu thông qua việc tính toán tần số cơ bản (F0) từ tín hiệu tiếng nói. Mục tiêu của luận án là đề xuất một mô hình có khả năng mô hình hoá loại đặc tính bị đứt gãy hay nói cách khác là mô hình được loại đặc trưng đầu vào chứa cả giá trị liên tục và giá trị rời rạc. Trong phạm vi nghiên cứu này luận án sẽ sử dụng hai phương pháp trích chọn đặc thanh điệu được sử dụng phổ biến là đặc trưng về độ lệch biên độ trung bình (AMDF - Average Magnitude Difference Function) và đặc trưng giá trị tương quan chéo đã chuẩn hoá (NCC - Normalized Cross-Correlation). NCC tính toán đặc trưng thanh điệu bằng phương pháp tương quan chéo. Cả NCC và AMDF đều thay thế các giá trị ở vùng vô thanh bằng giá trị 0. Mục đích việc sử dụng hai loại đặc trưng này trong luận án là muốn kiểm chứng chất lượng của mô hình đề xuất và xác định loại đặc trưng nào trong hai phương pháp NCC và AMDF phù hợp với loại mô hình này. Phương pháp tính toán NCC và AMDF được trình bày ở phần sau đây.

3.3.1. Đặc trưng thanh điệu NCC (giá trị tương quan chéo đã chuẩn hoá)

Phương pháp NCC [Talkin 1995] tính toán đặc trưng thanh điệu (pitch) dựa trên giả thiết tổng của các tích giữa 2 giá trị cách nhau đúng bằng chu kỳ của pitch sẽ có giá trị lớn nhất. Giá trị NCC được tính toán theo công thức (3.1).

$$NCC(k) = \frac{1}{\sqrt{e_0 e_k}} \sum_{k=0}^{N-T} s(n)s(n+k) \qquad (3.1)$$

Trong đó:

- *s(n):* tín hiệu rời rạc đầu vào với *0≤n≤N, N* là kích thước của khung tín hiệu.
- *k:* chu kỳ của pitch cần tìm, $k \leq T \leq N$. (*T* là kích thước của khung tinh toán)
- $e_k = \sum_{n-k}^{N-K+k} s^2(n)$.

NCC là phương pháp được cải tiến từ phương pháp tự tương quan Autucorrelation [Talkin 1995], cải tiến của NCC là giá trị của nó được chuẩn hoá theo hàm năng lượng tương ứng với khung tín hiệu đã được tính toán. Vì vậy NCC sẽ ít bị ảnh hưởng bởi các thành phần biến đổi nhanh trong s(n). Sau khi tính toán các giá trị NCC thì pitch có thể được xác định như công thức (3.2).

$$pitch = \tau, nếu\ NCC(\tau) = agr_{max}\{NCC(k)\}, k = 1, .., T-1 \qquad (3.2)$$

3.3.2. Đặc trưng thanh điệu AMDF (độ lệch biên độ trung bình)

NCC là một phương pháp tính toán pitch cho kết quả tốt và thực tế hiện nay phương pháp này được tích hợp vào rất nhiều công cụ xử lý cũng như nhận dạng tiếng nói được sử dụng phổ biến trên thế giới như Speech Signal Processing Toolkit (SPTK) [SPTK 2014], Kaldi [P. a. Daniel 2011] và SNACK [Snack 2004]. Tuy nhiên nhược điểm của NCC là tốc độ tính toán chậm do sử dụng phép tính nhân/chia trong quá trình tính toán. Điều này có thể khiến các hệ thống nhận dạng online làm việc chậm. Phương pháp tính toán pitch dựa trên độ lệch biên độ trung bình AMDF sẽ khắc phục nhược điểm này. Phương pháp AMDF [Talkin 1995] xác định pitch dựa trên giả thiết là tổng của hiệu giữa hai giá trị cách nhau đúng bằng chu kỳ pitch sẽ có giá trị nhỏ nhất. Công thức xác định AMDF được cho ở công thức (3.3).

$$AMDF(k) = \frac{1}{K}\sum_{k=0}^{N-T} |s(n) - s(n+k)| \qquad (3.3)$$

Trong đó:

- *N là* kích thước của khung tín hiệu, *s(n)*là giá trị tín hiệu đầu vào.
- *k:* chu kỳ của pitch cần tìm, $k \leq T \leq N$.

AMDF được xác định chỉ thông qua các phép tính Cộng và Trừ, vì thế tốc độ cải thiện đáng kể so với NCC. Sau khi tính toán các giá trị AMDF, pitch được xác định theo như công thức (3.4).

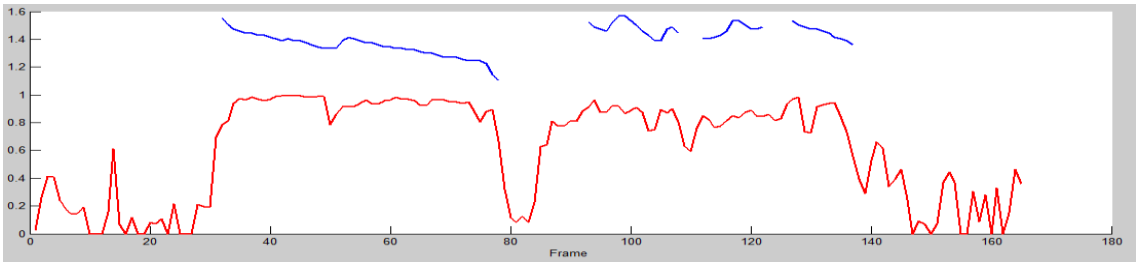
$$pitch = \tau, nếu\ AMDF(\tau) = agr_{min}\{AMDF(k)\}, k = 1, .., T-1 \quad (3.4)$$

3.3.3. Trích chọn NCC và AMDF sử dụng công cụ SNACK

Mục tiêu của luận án là tìm loại đặc trưng thích hợp và cải tiến nâng cao chất lượng đặc trưng đó khi áp dụng mô hình MSD-HMM cho nhận dạng tiếng Việt có thanh điệu, do vậy để tính toán đặc trưng NCC và AMDF luận án sử dụng công cụ mã nguồn mở có sẵn SNACK [Snack 2004]. NCC và AMDF thu được bằng cách sử dụng hàm Pitch của SNACK trên tất cả các câu phát âm trong cả hai bộ dữ liệu huấn luyện và thử nghiệm.

Đường phía trên trong Hình 3-2 mô tả đường đặc tính AMDF, đường phía dưới mô tả đặc tính NCC của câu phát âm "xem ra chữa được bách bệnh" sử dụng công cụ SNACK. So sánh với đường đặc tính NCC cho thấy AMDF bị đứt gãy nhiều hơn so

với NCC. Đặc trưng NCC khá phù hợp với các mô hình liên tục do đường đặc tính của nó khá liên tục và có thể dễ dàng làm trơn đường này bằng các thuật toán đơn giản.



Hình 3-2: Đặc tính AMDF và NCC của câu phát âm "xem ra chữa được bách bệnh"

Sau khi tính toán AMDF và NCC như ở trên, các giá trị này được lấy logarithm. Để tăng thêm thông tin về ngữ cảnh thời gian luận án sử dụng thêm các giá trị độ lệch lân cận delta và double delta như công thức (3.5) và (3.6). Như vậy đặc trưng pitch cuối cùng được sử dụng trong luận án này sẽ là một mảng 3 chiều $F0[3 \times M]$ trong đó *F0[1,i]* là giá trị NCC hoặc AMDF, *F0[2,i]* là delta của *F0[1,i]*, *F0[3,i]* là delta của *F0[2,i]* với *i=0,..,M* (M là kích thước của F0)

$$delta(i) = \sum_{k=1}^{K} |F0(i-k) - F0(i+k)| \qquad (3.5)$$

$$double_{delta} = \sum_{k=1}^{K} |delta(i-k) - delta(i+k)| \qquad (3.6)$$

Trong đó: *F0(i)* là giá trị NCC hoặc AMDF tại khung thứ *i*; *K* là độ dài hàm delta, trong luận án *K* được chọn là 5.

3.4. Tổng quan về mô hình MSD-HMM

Mô hình HMM có thể được sử dụng để mô hình hoá các giá trị liên tục hoặc rời rạc, tuy nhiên HMM không thể mô hình hoá được các loại mẫu chứa cả hai loại này. Chính vì thế nếu một hệ thống nhận dạng tiếng nói muốn sử dụng đặc trưng pitch (NCC hoặc AMDF) thì cần có bước tiền xử lý đặc trưng này để bổ sung các giá trị liên tục vào vùng vô thanh, nơi không tồn tại pitch. Hiện nay gần như tất cả các hệ thống nhận dạng tiếng nói có sử dụng đặc trưng thanh điệu đều thực hiện theo nguyên tắc này. Tuy nhiên phương pháp này đã làm biến đổi đặc trưng thanh điệu, các phương pháp tiền xử lý đã tạo ra các giá trị pitch nhân tạo cho các vùng vô thanh. Và rõ ràng

như vậy không thể đạt được chất lượng nhận dạng tốt nhất, đặc biệt là cho các ngôn ngữ mà thanh điệu ảnh hưởng đến ngữ nghĩa như tiếng Việt. Mô hình MSD-HMM được đề xuất bởi Tokuda là một giải pháp cho vấn đề không liên tục này của pitch. Điểm đặc biệt chính của MSD-HMM là có khả năng mô hình hoá mẫu đầu vào chứa cả hai loại giá trị liên tục và rời rạc. Như vậy mô hình này có thể được áp dụng để mô hình hoá đặc trưng pitch bằng cách tại các vùng hữu thanh ta giữ nguyên giá trị pitch (số thực), còn tại các vùng vô thanh có thể gán một nhãn bất kỳ như "unvoiced". Đặc trưng đầu vào pitch lúc này chứa cả giá trị số và giá trị rời rạc (nhãn unvoiced). Rõ ràng là đặc trưng kiểu này mô tả đúng đắn nhất về đặc tính thanh điệu.

3.4.1. Định nghĩa MSD-HMM

Mô hình MSD-HMM là một loại mô hình cải tiến của HMM dựa trên mô hình MSD. Xét mô hình HMM λ gồm *N* trạng thái được định nghĩa bởi 3 thành phần như sau $\lambda = \{\pi, A, B\}$. Trong đó $\pi = \{\pi_1, \pi_2, .., \pi_N\}$ tập các giá trị xác suất để trạng thái S_i*(i=1,..,N)* là trạng thái đầu. A là một ma trận 2 chiều kích thước $N \times N$, với $A[i, j]$ là xác suất để trạng thái S_i trở thành S_j. $B = \{b_{1k}, b_{2k}, \dots, b_{ik}, \dots, b_{Nk}\}$ là tập xác suất để trạng thái S_i thu được một quan sát o_k, hay còn gọi là hàm xác suất phát tán. Trong nhận dạng tiếng nói mô hình HMM được áp dụng và b_{ik} thường được thay thế bằng các hàm xác suất Gaussian. Nếu cho một vector quan sát $O = \{o_1, o_2, .., o_t, .., o_T\}$ có kích thước *T* thì xác suất của O được tính từ mô hình HMM λ theo công thức (3.7).

$$P(O|\lambda) = \sum_{s \epsilon S} \prod_{t=1}^{T} a_{t,t+1} b_t(o_t) \qquad (3.7)$$

Trong đó: $a_{t-1,t}$ là xác suất để trạng thái s_{t-1} chuyển sang trạng thái s_t. π_{s_0} là xác suất trạng thái đầu của chuỗi trạng thái *S* tương ứng với chuỗi quan sát *O*. $b_t(o_t)$ là hàm xác suất phát tán của trạng thái s_t đối với vector quan sát o_t.

Để nâng cao chất lượng mô hình thì hàm trộn Gaussian (GMM) với *M* thành phần trộn (mixture) thường được sử dụng làm hàm xác suất phát tán. Trong trường hợp này hàm phát tán $b(o_t)$ của vector quan sát đầu vào o_t được viết lại như công thức (3.8).

$$b(o_t) = \sum_{i=1}^{M} w_i \, G(o_t|\mu_i, \Sigma_i) \qquad (3.8)$$

Trong đó: w_i là trọng số của thành phần thứ *i*. $G(o_t|\mu_i\Sigma_i)$là hàm Gaussian đặc trưng bởi hai thành phần là vector trung bình μ_i và ma trận hiệp phương sai Σ_i của vector đầu vào o_t có số chiều là *D*. Hàm này có dạng như công thức (3.9).

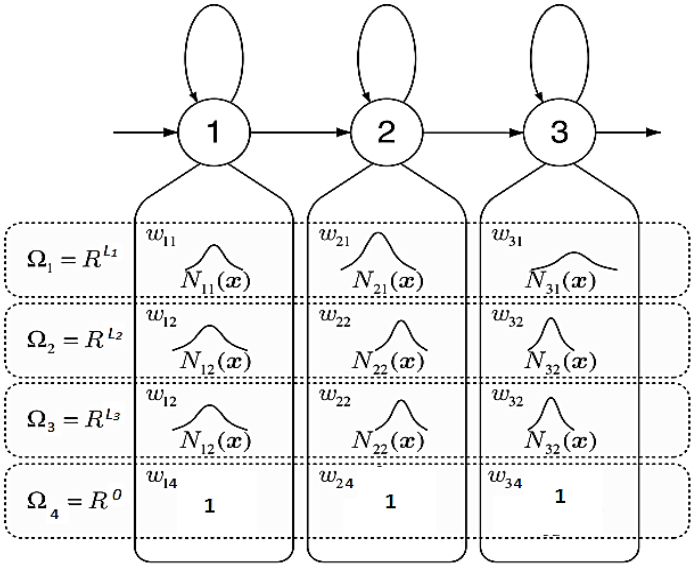
$$G(o_t|\mu_i,\Sigma_i) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} exp\left\{-\frac{1}{2}(o_t-\mu_i)^t\Sigma_i^{-1}(o_t-\mu_i)\right\} \quad (3.9)$$

Hàm $G(o_t|\mu_i\Sigma_i)$ được định nghĩa với biến o_t là vector có giá trị liên tục, hàm này không làm việc hay nói cách khác không được định nghĩa với o_t là biến rời rạc. Đây là lý do tại sao mô hình HMM-GMM thông thường không thể sử dụng được với loại đặc trưng chứa cả giá trị liên tục và rời rạc. Mô hình MSD-HMM [Tokuda 1999] cải tiến lại mô hình HMM để nó có thể làm việc với loại đặc trưng chứa cả hai giá trị liên tục và rời rạc bằng cách giữ nguyên các thành phần giống như mô hình HMM và định nghĩa lại hàm xác suất phát tán dựa trên lý thuyết về mô hình phân bố đa không gian như công thức (3.10).

$$b(o_t) = \sum_{g=1}^{G} sw_g\, G_g(o_t|\pi,\Sigma) \qquad (3.10)$$

Trong đó o_t là vector đầu vào có kích thước thay đổi, mỗi loại kích thước *L* của o_t được gán với một không gian Ω_g có kích thước *L* chiều được đặc trưng bởi trọng số sw_g và hàm Gaussian tương ứng $G_g(o_t|\pi,\Sigma)$. Nếu số chiều *L>0* thì hàm $G_g(o_t|\pi,\Sigma)$ được định nghĩa như công thức (3.9), nếu *L=0* trong trường hợp *o* là biến rời rạc thì $G_g(o_t|\pi,\Sigma)$ được định nghĩa là 1.

Hình 3-3 Minh họa một ví dụ về mô hình MSD-HMM ba trạng thái với số không gian là *G=4*, $\Omega = \{\Omega_1,\Omega_2,\Omega_3,\Omega_4\}$. Trong đó số chiều của Ω_g là L_g với *g=1,2,3,4*. Riêng không gian thứ tư Ω_4 có số chiều $L_4 = 0$, và hàm Gaussian cho không gian này được định nghĩa là 1.



Hình 3-3: Mô hình MSD-HMM 3 trạng thái, 4 không gian(R^l là không gian thực kích thước l chiều, N_{ig} là hàm Gaussian của trang thái S_i trong không gian $\boldsymbol{\Omega_g}$)

Như vậy công thức tính xác suất cho một quan sát *O* với mô hình MSD-HMM mới được viết lại dựa vào ba công thức (3.7), (3.8) và (3.10) như sau.

$$P(O|\lambda) = \sum_{s\epsilon S} \prod_{t=1}^{T} a_{t,t+1} b_t(o_t)$$

$$= \sum_{s\epsilon S} \prod_{t=1}^{T} a_{t,t+1} \sum_{g=1}^{G} sw_g \, G_g(o_t|\pi, \Sigma)$$

$$= \sum_{s\epsilon S} \prod_{t=1}^{T} a_{t,t+1} \sum_{g=1}^{G} sw_g \sum_{i=1}^{M} w_i \, G_g(o_t|\mu_i, \Sigma_i) \qquad (3.11)$$

3.4.2. Ước lượng tham số cho MSD-HMM

Việc huấn luyện hay ước lượng tham số [Tokuda 1999] cho mô hình MSD-HMM λ theo một tập quan sát *O* cho trước được thực hiện tương tự như mô hình HMM.

a) Hàm Q

Giả sử tại thời điểm hiện tại ta đang có mô hình λ', cần ước lượng lại tham số cho nó để thu được mô hình mới λ. Khi đó tham số của mô hình λ được xác định theo hàm phụ Q như công thức (3.12).

$$(\lambda', \lambda) = \sum_{s \in S, g \in G} P(O, s, g|\lambda') log P(O, s, g|\lambda) \qquad (3.12)$$

Trong đó: $s \in S$, $g \in G$ là chuỗi trạng thái tương ứng với quan sát O.

Hàm Q xác định với ba định lý sau:

- *Định lý 1*: nếu $Q(\lambda', \lambda) \geq Q(\lambda', \lambda')$ thì $P(O|\lambda) \geq Q(O|\lambda')$
- *Định lý 2*:Các tham số thuộc về từng không gian Ω_g là độc lập với nhau.
- *Định lý 3*: Các tham số của mô hình λ trong hàm $P(O|\lambda)$ là bộ tham số tới hạn nếu và chỉ nếu nó là tới hạn trong hàm Q.

b) Tối ưu giá trị hàm Q

Nhiệm vụ của bước này là đi xác định các tham số cho mô hình λ để tối ưu giá trị hàm $Q(\lambda', \lambda)$ với O và λ cho trước. Hàm $Q(\lambda', \lambda)$ ở công thức (3.12) được viết lại như sau:

$$Q(\lambda', \lambda) = \sum_{s \in S, g \in G} P(O, s, g|\lambda') log a_{s_0 s_1}$$

$$+ \sum_{s \in S, g \in G} P(O, s, g|\lambda') \sum_{t=1}^{T-1} log\, a_{s_t s_{t+1}}$$

$$+ \sum_{s \in S, g \in G} P(O, s, g|\lambda') \sum_{t=1}^{T-1} log\, sw_{s_t g_t}$$

$$+ \sum_{s \in S, g \in G} P(O, s, g|\lambda') \sum_{t=1}^{T-1} log\, g\left(o_t \middle| \pi_{s_t g_t}, \Sigma_{s_t g_t}\right) \qquad (3.13)$$

Trong đó các số hạng thứ nhất, hai và bốn được ước lượng tương tự như quá trình ước lượng tham số cho mô hình HMM. Trong mô hình MSD-HMM ta cần ước lượng thêm các trọng số sw_g cho không gian $\Omega_g \in \Omega$ thông qua số hạng thứ ba trong công thức (3.13). Cụ thể từng thành phần trong (3.13) được viết lại như sau:

- *Số hạng thứ nhất của (3.13)*

$$\sum_{s\in S, g\in G} P(O,s,g|\lambda')log a_{s_0 s_1}$$

$$= \sum_{i=1}^{N}\sum_{g\in G} P(O,s_i=i,g|\lambda')log a_{s_0 i}$$

$$= \sum_{i=1}^{N}\sum_{g\in G} P(O,s_i=i,g|\lambda')log \pi_i \qquad (3.14)$$

- *Số hạng thứ 2 của (3.13)*

$$\sum_{s\in S, g\in G} P(O,s,g|\lambda')\sum_{t=1}^{T-1} log\, a_{s_t s_{t+1}}$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{t=1}^{T-1}\sum_{g\in G} P(O,s,g|\lambda')log a_{ij} \qquad (3.15)$$

- *Số hạng thứ 3 của (3.13)*

$$\sum_{s\in S, g\in G} P(O,s,g|\lambda')\sum_{t=1}^{T-1} log\, sw_{s_t g_t}$$

$$= \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{g\in G} P(O,s,g|\lambda')log sw_{ig} \qquad (3.16)$$

- *Số hạng thứ 4 của (3.13)*

$$\sum_{s\in S,g\in G} P(O,s,g|\lambda') \sum_{t=1}^{T-1} log\ g\left(o_t|\pi_{s_t g_t}, \Sigma_{s_t g_t}\right)$$

$$\sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{g\in G} P(O,s,g|\lambda') log\ g\left(o_t|\pi_{s_t g_t}, \Sigma_{s_t g_t}\right) \qquad (3.17)$$

Quá trình huấn luyện sẽ đi xác định các giá trị π_i, a_{ij}, sw_{ig}, và $g\left(o_t|\pi_{s_t g_t}, \Sigma_{s_t g_t}\right)$ trong các công thức (3.14, 3.15, 3.17) sao cho giá trị của hàm $Q(\lambda', \lambda)$ đạt giá trị lớn nhất thỏa mãn các điều kiện: $\sum_{i=1}^{N} \pi_i = 1$, $\sum_{j=1}^{N} a_{ij} = 1$, $\sum_{g}^{G} sw_g = 1$.

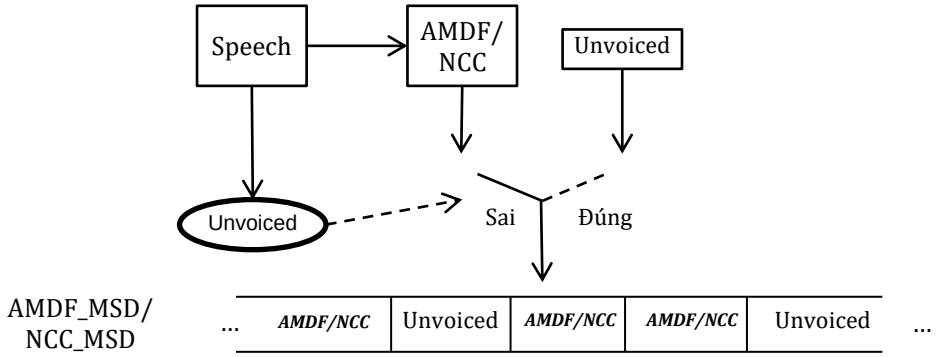
3.5. Các nghiên cứu đã công bố về áp dụng MSD-HMM trong nhận dạng tiếng nói

Mô hình MSD-HMM ban đầu được đề xuất để áp dụng cho các mô hình âm học trong tổng hợp tiếng nói. Các thử nghiệm tổng hợp tiếng nói trên các ngôn ngữ như tiếng Thái lan [Suphattharachai 2011], tiếng Nhật [Takashi 2002], tiếng Mandarin [Y. a. Qian 2006],… đã cho thấy MSD-HMM hiệu quả hơn so với mô hình HMM. Do MSD-HMM có thể mô hình hóa được đặc trưng thanh điệu đứt gãy đúng như bản chất tự nhiên của nó nên các mô hình tổng hợp tiếng nói dựa trên MSD-HMM thể hiện được ngữ điệu tự nhiên của một phát âm. Từ ưu điểm và hiệu quả này MSD-HMM cũng đã được nghiên cứu áp dụng cho nhận dạng tiếng nói cho các ngôn ngữ có thanh điệu. Tính đến hiện tại MSD-HMM đã được nghiên cứu áp dụng cho hai ngôn ngữ là tiếng Quan thoại (Mandarin) của Trung Quốc và tiếng Ba tư. Nghiên cứu của tác giả Qian [Y. a. Qian 2009] là nghiên cứu đầu tiên về việc áp dụng MSD-HMM cho nhận dạng tiếng Mandarin phát âm liên tục từ vựng lớn. Trong nghiên cứu này tác giả đề xuất áp dụng mô hình MSD-HMM với hai luồng đặc trưng đầu vào độc lập (2-streams MSD-HMM model) để mô hình hóa các âm vị có thông tin thanh điệu. Luồng thứ nhất sử dụng đặc trưng ngữ âm MFCC, luồng thứ hai sử dụng đặc trưng thanh điệu Pitch. Tác giả đã tiến hành các thử nghiệm trên ba loại dữ liệu là tiếng nói đọc, tiếng nói ngẫu nhiên và tiếng nói của các chữ số. Kết quả cho thấy mô hình MSD-HMM đã tăng chất lượng nhận dạng lên tương ứng với ba loại dữ liệu trên lần lượt là 21%, 8.4% và 17.4% tương đối so với hệ thống sử dụng mô hình HMM. Đối với tiếng Ba Tư [Fatemeh 2013] nhóm tác giả Fatemeh đã áp dụng mô hình MSD-HMM để mô hình hóa và nhận dạng ngữ điệu của một câu phát âm ở mức Từ, bước này làm bước tiền xử lý để lấy thông tin về ngữ điệu cho câu phát âm trong quá trình huấn luyện mô hình

tổng hợp tiếng nói. Độ chính xác nhận dạng cụm từ theo ngữ điệu đạt 80.08%, và độ chính xác gán nhãn cụm từ theo thanh điệu là 73.5%.

3.6. Chuẩn hóa đặc trưng AMDF và NCC cho mô hình MSD-HMM

Để đánh giá và so sánh hiệu quả giữa mô hình HMM và mô hình MSD-HMM, luận án sẽ tiến hành làm các thử nghiệm tương đồng trên cả hai loại mô hình này. HMM và MSD-HMM sẽ được sử dụng để mô hình hóa cho cùng một tập âm vị có thông tin thanh điệu như đã trình bày ở Chương 2. Tuy nhiên do mô hình HMM chỉ mô hình hoá được đặc trưng chứa giá trị liên tục (hoặc chỉ chứa giá trị rời rạc), trong khi MSD-HMM thì có thể mô hình hóa được đặc trưng chứa cả hai loại giá trị. Vì thế đặc trưng thanh điệu đầu vào cho mô hình HMM trong luận án này sẽ là AMDF và NCC được tính toán như công thức (3.1) và (3.3). Còn đặc trưng thanh điệu cho mô hình MSD-HMM sẽ được cải tiến lại từ AMDF và NCC ban đầu theo công thức (3.7) và (3.8) để thu được hai đặc trưng tương ứng mới tương thích với mô hình MSD-HMM ký hiệu là AMDF_MSD và NCC_MSD. Ý tưởng chính của phương pháp này là thay thế các giá trị pitch tính được từ phương pháp AMDF và NCC bằng giá trị "unvoiced" cho các vector thuộc vùng vô thanh. Vùng vô thanh xác định được thông qua phương pháp so sánh ngưỡng năng lượng [Jurafsky 2008] như công thức (3.20).



Hình 3-4: Quá trình trích chọn đặc trưng thanh điệu cho HMM và MSD-HMM

Giả sử tín hiệu tiếng nói đầu vào *X* sau khi phân tách thành các khung rời rạc ta thu được $X' = \{x_k\}$ trong đó x_k là khung tín hiệu đầu vào thứ *k*, với *k=1,..,N* (N là tổng số khung tín hiệu sau khi được phân tách từ *X*).

$$AMDF\_MSD_k = \begin{cases} AMDF_k, nếu\ UV(x_k) = 1 \\ \text{"unvoiced"}, nếu\ UV(x_k) = 0 \end{cases} \quad (3.18)$$

$$NCC\_MSD_k = \begin{cases} NCC_k, nếu\ UV(x_i) = 1 \\ \text{"unvoiced"}, nếu\ UV(x_k) = 0 \end{cases} \quad (3.19)$$

Trong đó NCC_i và $AMDF_i$ được tính theo công thức (3.7) và (3.8), và:

$$UV(x_k) = \begin{cases} 0 = \text{voice}, e(x_k) \geq \varphi \\ 1 = unvoiced, nếu\ e(x_k) < \varphi \end{cases} \quad (3.20)$$

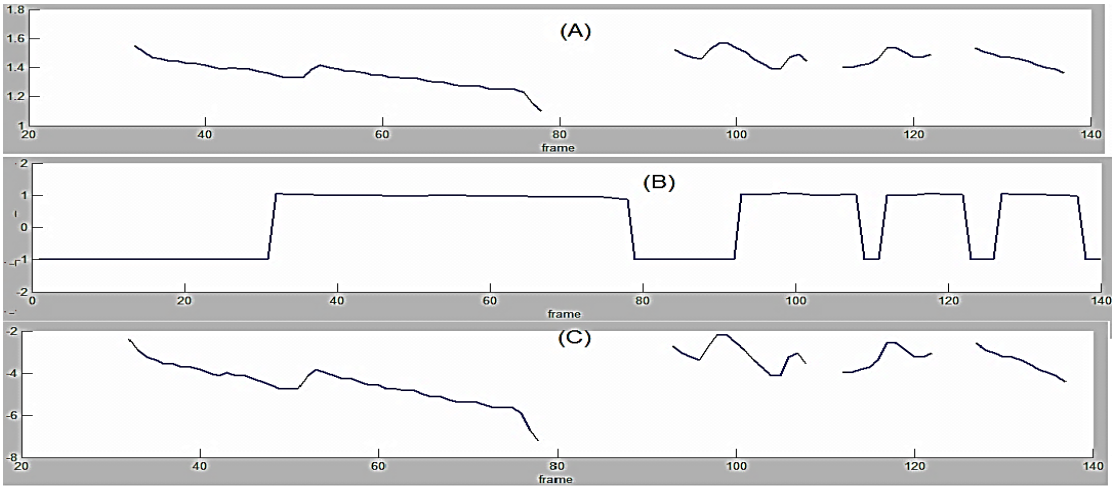
Trong đó $e(x_k) = \sum_{l=11}^{L} {x_k}_l^2$ là hàm năng lượng với L là độ dài của x_k, φ là hệ số. Cả hai loại đặc trưng này sau đó được áp dụng phương pháp chuẩn hóa như công thức (3.21). Quy trình tính toán AMDF_MSD và NCC_MSD được minh họa như sơ đồ ở Hình 3-4.

$$f0_m = \frac{\log(F0_m) - mean(F0)}{Dev(F0)},\ m=0,1,...,M \quad (3.21)$$

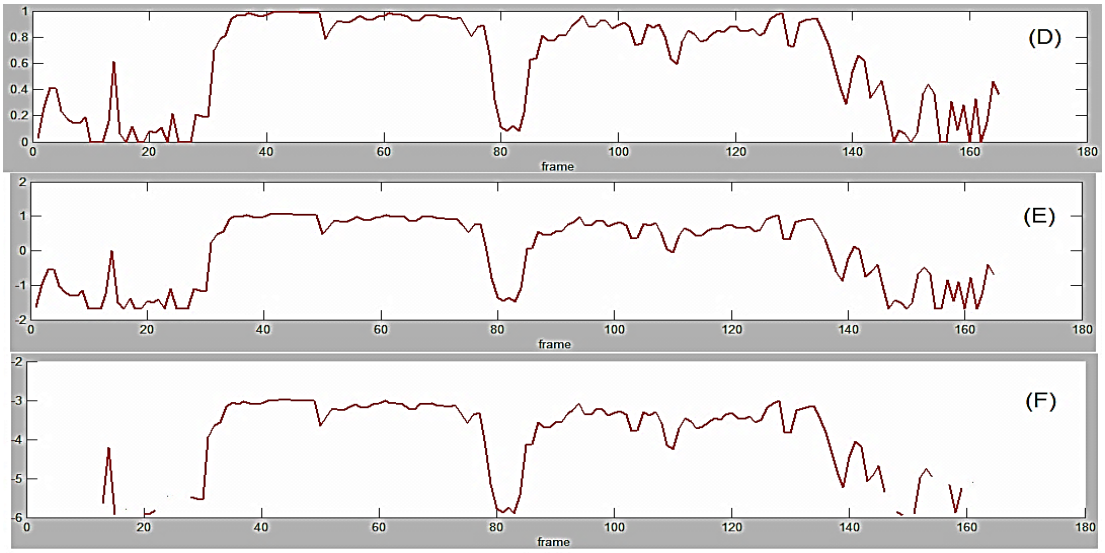
$$Mean(F0) = \frac{1}{M}\sum_{m=0}^{M-1} F0_m$$

$$Dev(F0) = \sqrt{\frac{1}{N}\sum_{m-0}^{M-1}(F0_m - Mean(F0))^2}$$

Trong đó: *F0* là đặc trưng AMDF/NCC của cả câu phát âm hiện thời có chiều dài *M*. $F0_m$ là giá trị AMDF/PLP của khung thứ *m* trong câu phát âm hiện thời. Hình 3-5 và Hình 3-6 mô tả đường đặc tính của AMDF-HMM, NCC-HMM, AMDF-MSD, và NCC-MSD của câu phát âm "*xem ra chữa được bách bệnh*".



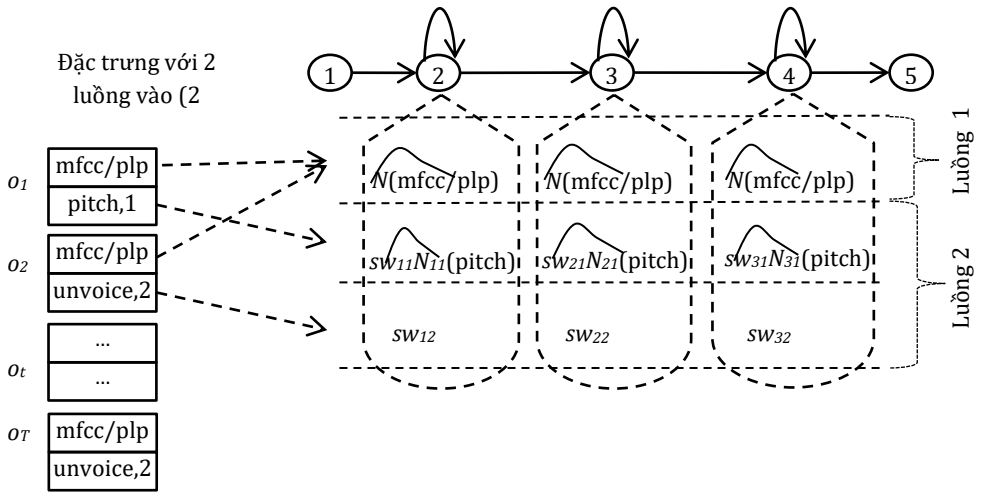
Hình 3-5: Đặc tính AMDF sau chuẩn hoá
của câu phát âm "chắc chữa được bách bệnh"
(Hình A đặc tính AMDF, B đặc tính AMDF-HMM, C đặc tính AMDF-MSD)



Hình 3-6: Đặc trình NCC sau chuẩn hoá
của câu phát âm "chắc chữa được bách bệnh"
(Hình D đặc tính NCC, E đặc tính NCC-HMM, F đặc tính NCC-MSD)

3.7. Áp dụng mô hình MSD-HMM cho nhận dạng tiếng Việt có thanh điệu

Trong phạm vi nghiên cứu này luận án đề xuất sử dụng mô hình MSD-HMM 5 trạng thái kiểu trái phải (left-right) với nhiều hơn 1 luồng dữ liệu vào cho nhận dạng tiếng Việt với mục đích kết hợp cả đặc trưng ngữ âm và đặc trưng thanh điệu vào một mô hình. Trong đó luồng thứ nhất dành cho đặc trưng ngữ âm (MFCC/PLP). Luồng này sử dụng một không gian số thực duy nhất có số chiều đúng bằng kích thước của vector đầu vào (do loại đặc trưng này là liên tục). Từ luồng thứ hai sẽ được sử dụng cho đặc trưng pitch. Ở các luồng này sẽ sử dụng hai không gian $\Omega = \{\Omega_1, \Omega_2\}$, trong đó Ω_1 là không gian số thực có số chiều là d tương ứng với kích thước của vector đặc trưng pitch đầu vào. Ω_2 chỉ có một giá trị duy nhất là nhãn "unvoiced". Khi đó mỗi một vector đặc trưng pitch đầu vào sẽ có hai thành phần $o_{pitch} = \{x, g\}$, nếu x là một số thực ($x \in R^d$) thì $g=1$ để chỉ $x \in \Omega_1$là giá trị pitch, nếu $x = "unvoiced"$ thì $g=2$ để chỉ $x \in \Omega_2$, giá trị này thể hiện khung hiện thời không tồn tại thanh điệu hay là vùng vô thanh. Hình 3-7 minh hoạ ví dụ mô hình MSD-HMM 5 trạng thái với 2 luồng dữ liệu vào như mô tả ở trên. Với mô hình này các đặc trưng pitch được trích chọn từ các thuật toán đã đề xuất có thể được mô hình hoá trực tiếp mà không cần bổ sung thêm giá trị tại các vùng vô thanh (vùng đứt gãy). Mô hình này sẽ được sử dụng để mô hình hóa các âm vị có thanh điệu của tiếng Việt được tạo ra bởi thuật toán VN-G2P.

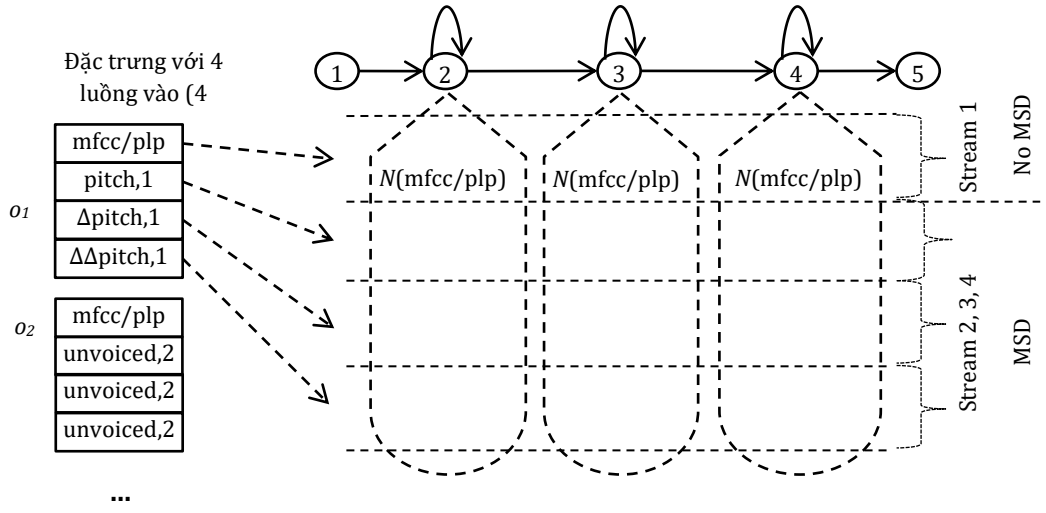


Hình 3-7: Mô hình MSD-HMM left-right 5 trạng thái, 2 luồng

3.8. Cài đặt thử nghiệm và kết quả

Qua các thử nghiệm ở Chương 2 cho thấy hệ thống sử dụng mô hình thanh điệu với từ điển Tona-Dict luôn cho chất lượng nhận dạng tốt hơn với từ điển NonTonal-Dict. Vì thế tất các thử nghiệm ở bước này sẽ chỉ được cài đặt trên từ điển Tonal-Dict.

Mô hình MSD-HMM được sử dụng trong thử nghiệm này có cấu hình gồm 5 trạng thái theo kiểu trái-phải (left-right) với 4 luồng đầu vào. Luồng thứ nhất được sử dụng để mô hình hoá cho đặc trưng ngữ âm MFCC/PLP. Luồng thứ 2, 3, 4 được áp dụng MSD với hai không gian. Không gian thứ nhất có số chiều là 1 để mô hình hóa giá trị đặc trưng thanh điệu liên tục. Không gian thứ hai chứa một giá trị duy nhất là ký hiệu "unvoiced". Luồng thứ 2 được sử dụng để mô hình hoá đặc trưng Pitch (NCC/AMDF), luồng thứ 3 mô hình hoá đặc trưng delta của Pitch, luồng thứ 4 mô hình hoá đặc trưng double delta của Pitch. Hình 3-8 minh hoạ cấu hình của mô hình MSD-HMM được sử dụng trong thí nghiệm này.



Hình 3-8: Mô hình MSD-HMM 5 trạng thái, 4 luồng đầu vào

3.8.1. Dữ liệu, mô hình ngôn ngữ, từ điển

Dữ liệu huấn luyện và thử nghiệm là VOV, VOV-test. Mô hình ngôn ngữ sử dụng mô hình VOV-BiGram-LM lấy từ hệ thống cơ sở. Từ điển sử dụng tập âm vị có thanh điệu Tonal-Dict với 154 âm vị. Kích thước từ điển là 4908 từ.

3.8.2. Trích chọn đặc trưng

1) Đặc trưng ngữ âm

Hai loại đặc trưng ngữ âm được sử dụng trong thử nghiệm này là MFCC và PLP tương tự như các thử nghiệm ở Chương 2. Mỗi vector chứa 39 thành phần (bao gồm 13 MFCC/PLP, 13 ΔMFCC/PLP, và 13 ΔΔMFCC/PLP). Đặc trưng ngữ âm này sẽ được đưa vào luồng thứ nhất của mô hình MSD-HMM.

2) Đặc trưng thanh điệu

Để đánh giá hiệu quả của mô hình MSD-HMM so với mô hình HMM, các mô hình này sẽ được thử nghiệm trên cùng điều kiện dữ liệu, từ điển, mô hình ngôn ngữ và đặc trưng ngữ âm. Sự khác biệt là đặc trưng thanh điệu. Mô hình HMM sẽ sử dụng AMDF/NCC, còn mô hình MSD-HMM sẽ sử dụng AMDF_MSD/NCC_MSD.

3.8.3. Thử nghiệm mô hình HMM

Bốn hệ thống được xây dựng sử dụng mô hình HMM như đã được sử dụng cho hệ thống cơ sở. Các hệ thống này được huấn luyện trên bốn loại đặc trưng khác nhau tương ứng là MFCC+AMDF, MFCC+NCC, PLP+AMDF, và PLP+NCC. Mục tiêu của thử nghiệm này là đánh giá hiệu quả của hệ thống nhận dạng khi tích hợp thêm đặc trưng thanh điệu vào đặc trưng ngữ âm (MFCC/PLP). Đồng thời tìm ra loại đặc

trưng thanh điệu nào trong hai loại AMDF, NCC phù hợp với mô hình HMM thông thường. Tất các mô hình âm học được huấn luyện theo quy trình giống như hệ thống cơ sở đã trình bày ở mục 2.8. Các hệ thống được huấn luyện ở mức tri-phone với 2179 âm buộc, mỗi state sử dụng 16 thành phần trộn Gaussian. Kết quả thử nghiệm trên tập VOV-Test được trình bày ở Bảng 3-1.

Bảng 3-1: Kết quả thử nghiệm Pitch và MFCC/PLP với HMM

TT	Hệ thống	Đặc trưng	Từ điển	ACC (%)
1	Baseline	MFCC		77.70
2	HMM-3	PLP+AMDF	Tonal-Dict	74.34
3	HMM-4	MFCC+AMDF		76.10
4	HMM-5	PLP+NCC		79.09
5	**HMM-6**	**MFCC+NCC**		**80.26(+2,56)**

Từ kết quả cho thấy việc tích hợp thêm đặc trưng thanh điệu làm tăng chất lượng nhận dạng lên 2.56% tuyệt đối so với hệ thống cơ sở trên hệ thống sử dụng đặc trưng MFCC+NCC (HMM-6).

3.8.4. Thử nghiệm mô hình MSD-HMM

a) Huấn luyện hệ thống

Đầu tiên bốn hệ thống sử dụng mô hình MSD-HMM được huấn luyện trên bốn tổ hợp đặc trưng tương ứng là MFCC+AMDF-MSD, MFCC+NCC-MSD, PLP+AMDF-MSD, PLP+NCC-MSD. Các bước huấn luyện tương tự như mô hình Baseline, mỗi hệ thống cũng đều có 2179 âm buộc, mỗi state sử dụng 16 thành phần trộn Gaussian. Kết quả thử nghiệm trên tập VOV-stest như ở Bảng 3-2.

Bảng 3-2: Kết quả thử nghiệm mô hình MSD-HMM

TT	Hệ thống	Đặc trưng	Từ điển	ACC (%)
1	MSD-HMM-1	PLP+NCC_MSD	Tonal-Dict	76.47
2	MSD-HMM-2	PLP+AMDF_MSD		79.78
3	MSD-HMM-3	MFCC+NCC_MSD		77.64
4	MSD-HMM-4	MFCC+AMDF_MSD		80.37
5	MSD-HMM-5	PLP+NCC+AMDF_MSD		79.71
6	**MSD-HMM-6**	**MFCC+NCC+AMDF_MSD**		**80.80**

Dựa theo kết quả ở Bảng 3-1 và Bảng 3-2 cho thấy đặc trưng thanh điệu NCC thích hợp với mô hình HMM, trong khi đó AMDF_MSD lại thích hợp với mô hình MSD-HMM. Vì vậy luận án thử nghiệm thêm hai hệ thống nữa bằng việc kết hợp ba loại đặc trưng MFCC/PLP, NCC và AMDF_MSD (MFCC/PLP+NCC+AMDF_MSD). Trong đó MFCC/PLP+NCC sẽ được đưa vào luồng số 1 của mô hình MSD-HMM như đặc trưng chỉ chứa các giá trị liên tục, luồng này không được áp dụng MSD. AMDF_MSD sẽ được đưa vào các luồng còn lại 2, 3, 4 của mô hình MSD-HMM và có áp dụng MSD. Kết quả thử nghiệm trình bày hai dòng 5 và 6 trong Bảng 3-2.Qua các kết quả thử nghiệm cho thấy mô hình MSD-HMM cho chất lượng nhận dạng tốt hơn mô hình HMM khi được tích hợp cả đặc trưng thanh điệu là 0.54% tuyệt đối (MSD-HMM-6 trong Bảng 3-2 so với HMM-6 trong Bảng 3-1). Và tốt hơn 3.1% tuyệt đối so với hệ thống cơ sở (Baseline) không sử dụng đặc trưng thanh điệu.

3.9. Kết luận chương

Từ kết quả thí nghiệm luận án dẫn đến các kết luận như sau:

1) ***Mô hình MSD-HMM có hiệu quả với ngôn ngữ tiếng Việt***: Cụ thể hệ thống sử dụng mô hình MSD-HMM cho kết quả tốt hơn mô hình HMM là 0.54% (hệ thống MSD-HMM-6 so với HMM-6), tốt hơn 3.1% tuyệt đối (15% tương đối) so với hệ thống cơ sở (bộ dữ liệu 20 giờ huấn luyện, tập thử nghiệm 2300 câu). Trong khi đó với tiếng Mandarine [Y. a. Qian 2009] mô hình MSD-HMM đã tăng được 17% tương đối trên bộ dữ liệu 100 giờ huấn luyện và tập thử nghiệm 1600 câu. Rõ ràng

với mô hình mà luận án đã đề xuất áp dụng đã hiệu quả và thành công cho tiếng Việt.

2) ***Đặc trưng thanh điệu dựa trên phương pháp NCC thích hợp với mô hình HMM, đặc trưng thanh điệu dựa trên phương pháp AMDF thích hợp với mô hình MSD-HMM:*** Do đặc trưng dựa trên phương pháp AMDF bị đứt gãy nhiều hơn so với NCC, vì thế tại các đoạn đứt gãy này sẽ tạo nhiều mẫu huấn luyện hơn cho không gian thứ 2 trong mô hình MSD-HMM.

3.10. Các bài báo đã công bố liên quan đến nội dung của chương

1. **Nguyen Van Huy**, Luong Chi Mai, Vu Tat Thang, Do Quoc Truong, *Vietnamese recognition using tonal phoneme based on multi space distribution*, Journal of Computer Science and Cybernetics, Vietnam, ISSN 1813-9663, Vol 30, No 1, Jan-2014.

Chương 4: Tăng cường đặc trưng ngữ âm sử dụng mạng nơron

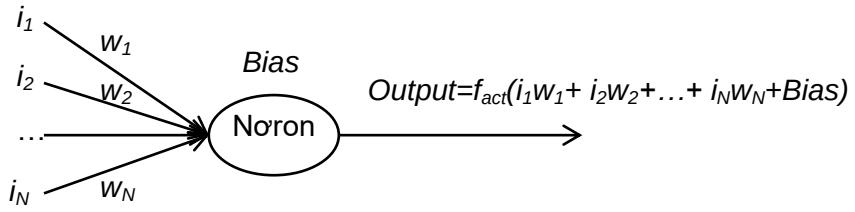
4.1. Tóm tắt chương

Mục tiêu của chương là xây dựng mô hình tăng cường chất lượng cho đặc trưng ngữ âm bằng mạng nơron cho tiếng Việt. Đây là một cách tiếp cận khá mới đã được áp dụng thành công cho tiếng Anh, Đức,…

Nội dung chính của chương bao gồm: Trình bày phương pháp trích chọn đặc trưng Bottleneck sử dụng mạng nơron và phương pháp áp dụng cho nhận dạng tiếng Việt. Bao gồm quy trình gán nhãn dữ liệu, huấn luyện mạng MLP, các bước tính toán đặc trưng Bottleneck, các bước tối ưu đặc trưng Bottleneck; Trình bày các thử nghiệm và kết quả cho nhận dạng tiếng Việt.

4.2. Tổng quan về mạng nơron MLP (Multilayer Perceptron)

Mạng nơron MLP (MultiLayer Perceptron) [Đức 2003] [Kriesel 2005] là một cấu trúc mạng gồm có một lớp vào (input), một lớp ra (output) và một hoặc nhiều lớp ẩn (hidden). Vector đầu vào sẽ được đưa qua lớp vào (input) của mạng và sau đó các tính toán được thực hiện lan truyền thẳng (feed-forward) từ lớp vào input sang các lớp ẩn và kết thúc ở lớp ra output. Hàm kích hoạt kết hợp với các nút ẩn hay các nút output có thể là hàm tuyến tính hay phi tuyến và có thể khác nhau giữa các nút. Hình 4-1 mô tả các thành phần cơ bản của một nút mạng. Hình 4-2 mô tả cấu trúc của một mạng MLP có 3 lớp (1 lớp đầu vào, 1 lớp ẩn và 1 lớp ra).



Hình 4-1: Cấu trúc cơ bản của một nút mạng

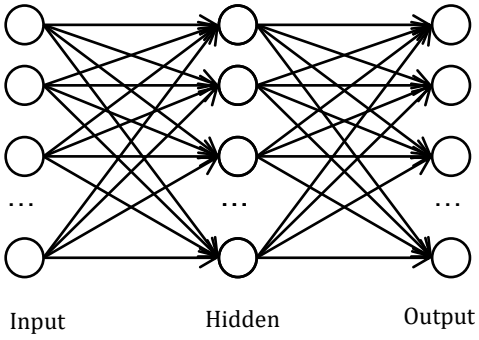
Trong đó: i_j, $j=1..N$ là giá trị đầu vào hoặc là giá trị của hàm kích hoạt từ lớp trước. w_j, $j=1..N$ là trọng số của kết nối từ nút j ở lớp trước đến nút hiện tại. *Bias* là hệ số. *Output* là giá trị đầu ra của hàm kích hoạt của nút hiện tại, giá trị này sẽ được lan truyền như là đầu vào cho các nút ở lớp kế tiếp. f_{act} là hàm kích hoạt.

Xét một mạng MLP có N lớp với kích thước của các lớp tương ứng là S_1,..,S_i,..,S_N. (Trong đó lớp đầu vào là S_1 và lớp đầu ra là S_N). Gọi giá trị kích hoạt của một nút j trong lớp thứ i là $A_{i,j}$, trọng số của liên kết giữa nó với nút thứ k trong lớp phía

trước *i-1* là $W_{i,j,k}$, và trọng số của nút 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:

$$A_{i,j} = f_{act_i}(\sum_{k=1}^{S_{i-1}} A_{i-1,k} * W_{i,j,k} + B_{i,j}) \qquad (4.1)$$

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



Hình 4-2: Mô hình mạng MLP ba lớp

Xét một tập mẫu đầu vào $\{X,Y\}$ trong đó $X=\{X_1,..,X_t,..,X_T\}$ là giá trị đầu vào, $Y=\{Y_1,..,Y_t,..,Y_T\}$ 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:

$$E = \frac{1}{2} * \sum_{t=1}^{T} \sum_{k=1}^{S_N} (Y_{t,k} - \hat{Y}_{t,k})^2 \qquad (4.2)$$

Trong đó: S_N là kích thước lớp đầu ra. $Y_{t,k}$ là giá trị mong muốn tại nút 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 nút thứ k ở lớp đầu ra đối với vector đầu vào X_t.

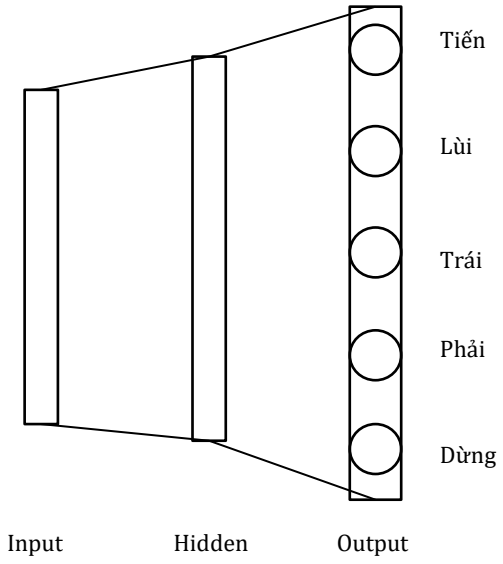
Như vậy mục tiêu của bước huấn luyện mạng là tối thiểu giá trị *E* trong công thức (4.2). Một trong các phương pháp huấn luyện phổ biến được sử dụng trong huấn luyện mạng MLP là phương pháp lan truyền ngược (Backpropagation). Ý tưởng chính của phương pháp tối thiểu giá trị *E* bằng cách dùng chính *E* để xác định lại các giá trị trong số $W_{i,j,k}$ trong công thức (4.1). Quá trình tính toán lại $W_{i,j,k}$ được thực hiện ngược lại từ lớp thứ *N* đến lớp thứ 2 của mạng theo công thức sau:

$$W_{i,j,k}^{q} = W_{i,j,k}^{q-1} + \Delta W_{i,j,k}^{q} \qquad (4.3)$$

$$\Delta W_{i,j,k}^{q} = -\alpha \frac{dE^{q-1}}{dW_{i,j,k}^{q-1}}$$

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

4.3. Ứng dụng mạng nơron trong nhận dạng tiếng nói



Hình 4-3: Mô hình MLP 3 lớp ứng dụng trong điều khiển

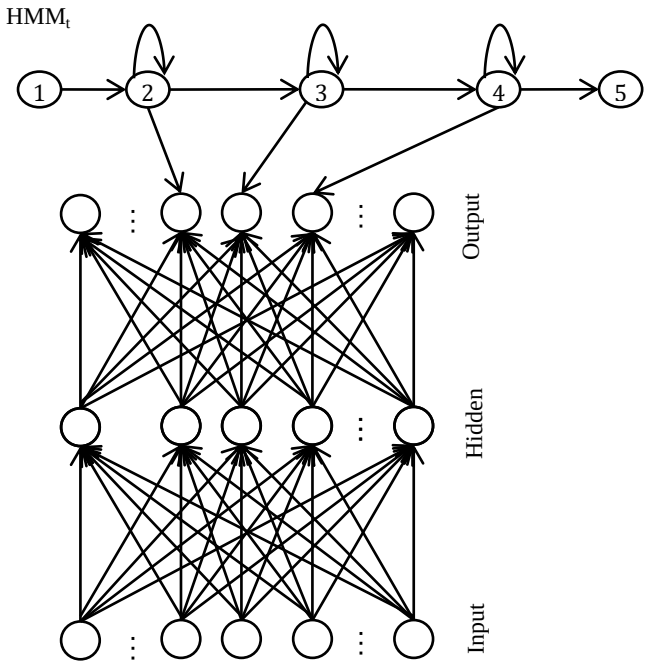
Có hai cách tiếp cận chính trong việc áp dụng mạng nơron cho nhận dạng tiếng nói. Cách tiếp cận thứ nhất là sử dụng mạng nơron như một mô hình âm học có chức năng phân lớp hay nhận dạng mẫu đầu vào. Cách tiếp cận này thường được sử dụng trong các hệ thống nhận dạng với từ vựng nhỏ như các hệ thống điều 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. Hình 4-3 mô tả một mạng

MLP ba lớp 5 đầu ra để nhận dạng 5 khẩu lệnh trong bài toán điều khiển robot di chuyển.

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 nơron thay vì là hàm GMM như cách truyền thống. Hình 4-4 mô tả mô hình lai ghép HMM với một mạng MLP ba lớp. Trong trường hợp này hàm tính giá trị xác suất (2.10) của một quan sát đầu vào *O* với mô hình HMM thứ *t* trong hệ thống được viết lại như sau:

$$P(Q|O,\lambda_t) = \sum_{q_t}^{Q} \pi_{tk} a_{t_{k-1}t_k}, f_{act_{q_t}} k = 1..N \qquad (4.4)$$

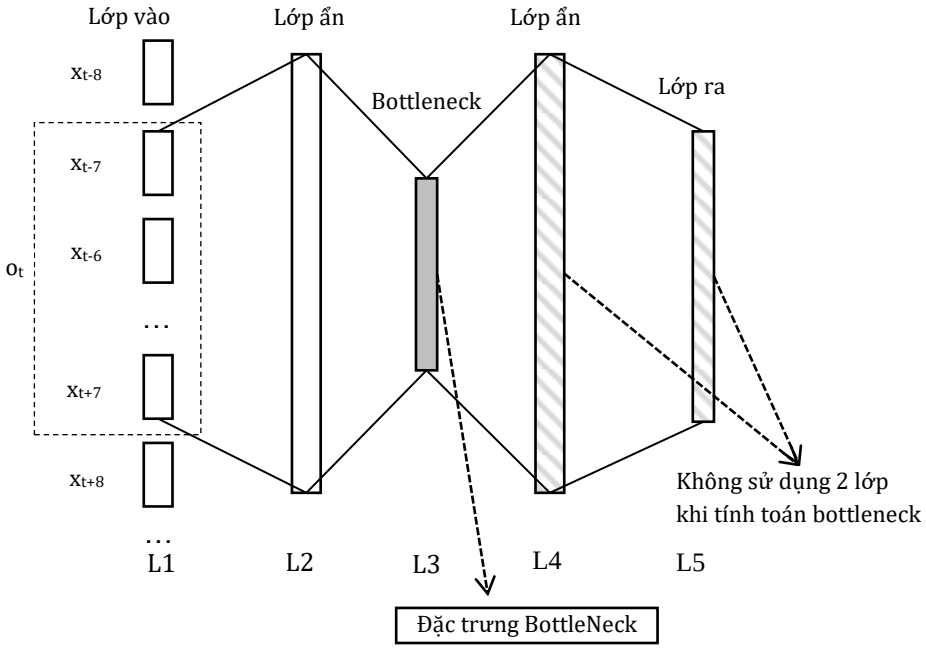
Trong đó: $f_{act_{q_t}}$ là giá trị hàm kích hoạt ở lớp ra tại nút mạng tương ứng với trạng thái q_t của mô hình HMM_t.



Hình 4-4: Mô hình lai ghép HMM-NN

4.4. Trích chọn đặc trưng Bottleneck sử dụng mạng MLP

4.4.1. Tổng quan về đặc trưng Bottleneck



Hình 4-5: Mô hình MLP để trích chọn đặc trưng Bottleneck

Phương pháp trích chọn các đặc trưng của tiếng nói sử dụng mạng nơron đang trở thành một phần quan trọng trong hệ thống nhận dạng tiếng nói [Janin 2006], phương pháp này nhằm tận dụng ưu điểm phân lớp của mạng nơron đồng thời khắc phục một trong các nhược điểm của mô hình Markov ẩn (HMM– Hidden Markov Model), mô hình HMM không mô hình hóa được các đặc tính phụ thuộc thời gian của tín hiệu tiếng nói do HMM giả thiết rằng mỗi trạng thái hiện tại chỉ phụ thuộc vào trạng thái ngay trước nó [Juang 1991] [Gales 2007]. Đã có nhiều phương pháp được đưa ra nhằm khắc phục nhược điểm trên của HMM, một phương pháp được sử dụng phổ biến là bổ sung thêm các vector đặc trưng lân cận với vector đặc trưng đang xét tại thời điểm *t*, tức là tổ hợp nhiều hơn một khung dữ liệu (frame) để đưa vào huấn luyện HMM tại thời điểm *t*, ta gọi đó là cửa sổ khung (frame windows). Ví dụ tại thời điểm *t* nếu ta dùng frame window là 7 thì một quan sát o_t sẽ là tổ hợp của 7 frames liên tiếp từ *(t-3)* đến *(t+3)*, $o_t=\{x_{t-3}, x_{t-2}, x_{t-1}, x_t, x_{t+1}, x_{t+2}, x_{t+3}\}$ với x_t là vector đặc trưng của tiếng nói tại thời điểm *t*. Tuy nhiên phương pháp này làm gia tăng kích thước của o_t, dẫn đến làm tăng kích thước của mô hình GMM. Một phương pháp vừa làm giảm kích thước của o_t đồng thời nâng cap chất lượng o_t đó là sử dụng mạng nơron. Bằng cách này các vector đầu vào o_t được đưa qua một mạng MLP đặc biệt đã được huấn luyện để tách những thông tin quan trọng và nén các thông tin này tạo ra một đặc trưng mới o'_t ở lớp ra (output) của mạng. Khi đó kích thước của o'_t có thể thiết lập được thông qua việc

thiết lập kích thước lớp output. Kích thước của o'_t thường nhỏ hơn rất nhiều so với o_t mà vẫn đảm bảo chứa đủ các thông tin quan trọng bao gồm cả thông tin về ngữ cảnh thời gian. Qua nhiều nghiên cứu đã cho thấy việc sử dụng o'_t như là một đầu vào cho mô hình HMM làm tăng đáng kể chất lượng nhận dạng của hệ thống.

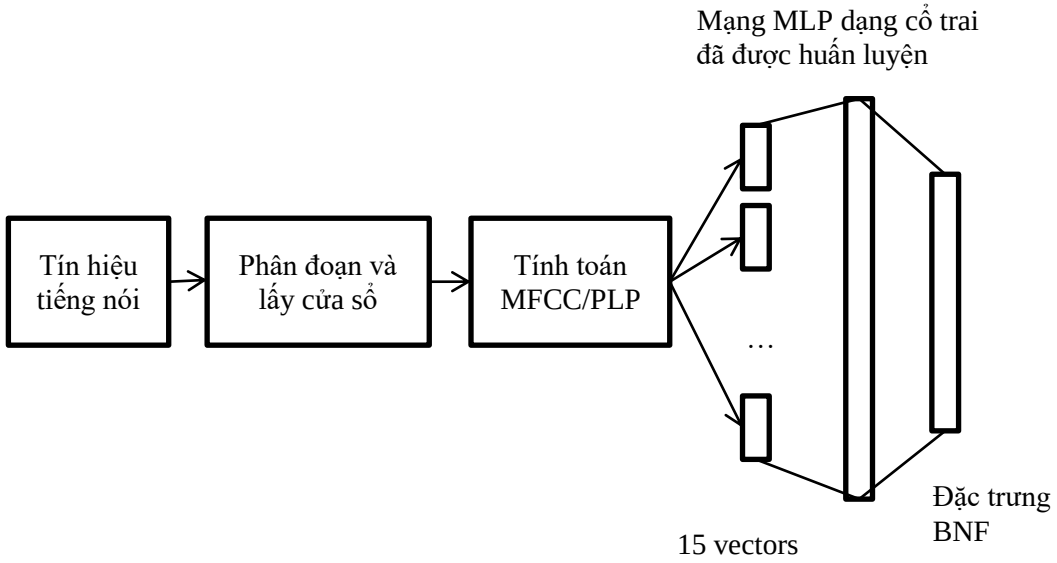
Đã có nhiều nghiên cứu nhằm tìm ra phương pháp trích chọn các đặc trưng của tiếng nói thông qua mạng nơron sao cho có thể sử dụng nó trực tiếp như một đầu vào cho các mô hình HMM, phương pháp Bottleneck là một trong các phương pháp được sử dụng rộng rãi và hiệu quả nhất hiện nay [H. a. Hermansky 2000] [Plahl 2011] [Christian 2011]. Ý tưởng chính của phương pháp là sử dụng một mạng nơron đa lớp MLP đã được huấn luyện để tính toán lại các đặc trưng đầu ra từ các vector đặc trưng đầu vào. Các đặc trưng thu được thực chất là các giá trị kích hoạt (activation) tại các nút mạng ở lớp output, tuy nhiên điều đặc biệt là lớp output được chọn để lấy các giá trị đặc trưng là một trong các lớp ẩn và có kích thước nhỏ, hàm kích hoạt được sử dụng khi tách đặc trưng thường là hàm tuyến tính (linear function) thay vì hàm phi tuyến (non-linear function) [F. a. Grézl 2007] [Vesely 2011] như lúc huấn luyện mạng, khi đó lớp ẩn được chọn để tính toán các đặc trưng gọi là lớp Bottleneck (BN) và đặc trưng thu được qua lớp BN gọi là đặc trưng Bottleneck (BNF).

Như vậy đặc trưng BNF là một dạng đặc trưng của tiếng nói được trích chọn thông qua một mạng MLP có cấu trúc dạng cổ chai. Để tăng tính hiệu quả của đặc trưng này chúng ta cần tìm ra một cấu trúc mạng MLP tốt nhất để tối ưu khả năng phân lớp của mạng khi áp dụng cho một ngôn ngữ hay trên một tập dữ liệu cụ thể. Bốn tham số cơ bản của mạng MLP cần xác định trong trường hợp này là số lớp ẩn, kích thước của các lớp ẩn, kích thước và vị trí của lớp BN.

Trong các nghiên cứu [F. a. Grézl 2007] và [F. a. Grézl 2008] các tác giả đã làm các thử nghiệm khác nhau nhằm tìm ra cấu trúc MLP và vị trí cho lớp BN tốt nhất áp dụng cho tiếng Anh. Cụ thể trong nghiên cứu [F. a. Grézl 2008], tác giả thử nghiệm với hai loại cấu trúc mạng MLP bốn lớp và năm lớp (một lớp input, một lớp output và hai hoặc ba lớp ẩn). Kết quả các thử nghiệm cho thấy cấu trúc mạng MLP năm lớp cho kết quả tốt hơn cấu trúc mạng MLP bốn lớp khoảng 1% tuyệt đối theo độ đo lỗi từ WER (Word Error Rate). Cũng theo các kết quả đó thì vị trí của lớp BN là lớp ẩn thứ hai sẽ cho kết quả tối ưu. Với vị trí này lớp BN sẽ tận dụng được khả năng phân lớp qua lớp ẩn thứ nhất. Kích thước của lớp BN nằm trong khoảng từ 25-65. Kết quả của nghiên cứu [F. a. Grézl 2007] và [Vesely 2011] đạt kết quả tốt nhất với kích thước của BN là 39. Việc chọn kích thước của lớp BN lớn hơn có thể làm giảm WER trên một bộ dữ liệu cụ thể, tuy nhiên độ giảm là rất nhỏ trong khi đó nó sẽ làm tăng đáng kể thời

gian huấn luyện mạng MLP và đồng thời cũng làm tăng kích thước của vector đặc trưng BNF dẫn đến làm tăng kích thước của mô hình HMM. Dựa trên các kết quả nghiên cứu này và các kết quả tương tự cho các ngôn ngữ khác như tiếng Anh, tiếng Tây Ban Nha [K. a. Kevin 2011] [Stuker 2011], luận án đề xuất cài đặt BNF cho tiếng Việt với cấu trúc mạng MLP năm lớp có dạng L1-L2-L3-L4-L5. Trong đó: L1 là lớp input, kích thước của L1 phụ thuộc vào kích thước của đặc trưng đầu vào. L2 và L4 là lớp ẩn thứ nhất và thứ ba. L3 là lớp BN. L5 là lớp ouput, kích thước của L5 phục thuộc vào số lớp (classes) đầu ra mà mạng MLP cần phân lớp. Kích thước của L2, L3 và L4 cần được xác định thông qua các thử nghiệm để thu được cấu hình tối ưu. Cấu trúc mạng MLP này được mô tả ở Hình 4-5.

4.4.2. Trích chọn đặc trưng Bottleneck (BNF)



Hình 4-6: Sơ đồ khối các bước trích chọn đặc trưng BNF

Quá trình trích chọn đặc trưng BNF được mô tả ở Hình 4-6. Toàn bộ dữ liệu huấn luyện sẽ được sử dụng như là đầu vào để trích chọn đặc trưng BNF. Tín hiệu tiếng nói sau khi được phân đoạn sử dụng cửa sổ có độ dài 25ms với tốc độ 10ms sẽ được đưa qua module phân tích để thu được đặc trưng PLP hoặc MFCC, sau đó mỗi 15 khung liên tiếp sẽ được tổ hợp để tạo ra một vector đầu vào cho MLP, ta gọi đầu vào này là *X*. Như đã trình bày ở mục 4.4.1 tại bước trích chọn đặc trưng này chúng ta chỉ sử dụng ba lớp đầu tiên của mạng MLP (L1, L2, L3) đã được huấn luyện để tính toán BNF. *X* sẽ được lan truyền thẳng từ lớp đầu vào L1 đến lớp L3, tại đây hàm kích hoạt tuyến tính được sử dụng để tính BNF như công thức sau:

$$BNF_k = \sum_{j=1}^{N} A_j * W_{k,j} + B_k \qquad (4.5)$$

Trong đó:

- BNF_k là giá trị kích hoạt của nút mạng thứ *k* trong lớp L3 (lớp BN), với *k=1,..,K*.

- N là kích thước của lớp ẩn thứ nhất L2.
- A_j là giá trị kích hoạt tại nút thứ j ở lớp ẩn thứ nhất L2 được tính theo công thức (4.1).
- $W_{k,j}$ trọng số của liên kết giữa nút j ở lớp L3 với nút thứ k trong lớp L2.
- B_k là hệ số Bias của nút thứ k tại lớp L3.

Sau khi thu được các giá trị BNF_k ở lớp BN của mạng MLP, các giá trị này được chuẩn hoá thông qua hàm trung bình và độ lệch chuẩn như công thức (4.6). Giá trị cuối cùng sau bước này sẽ được sử dụng như đặc trưng đầu vào cho mô hình âm học.

$$BNF'_t = \frac{BNF_t - mean(BNF_0)}{dev(BNF_0)} \tag{4.6}$$

$$mean(BNF_0) = \frac{1}{T}\sum_{t=1}^{T} BNF_t$$

$$dev(BNF_0) = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(BNF_t - mean(BNF_0))^2}$$

Trong đó:

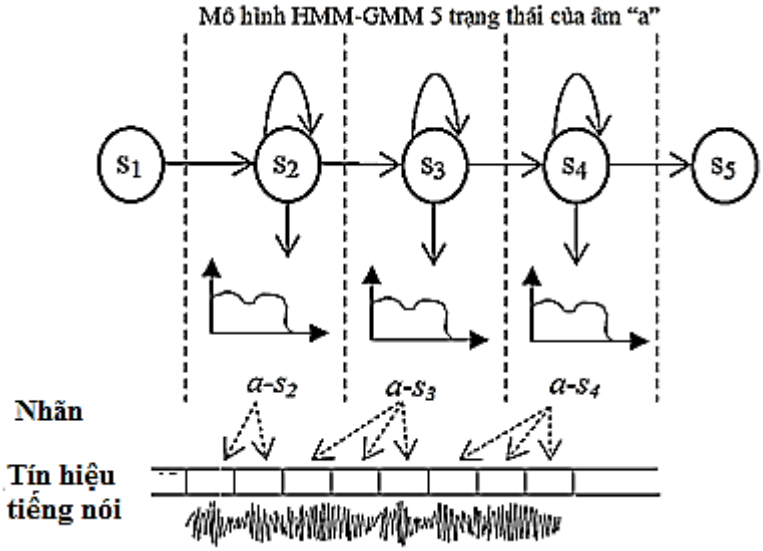
- BNF_t là giá trị BNF của vector đầu vào X_t trong chuỗi vector đầu vào X có độ dài T của chuỗi phát âm hiện thời.
- BNF_0 là chuỗi giá trị BNF của X.

4.5. Cài đặt thử nghiệm

4.5.1. Gán nhãn dữ liệu huấn luyện mạng

Để có dữ liệu huấn luyện cho mạng MLP trước khi được sử dụng để tính toán đặc trưng Bottleneck, luận án sử dụng hệ thống nhận dạng cơ sở Baseline để gán nhãn tự động cho toàn bộ dữ liệu. Đây là hệ thống đã được xây dựng trên từ điển có dấu Tonal-Dict gồm 154 âm vị. Việc gán nhãn được thực hiện bằng cách sử dụng hàm Hvite của HTK. Quá trình gán nhãn thực chất là quá trình nhận dạng cưỡng bức, tức là đã biết trước nội dung nhận dạng và chỉ thực hiện phân đoạn mẫu đầu vào theo thời gian tương ứng với nội dung đã có. Do kích thước tập âm vị của Tonal-Dict trong HMM-2 là 154, dữ liệu được gán nhãn ở mức trạng thái đơn âm (monophone stage), mà mỗi mô hình HMM có 3 trạng thái (không tính hai trạng thái đầu và cuối) nên tổng

số trạng thái đơn của các mô hình HMM trong HMM-2 là 154*3=462. Đối với âm câm (Silence) luận án chỉ sử dụng một trạng thái. Như vậy tổng số nhãn khác biệt trong tập dữ liệu gán nhãn lại là 463 (154*3+1). Đây cũng chính là kích thước lớp đầu ra L5 của mạng MLP sẽ được sử dụng trong luận án này. Hình 4-7 mô tả phương pháp gán nhãn mà luận án sử dụng cho âm "a". Chất lượng hay độ chính xác của nhãn phụ thuộc hoàn toàn vào chất lượng của mô hình âm học đã huấn luyện HMM-2. Do bộ dữ liệu để gán nhãn cũng chính là bộ dữ liệu sử dụng để huấn luyện HMM-2, bộ dữ liệu này thường chỉ có thông tin về phiên âm của dữ liệu mà không thông tin về nhãn thời gian, vì thế độ chính xác trong quá trình huấn luyện mô hình âm học (traning accuracy) có thể được dụng để đánh giá chất lượng của quá trình gánnhãn. Nếu tham số này càng lớn thì dộ chính xác của nhãn càng lớn.

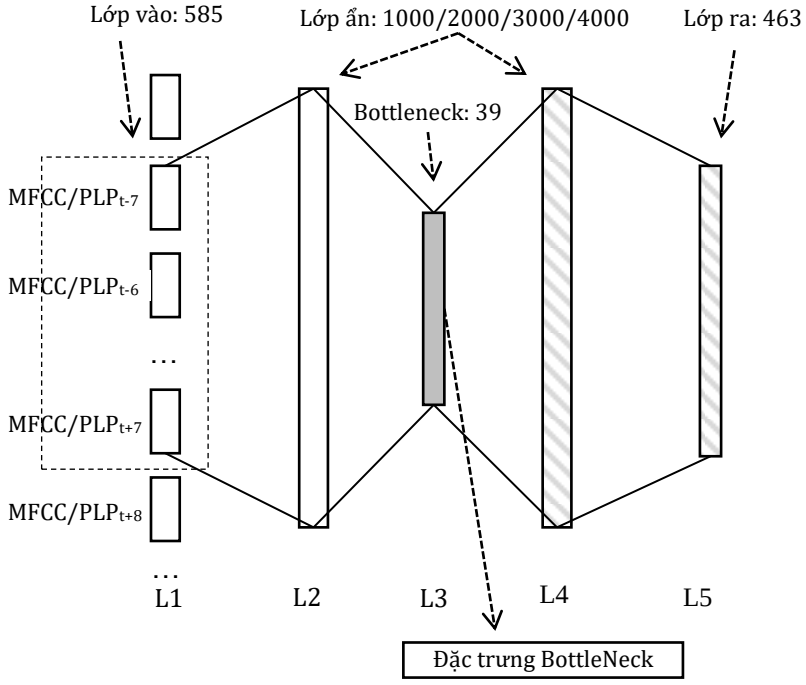


Hình 4-7: Gán nhãn mức monophone stage cho âm "a"

4.5.2. Lựa chọn cấu hình mạng MLP

Mô hình MLP được sử dụng để tính toán được trưng BNF trong luận án là một mạng MLP có 5 lớp L1, L2, L3, L4, L5 như đã mô tả ở Hình 4-5. Vấn đề gặp phải khi thiết lập cấu hình MLP đó là lựa chọn kích thước của các lớp từ L1 đến L5 bằng bao nhiêu để hệ thống nhận dạng đạt kết quả tốt nhất. Trong đó kích thước L1 là kích thước đầu vào. Trong các thử nghiệm ở đây luận án sử dụng đặc trưng đầu vào là MFCC và PLP với cửa sổ khung frame windows là 15. Tức là tổ hợp liên tiếp 15 vector MFCC/PLP (7 vector bên trái và 7 vector bên phải của một vector hiện thời) làm đầu vào cho mạng MLP. Như vậy kích thước của lớp đầu vào L1 sẽ là 39*15=585 (trong đó số 39 là kích thước của một vector MFCC/PLP). Qua các nghiên cứu về BNF đã công bố trên các ngôn ngữ khác nhau cho thấy kích thước lớp BN thường

được chọn trong khoảng từ 25 đến 45, kích thước của L2 và L4 trong khoảng từ 1000 đến 4000. Bước đầu để đánh giá hiệu quả của Bottleneck với tiếng Việt luận án chọn kích thước lớp BN là 39 để thử nghiệm trước. Kích thước của L2 và L4 sẽ được thay đổi với các giá trị {1000,2000,3000,4000} khi thử nghiệm để tìm ra cấu trúc mạng tối ưu. Hình 4-8 mô tả cấu hình mạng MLP mà luận án lựa chọn để thử nghiệm với tiếng Việt.



Hình 4-8: Cấu hình mạng MLP thử nghiệm cho tiếng Việt

4.5.3. Huấn luyện mạng MLP

Tất cả các mạng MLP thử nghiệm đều được huấn luyện bằng công cụ Quicknet [Farber 1997]. Luận án sử dụng hàm Sigmoid như công thức (4.7) làm hàm kích hoạt ở lớp ẩn và hàm Softmax như công thức (4.8) làm hàm kích hoạt ở lớp ra. Tất cả các mạng đều được huấn luyện với hệ số học (learning rate) khởi đầu là 0.05. Các vòng lặp huấn luyện được thực hiện liên tiếp và dừng lại khi tham số đánh giá chéo (Cross Validation Accuracy - CV) trên tập VOV-test giữa hai vòng liên tiếp lệch nhau là 0.001. CV là tỷ lệ nhận dạng chính xác của mạng trên tập dữ liệu thử nghiệm.

$$Sigmoid_{i,j} = (1 + e^{-A_{i,j}})^{-1} \qquad (4.7)$$

$$Softmax_{i,j} = \frac{e^{A_{i,j}}}{\sum_{k=1}^{S_{L5}} e^{A_{i,k}}} \qquad (4.8)$$

Trong đó: $A_{i,j}$ là giá trị kích hoạt tuyến tính tại nút thứ j trong lớp i như công thức (4.1). $S_{L5} = 463$ là kích thước lớp Output.

Bảng 4-1: Kết quả huấn luyện mạng MLP với kích thước L2 và L4 thay đổi

TT	Đặc trưng đầu vào	Ký hiệu	Cấu hình mạng	Cross Validation Accuracy(%) (CV)
1	PLP	PLP-1-1	585x1000x39x1000x463	56.79
2		PLP-1-2	585x1000x39x2000x463	56.85
3		PLP-1-3	585x1000x39x3000x463	56.87
4		PLP-1-4	585x1000x39x4000x463	55.90
5		PLP-2-1	585x2000x39x1000x463	59.17
6		PLP-2-2	585x2000x39x2000x463	59.22
7		PLP-2-3	585x2000x39x3000x463	58.81
8		PLP-2-4	585x2000x39x4000x463	57.21
9		PLP-3-1	585x3000x39x1000x463	60.54
10		PLP-3-2	585x3000x39x2000x463	60.67
11		PLP-3-3	585x3000x39x3000x463	61.03
12		PLP-3-4	585x3000x39x4000x463	60.73
13		PLP-4-1	585x4000x39x1000x463	62.43
14		**PLP-4-2**	**585x4000x39x2000x463**	**62.72**
15		PLP-4-3	585x4000x39x3000x463	62.72
16		PLP-4-4	585x4000x39x4000x463	61.75
17	MFCC	MFCC-1-1	585x1000x39x1000x463	50.01
18		MFCC-1-2	585x1000x39x2000x463	50.92
19		MFCC-1-3	585x1000x39x3000x463	49.13
20		MFCC-1-4	585x1000x39x4000x463	48.22
21		MFCC-2-1	585x2000x39x1000x463	52.77
22		MFCC-2-2	585x2000x39x2000x463	52.89
23		MFCC-2-3	585x2000x39x3000x463	51.11
24		MFCC-2-4	585x2000x39x4000x463	50.16
25		MFCC-3-1	585x3000x39x1000x463	54.09
26		MFCC-3-2	585x3000x39x2000x463	54.57
27		MFCC-3-3	585x3000x39x3000x463	55.78
28		MFCC-3-4	585x3000x39x4000x463	54.32
29		MFCC-4-1	585x4000x39x1000x463	55.21
30		MFCC-4-2	585x4000x39x2000x463	56.53
31		**MFCC-4-3**	**585x4000x39x3000x463**	**56.57**
32		MFCC-4-4	585x4000x39x4000x463	56.13

Bảng 4-1 trình bày kết quả huấn luyện mạng các mạng MLP với kích thước L2 và L4 thay đổi trên tập dữ liệu VOV. Chất lượng của mạng được đánh giá thông qua tham số CV trên tập VOV-test. Từ kết quả thử nghiệm cho thấy bộ kích thước L2 và L4 tương ứng là 4000 và 2000 cho kết quả CV tốt nhất trên đặc trưng PLP và 4000, 3000 cho CV tốt nhất trên đặc trưng MFCC. Các bộ kích thước này sẽ được sử dụng để trích chọn đặc trưng BNF trong thử nghiệm áp dụng BNF cho mô hình HMM.

4.5.4. Áp dụng đặc trưng BNF với mô hình HMM

1) Dữ liệu, từ điển, mô hình ngôn ngữ

- Dữ liệu huấn luyện: VOV.
- Dữ liệu thử nghiệm: VOV-Test.
- Từ điển: Tonal-Dict của hệ thống HMM-2.
- Mô hình ngôn ngữ: VOV-BiGram-LM của hệ thống Baseline.

2) Trích chọn đặc trưng BNF

Mô hình mạng MLP PLP-4-2 được sử dụng để tính toán đặc trưng BNF_{PLP} với đầu vào PLP và MFCC-4-3 được sử dụng để tính toán đặc trưng BNF_{MFCC} với đầu vào MFCC. Quy trình tính toán đặc trưng BNF được thực hiện như Hình 4-6. Kích thước của BNF_{PLP} và BNF_{MFCC} đều là 39 do kích thước của lớp BN đã được chọn là 39. BNF_{PLP} và BNF_{MFCC} sau đó được chuẩn hoá theo công thức (4.6) và được sử dụng trực tiếp như là đầu vào để huấn luyện các mô hình HMM.

3) Huấn luyện mô hình âm học

Hai hệ thống sử dụng mô hình HMM được khai báo lại với kích thước đặc trưng đầu vào là 39. Sau đó các mô hình này được huấn luyện theo các bước và tham số tương tự như hệ thống cơ sở. Hệ thống cuối cùng được huấn luyện ở mức tri-phone với 2179 âm buộc.

4) Kết quả thử nghiệm

Kết quả nhận dạng của hai hệ thống sử dụng đặc trưng BNF trên tập VOV-Test được trình bày ở Bảng 4-2. So sánh với hệ thống HMM-2 là hệ thống đã được sử dụng để gán nhãn dữ liệu huấn luyện mạng thì hệ thống sử dụng đặc trưng BNF_{MFCC} tốt hơn 1.25% tuyệt đối. Qua kết quả này cho thấy đặc trưng BNF có hiệu quả với tiếng Việt, và với kích thước lớp BN không đổi thì mạng có CV càng cao sẽ cho đặc trưng BNF càng tốt. Vấn đề tiếp theo là cần tìm kích thước đặc trưng BNF hay kích thước của lớp BN để thu được chất lượng nhận dạng tốt nhất.

Bảng 4-2: Kết quả thử nghiệm đặc trưng BNF

TT	Hệ thống	Đặc trưng	Từ điển	ACC (%)
1	HMM-2	MFCC	Tonal-Dict	78.31
2	BNF-1	BNF_{PLP}		79.33
3	**BNF-2**	**BNF_{MFCC}**		**79.56(+1.25)**

4.6. Tối ưu đặc trưng Bottleneck

4.6.1. Huấn luyện mạng MLP với kích thước BN thay đổi

Bảng 4-3: Kết quả huấn luyện mạng MLP với kích thước lớp BottleBeck thay đổi

TT	Đặc trưng đầu vào	Ký hiệu	Cấu hình mạng	Accuracy(%)
1	PLP	PLP-4-2-45	585x4000x45x2000x463	79.15
2		PLP-4-2-39	585x4000x39x2000x463	79.33
3		PLP-4-2-33	585x4000x33x2000x463	80.15
4		PLP-4-2-29	585x4000x29x2000x463	80.60
5		PLP-4-2-25	585x4000x25x2000x463	81.63
6		PLP-4-2-21	585x4000x21x2000x463	82.23
7		PLP-4-2-17	585x4000x17x2000x463	83.20
8		**PLP-4-2-13**	**585x4000x13x2000x463**	**83.30**
9		PLP-4-2-11	585x4000x11x2000x463	83.19
10		PLP-4-2-09	585x4000x09x2000x463	81.99
11	MFCC	MFCC-4-3-45	585x4000x45x3000x463	79.03
12		MFCC-4-3-39	585x4000x39x3000x463	79.56
13		MFCC-4-3-33	585x4000x33x3000x463	80.91
14		MFCC-4-3-29	585x4000x29x3000x463	81.37
15		MFCC-4-3-25	585x4000x25x3000x463	82.46
16		MFCC-4-3-21	585x4000x21x3000x463	82.67
17		MFCC-4-3-17	585x4000x17x3000x463	83.76
18		**MFCC-4-3-13**	**585x4000x13x3000x463**	**84.18**
19		MFCC-4-3-11	585x4000x11x3000x463	83.51
20		MFCC-4-3-09	585x4000x09x3000x463	81.10

Để tìm ra kích thước lớp BN tối ưu luận án tiếp tục huấn luyện thêm các mạng MLP có kích thước của lớp BN thay đổi. Tuy nhiên ở bước này kích thước của các lớp L2 và L4 là không đổi và được lấy từ bộ kích thước cho kết quả CV tốt nhất ở mục 4.5.3. Như vậy cấu hình của các mạng MLP sử dụng đặc trưng đầu vào MFCC là 585x4000x*Y*x3000x463, của các mạng MLP sử dụng đặc trưng đầu vào PLP là 585x4000x*Y*x2000x463. Trong đó *Y* là kích thước lớp BN sẽ được thay đổi với các giá

trị sau Y={9,11,13,17,21,25,29,33,39,45}. Quá trình huấn luyện mạng và các tham số huấn luyện được chọn tương tự như đã thực hiện ở bước huấn luyện với kích thước L2, L4 thay đổi. Kết quả huấn luyện mạng được trình bày ở Bảng 4-3.

4.6.2. Cài đặt thử nghiệm với đặc trưng BN có kích thước thay đổi

1) Dữ liệu, từ điển, mô hình ngôn ngữ

- Dữ liệu huấn luyện: VOV.
- Dữ liệu thử nghiệm: VOV-Test.
- Từ điển: Tonal-Dict.
- Mô hình ngôn ngữ: VOV-BiGram-LM của hệ thống Baseline.

2) Trích chọn đặc trưng BNF

Ở bước này có 20 loại đặc trưng BNF được trích chọn tương ứng với 20 cấu hình mạng với kích thước lớp Bottleneck thay đổi như đã trình bày ở Bảng 4-3. Tất cả các quy trình tính toán đặc trưng BNF đều được thực hiện tương tự như đã trình bày ở mục 4.4.2. Sự khác biệt duy nhất giữa các đặc trưng đó là kích thước, kích thước của đặc trưng BNF đầu ra trùng với kích thước lớp BN của mạng MLP được sử dụng để tính toán.

3) Huấn luyện mô hình âm học

20 hệ thống được tạo ra tương ứng với 20 loại đặc trưng BNF đã được trích chọn ở mục trên. Các mô hình HMM của các hệ thống được huấn luyện theo các bước và tham số tương tự như hệ thống cơ sở. Hệ thống cuối cùng được huấn luyện ở mức tri-phone với 2179 âm buộc.

4) Kết quả thử nghiệm

Kết quả nhận dạng của các hệ thống sử dụng đặc trưng BNF trên tập VOV-Test được trình bày ở Bảng 4-3 cột Accuracy. Từ kết quả này cho thấy tất cả các đặc trưng BN có kích thước khác nhau đều cho chất lượng nhận dạng tốt hơn hệ thống HMM-2 và đặc trưng được trích chọn từ cấu hình mạng có kích thước lớp BN là 13 cho chất lượng nhận dạng tốt nhất. Cụ thể là đặc trưng từ mạng MFCC-4-3-13 cho chất lượng tốt hơn 6.48% tuyệt đối so với hệ thống cơ sở (dòng 18 trong Bảng 4-3).

4.7. Kết luận chương

Từ các kết quả thử nghiệm đã được thực hiện trong chương này luận án đi đến một số kết luận như sau:

1) Đặc trưng Bottleneck có hiệu quả với nhận dạng tiếng Việt. Kết quả thử nghiệm tốt nhất cho thấy chất lượng nhận dạng tăng lên 6.48% tuyệt đối (29% tương đối) so với hệ thống cơ sở sau khi áp dụng đặc trưng này. So sánh với các nghiên cứu đã áp dụng cho tiếng Anh, Đức [K. a. Kevin 2013]. Bottleneck giúp tăng 10% tương đối cho tiếng Anh (350 giờ dữ liệu huấn luyện), 8% tương đối cho tiếng Đức (200h giờ dữ liệu huấn luyện) trên bộ dữ liệu thu từ giảng đường (Tedtalk). Kết quả này cho thấy Bottleneck đã được nghiên cứu áp dụng thành công cho tiếng Việt.
2) Để tối ưu chất lượng nhận dạng khi áp dụng đặc trưng Bottleneck thì cần phải có các bước thử nghiệm để lựa chọn ra cấu hình mạng tối ưu. Cấu hình mạng tối ưu phụ thuộc vào từng ngôn ngữ và tập dữ liệu cụ thể.

4.8. Các bài báo đã công bố liên quan đến nội dung của chương

1. **Van Huy Nguyen**, Chi Mai Luong, Tat Thang Vu, *Tonal phoneme based model for Vietnamese LVCSR*, IEEE Conference of the Oriental chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (OCOCOSDA), Shanghai-China, Oct-2015.
2. **Nguyễn Văn Huy**, *Nâng cao chất lượng đặc trưng Bottleneck cho nhận dạng tiếng Việt*, Tạp chí Khoa học và Công nghệ Đại học Thái Nguyên, ISSN 1859-2171, Tập 137, Số 07, 2015.
3. **Nguyen Van Huy**, Luong Chi Mai, Vu Tat Thang, *Áp dụng Bottle neck Feature cho nhận dạng tiếng Việt*, Journal of Computer Science and Cybernetics, Vietnam, ISSN 1813-9663, Vol 29, No 4, Oct-2013.
4. Kevin Kilgour, Christian Mohr, Michael Heck, Quoc Bao Nguyen, **Van Huy Nguyen**, Evgeniy Shin, Igor Tseyzer, Jonas Gehring, Markus Muller, Matthias Sperber, Sebastian Stuker and Alex Waibel , *The 2013 KIT IWSLT Speech-to-Text Systems for German and English*, International Workshop on Spoken Language Translation (IWSLT), Germany, Dec-2013.

Chương 5: Cải tiến đặc trưng thanh điệu sử dụng mạng nơron và mô hình tích hợp MSD-HMM với Bottleneck

5.1. Tóm tắt chương

Kết quả ở Chương 4 cho thấy Bottleneck có hiệu quả rất tốt trong việc tăng cường đặc trưng ngữ âm cho tiếng Việt. Mục tiêu của chương này của luận án đề xuất một phương pháp mới để tăng cường đặc trưng thanh điệu tượng tự phương pháp Bottleneck. Đặc trưng cải tiến mới này gọi là Tonal-Bottleneck. Tonal-Bottleneck khác Bottleneck ở chỗ nó là đặc trưng thanh điệu và được chỉnh sửa bằng cách bổ sung thêm các vùng đứt gãy tương thích với mô hình MSD-HMM. Từ kết quả thành công cho việc tăng cường đặc trưng ngữ âm của Bottleneck, tăng cường đặc trưng thanh điệu của đặc trưng cải tiến Tonal Bottlenek và mô hình thanh điệu sử dụng MSD-HMM, luận án đi đến đề xuất mô hình tích hợp ba thành phần này vào một mô hình duy nhất.

Nội dung chính của chương bao gồm: Trình bày phương pháp cải tiến của Bottleneck. Mạng nơron MLP sẽ được sử dụng để tính toán đặc trưng thanh điệu (Tonal-Bottleneck - TBNF) tương thích với mô hình MSD-HMM. Sau đó đặc trưng mới này sẽ được áp dụng cho nhận dạng tiếng Việt. Trình bày phương pháp tích hợp mô hình MSD-HMM với BNF và TBNF vào một hệ thống.

5.2. Trích chọn đặc trưng thanh điệu sử dụng mạng nơron

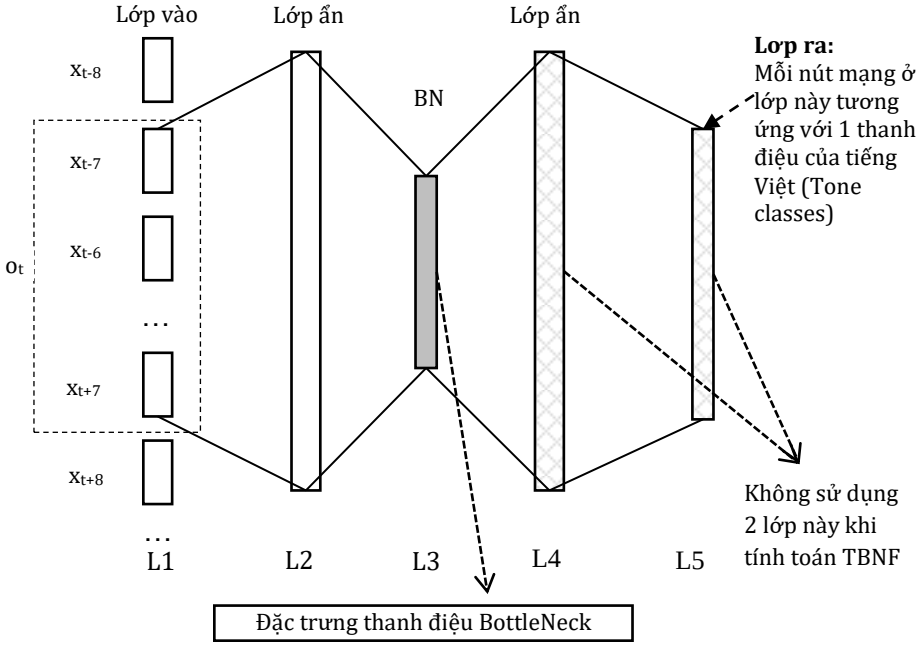
5.2.1. Đặc trưng thanh điệu Tonal Bottleneck (TBNF)

Qua kết quả thử nghiệm ở Chương 4 về việc áp dụng đặc trưng BNF cho nhận dạng tiếng Việt đã cho thấy BNF có hiệu quả và làm tăng đáng kể chất lượng nhận dạng của hệ thống. BNF ở trong các thử nghiệm trước thực chất được tính toán dựa trên hai loại đặc trưng đầu vào là MFCC và PLP, như vậy có thể coi mạng MLP trong trường hợp đó như một mô hình biến đổi đặc trưng đầu vào để thu được đặc trưng mới tốt hơn. Từ kết quả này luận án đi đến đề xuất sử dụng mạng MLP để trích chọn đặc trưng thanh điệu cho nhận dạng tiếng Việt. Đặc trưng mới này được gọi là đặc trưng thanh điệu Bottleneck (Tonal Bottleneck Feature - TBNF). Hai lý do chính để áp dụng đặc trưng TBNF cho nhận dạng tiếng Việt đó là:

1) Các loại đặc trưng thanh điệu AMDF/NCC giúp làm tăng chất lượng nhận dạng cho tiếng Việt như các thử nghiệm đã thực hiện ở Chương 3. Vì vậy cần thiết có một phương pháp làm tăng chất lượng cho hai loại đặc trưng này để thu được kết quả nhận dạng tốt hơn nữa.

2) Các phương pháp tính toán đặc trưng thanh điệu AMDF và NCC thực hiện các tính toán để xác định giá trị Pitch trong một cửa sổ khung có kích thước thông thường từ 20ms đến 45ms. Trong khi đó Pitch lại tồn tại trong suốt vùng hữu thanh của một âm tiết. Kích thước của một âm tiết thường lớn hơn 32ms vì thế các đặc trưng AMDF và NCC sẽ không mang đầy đủ thông tin về sự biến đổi của Pitch trong suốt vùng hữu thanh của âm tiết hiện tại. Để khắc phục nhược điểm này ta có thể tổ hợp thêm các vector đặc trưng Pitch lân cận với vector hiện thời để làm tăng độ dài về mặt thời gian. Sau đó sử dụng vector tổ hợp này như là một vector đặc trưng đầu vào. Tuy nhiên cách tổ hợp này làm gia tăng kích thước của đặc trưng Pitch đầu vào vì vậy cần có một phương pháp biến đổi nó sang một dạng đặc trưng mới tốt hơn và có kích thước nhỏ hơn. Trong phạm vi nghiên cứu này luận án đề xuất sử dụng mạng MLP để biến đổi vector tổ hợp này tương tự như quy trình tính toán đặc trưng BNF ở Chương 4.

5.2.2. Trích chọn đặc trưng thanh điệu TBNF



Hình 5-1: Mô hình mạng MLP để trích chọn đặc trưng TBNF

Để tính toán đặc trưng TBNF luận án sử dụng mạng MLP có 5 lớp tương tự như đã sử dụng để tính toán đặc trưng BNF ở Chương 4. Tuy nhiên kích thước của các lớp sẽ được thử nghiệm để xác định lại sao cho có thể thu được TBNF tối ưu. Sự khác biệt chính giữa hai loại mạng MLP để tính toán BNF và TBNF là ở lớp đầu ra output trong quá trình huấn luyện mạng. Khi tính toán đặc trưng BNF thì lớp ra của mạng là số trạng thái âm vị đơn (monophone stage), nhưng khi tính toán TBNF lớp ra sẽ là số

thanh điệu hoặc số trạng thái HMM của các thanh điệu. Mô hình mạng MLP để tính toán TBNF được mô tả ở Hình 5-1.

Trong mô hình mạng MLP ở Hình 5-1 kích thước của các lớp ẩn L2, L4 và lớp Bottleneck L3 sẽ được thử nghiệm để chọn ra bộ kích thước tối ưu. Kích thước của lớp đầu ra L5 sẽ được chọn theo số lớp thanh điệu. Kích thước của lớp đầu vào L1 được chọn theo kích thước của đặc trưng đầu vào. Đặc trưng đầu vào được chọn ở bước này là NCC, đây là đặc trưng thanh điệu cho kết quả nhận dạng tốt nhất ở các bước thử nghiệm trước đó với mô hình HMM khi kết hợp với MFCC hoặc PLP. Để tăng thông tin về ngữ cảnh thời gian luận án tổ hợp 15 vector đầu vào liên tiếp $X=\{x_{t-7},\dots,x_t,\dots,x_{t+7}\}$ để làm đầu vào cho mạng MLP. Như vậy kích thước của L1 (hay số chiều của vector *X*) sẽ là 15*3=45 (giá trị 3 là kích thước của mỗi vector NCC).

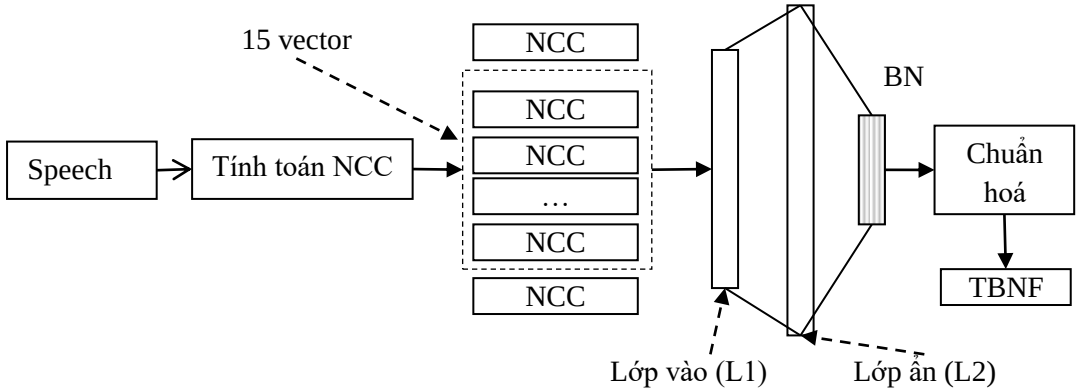
TBNF được tính toán theo công thức (5.1) và sau đó được áp dụng hàm chuẩn hoá như công thức (5.2). Hình 5-2 mô tả các bước thực hiện để tính toán TBNF.

$$TBNF_k = \sum_{j=1}^{N} f_{act_j} * W_{k,j} + b_k^{BN}, k = 1, \dots, K \quad (5.1)$$

Trong đó:

$$f_{act_j} = Sigmoid(\sum_{q=1}^{45} x_{t_q} * W_{j,q} + b_j^{L2}, j = 1, \dots, N)$$

- $TBNF_k$ là giá trị kích hoạt của nút mạng thứ *k* trong lớp BN, với *K* là kích thước của lớp BN.
- *N* là kích thước của lớp ẩn thứ nhất L2.
- f_{act_j} là giá trị kích hoạt tại nút thứ *j* ở lớp ẩn thứ nhất L2.
- Hàm *Sigmoid* được thính toán theo công thức (4.7).
- $W_{k,j}$là trọng số của liên kết giữa nút *k* ở lớp L3 với nút thứ *j* trong lớp L2.
- $W_{j,q}$ là trọng số liên kết giữa nút *j* ở lớp L2 với nút thứ *q* trong lớp đầu vào L1.
- b_k^{BN} là hệ số Bias của nút thứ *k* tại lớp Bottleneck.
- b_j^{L2}là hệ số Bias của nút thứ j tại lớp L2.
- x_{t_q} là phần tử thứ *q* trong vector đặc trưng đầu vào x_t.



Hình 5-2: Sơ đồ khối các bước tính toán TBNF

$$TBNF'_t = \frac{TBNF_t - mean(TBNF_0)}{dev(TBNF_0)} \quad (5.2)$$

$$mean(TBNF_0) = \frac{1}{T}\sum_{k=1}^{T} TBNF_k$$

$$Dev(TBNF_0) = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(TBNF_t - Mean(TBNF_0))^2}$$

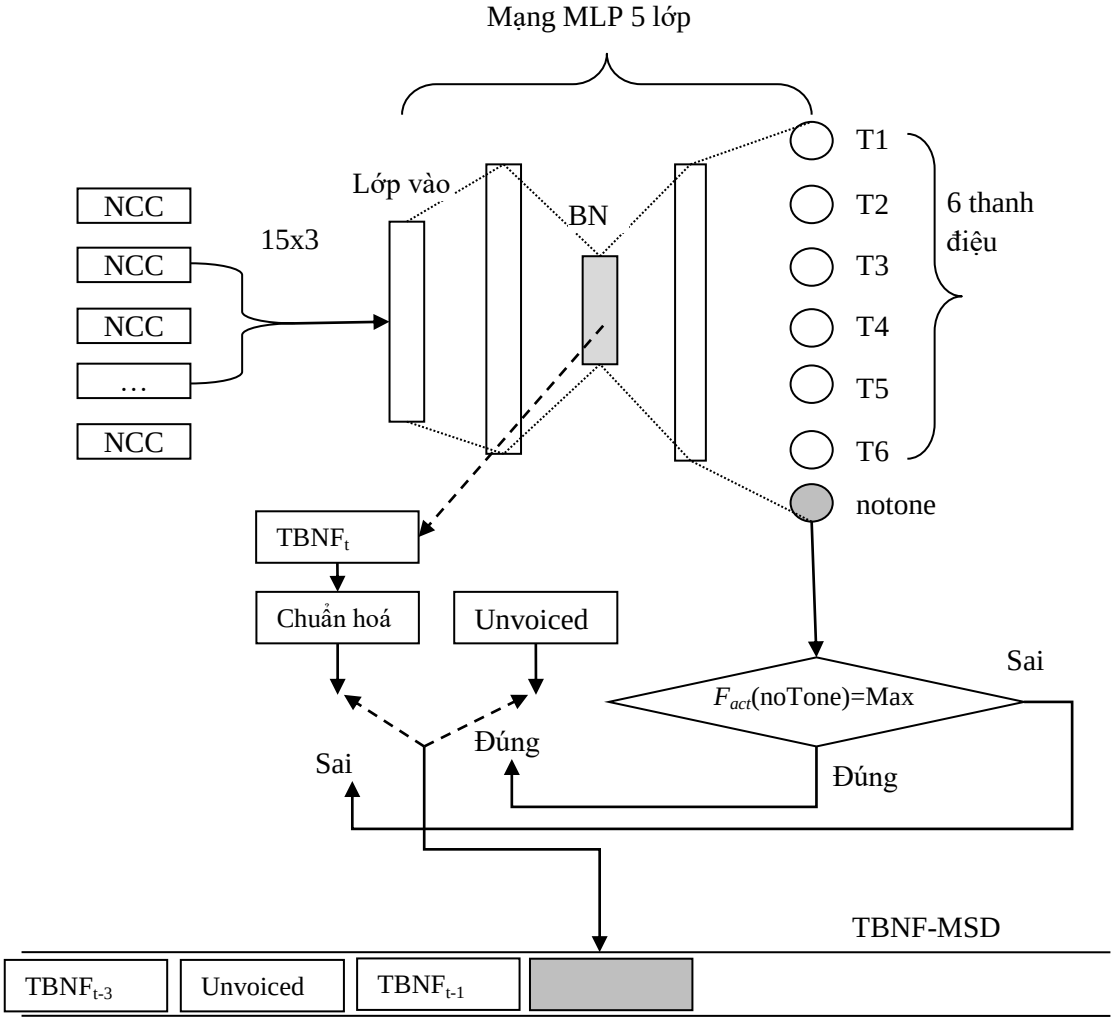
Trong đó:

- $TBNF_t$ là giá trị TBNF của vector đầu vào x_t trong chuỗi vector đặc trưng đầu vào X có độ dài T.
- $TBNF_0$ là chuỗi giá trị BNF của X.

5.2.3. Cải tiến đặc trưng TBNF cho mô hình MSD-HMM

TBNF thực chất là các giá trị kích hoạt tại lớp BN vì thế đây là một loại đặc trưng liên tục, hay nói cách khác là TBNF chỉ chứa các giá trị số thực. Để sử dụng được TBNF cho mô hình MSD-HMM thì TBNF cần được bổ sung thêm các giá trị rời rạc. Luận án đề xuất phương pháp biến đổi đặc trưng TBNF sang một dạng đặc trưng khác tương thích với mô hình MSD-HMM gọi là TBNF_MSD. Các giá trị của TBNF_MSD sẽ thuộc một trong hai không gian $\{\Omega_1, \Omega_1\}$. Trong đó Ω_1 là không gian số thực có số chiều đúng bằng kích thước của lớp BN. Ω_2 là không gian chứa giá trị rời rạc. Ω_2 chỉ có duy nhất một giá trị là ký hiệu "unvoiced". Ý tưởng chính của việc tính toán TBNF_MSD là sửa lại TBNF bằng cách gán tất cả các vector của TBNF thuộc vùng vô thanh bằng giá trị "unvoiced". Các vector thuộc TBNF bị gán giá trị

này là các vector tương ứng với vector đầu vào của mạng x_t nếu x_t được mạng đoán nhận là thuộc lớp vô thanh "notone".



Hình 5-3: Sơ đồ khối các bước biến đổi TBNF sang TBNF-MSD

Xét chuỗi vector đặc trưng đầu vào $X=\{x_1,..,x_t,..,x_T\}$ có độ dài T. Mạng MLP sử dụng để tính toán TBNF_MSD vẫn là mạng MLP đã được sử dụng để tính toán TBNF, nhưng cả 5 lớp của mạng này sẽ được sử dụng chứ không phải chỉ 3 lớp mạng đầu tiên được dùng như phương pháp tính toán TBNF. Trong phương pháp này mạng MLP được sử dụng với hai chức năng là tính TBNF và đoán nhận một vector đầu vào x_t thuộc vùng vô thanh hay không. TBNF_MSD được tính toán như công thức (5.3).

$$TBNF\_MSD_t = \begin{cases} TBNF_t, nếu\ agrmax\big(O(x_t)\big) \neq "Notone" \\ "\text{unvoiced}", nếu\ agrmax\big(O(x_t)\big) = "Notone" \end{cases} \quad (5.3)$$

Trong đó:

- $TBNF_t$ là giá trị *TBNF* của x_t xác định được như công thức (5.1).
- $O(x_t)$ là một vector chứa giá trị của các nút mạng ở lớp đầu ra L5 của mạng MLP.
- "Notone" chỉ nút mạng ở lớp ra L5 tương ứng với lớp các mẫu đầu vào x_t không tồn tại thanh điệu (notone).

Hình 5-3 mô tả phương pháp tính toán TBNF-MSD như đã trình bày ở trên với giả thiết số lớp thanh điệu là 6 và 1 lớp cho trường hợp không tồn tại thanh điệu (notone).

5.3. Gán nhãn dữ liệu

Mạng MLP cần được huấn luyện trước khi được sử dụng để tính toán đặc trưng TBNF. Vấn đề đặt ra là phương pháp gán nhãn cho dữ liệu huấn luyện như thế nào. Khác với trường hợp tính toán BNF, đặc trưng BNF thu được là một dạng của đặc trưng ngữ âm vì thế BNF biểu diễn đặc tính cho các âm vị. Trong trường hợp này TBNF lại biểu diễn đặc tính cho các thanh điệu. Như vậy có thể coi mạng MLP trong trường hợp này được dùng để phân lớp thanh điệu cho đặc trưng đầu vào. Vì thế dữ liệu huấn luyện mạng cần được gán nhãn ở mức thanh điệu. Vấn đề tiếp theo là thực hiện gán nhãn dữ liệu ở mức trạng thái HMM của thanh điệu (tone stage) hay mức thanh điệu (tone). Để trả lời câu hỏi này luận án tiến hành các thử nghiệm cả hai phương án để chọn ra phương án tối ưu.

5.3.1. Gán nhãn mức trạng thái HMM của thanh điệu (Tone Stage Labeling - TSL)

Mô hình HMM được sử dụng trong các thử nghiệm của luận án là một mô hình HMM dạng left-right có 5 trạng thái. Như đã trình bày ở mục 0 thì tiếng Việt có 6 thanh điệu. Nếu gán nhãn dữ liệu ở mức trạng thái của HMM thì tổng số nhãn sẽ là 6*3=18. Trong đó giá trị 6 tương ứng với 6 thanh điệu, 3 là số trạng thái của mỗi mô hình HMM đã bỏ qua 2 trạng thái đầu vào cuối. Ở các vùng vô thanh không tồn tại thanh điệu, các vùng này sẽ được gán là noTone. Vậy tổng số nhãn ở mức trạng thái HMM để gán cho dữ liệu huấn luyện sẽ là 18+1=19.

- ***Thuật toán gán nhãn thanh điệu tự động:***

Đầu vào: File way và file phiên âm (transcript) của dữ liệu cần gán nhãn
Đầu ra: Nhãn theo thời gian mức trạng thái HMM của thanh điệu cho dữ liệu đầu vào
Thuật toán:
Bước 1: Gán nhãn âm vị (Y):

Xét chuỗi vector đầu vào $X=\{x_t\}, t=1,..,T$ thì nhãn mức âm vị của X là:

$$Y = Label(X)=\{y_t\}$$
$$W=Word(X)=\{w_t\}$$

Trong đó: $y_t \in P$ là nhãn của x_t, P là tập âm vị mức trạng thái HMM; $t=1..T$; W nhãn mức từ tương ứng của x_t; *Label(X)* và *Word(X)* là thủ tục gán nhãn cưỡng bức cho X bằng cách sử dụng một hệ thống nhận dạng đã có. Quá trình gán nhãn cưỡng bước là quá trình phân đoạn thời gian theo âm vị hoặc theo từ đã biết trước cho chuỗi âm thanh đầu vào sử dụng một mô hình nhận dạng đã có.

Bước 2: Gán nhãn thanh điệu thô (Z)

$$Z = \{z_t\} = \begin{cases} Tone(y_t): y_t = Necleus/Coda(w_t) \\ \text{"Notone"}: y_t = Initial/Onset(w_t) \end{cases}$$

Trong đó: z_t là nhãn thanh điệu của x_t; *Tone*(y_t) là hàm lấy thanh điệu từ nhãn âm vị y_t bằng cách xóa bỏ các ký hiệu âm vị và giữ lại ký hiệu thanh điệu. *Initial/Onset/Necleus/Coda*(w_t) là thanh phần âm đầu, âm đệm, âm chính và âm cuối của từ w_t.

Bước 3: Chuẩn hóa (bỏ các nhãn thanh điệu thuộc vùng vô thanh của *X*)

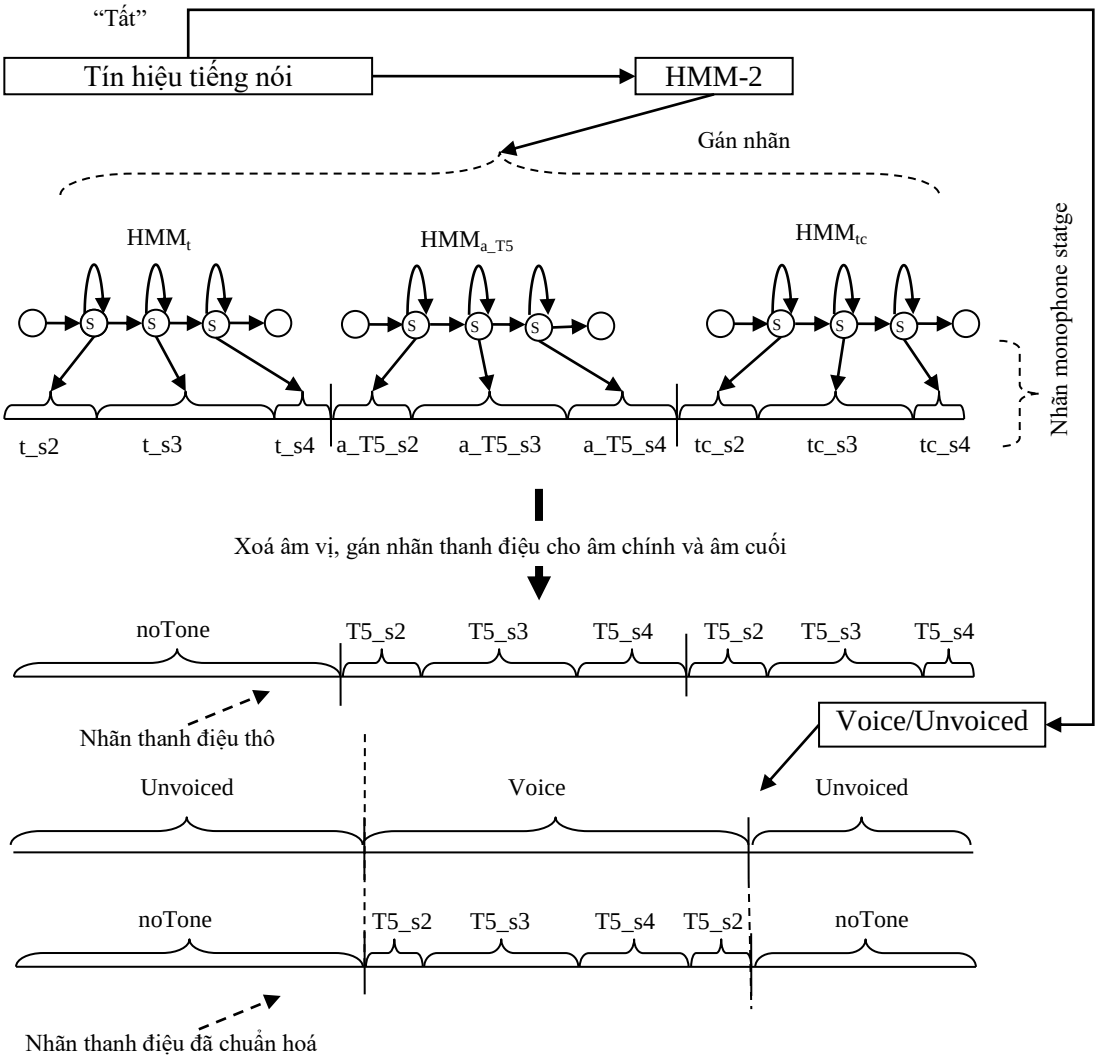
$$Q = \{q_t\} = \begin{cases} z_t: UV(x_t) = 1 \\ \text{"}Notone\text{"}: UV(x_t) = 0 \end{cases}$$

Trong đó: q_t là nhãn thanh điệu đã chuẩn hóa của x_t, *UV*(x_t) là hàm xác định vector x_t thuộc vùng vô thanh theo công thức (3.20).

Kết thúc thuật toán.

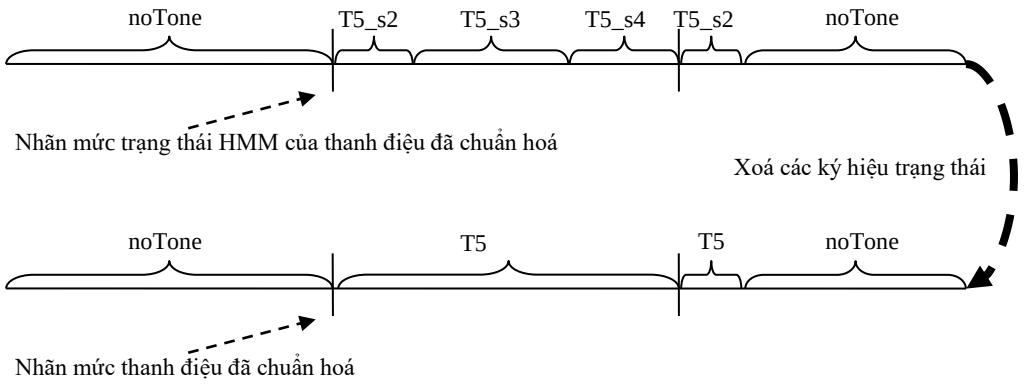
Quy trình gán nhãn được thực hiện như sau. Đầu tiên hệ thống HMM-2 đã được xây dựng ở mục 0 được sử dụng để gán nhãn cưỡng bức cho tập dữ liệu huấn luyện VOV. Sau bước này ta thu được bộ dữ liệu đã được gán nhãn ở mức monophone stage. Thông tin thanh điệu tương ứng với mỗi âm tiết được xác định thông qua thông tin thanh điệu trong nhãn âm vị của nó. Nhãn thanh điệu này được gán cho toàn bộ đoạn dữ liệu tương ứng với phần Âm chính và Âm cuối của âm tiết hiện tại. Phần Âm đầu được gán nhãn là notone. Sau bước này ta thu được dữ liệu đã được gán nhãn thanh điệu ở mức thô. Trong tiếng nói phát âm liên tục thì hai âm tiết cạnh nhau có thể ảnh hưởng đến nhau dẫn đến vùng vô thanh và hữu thanh có thể bị ảnh hưởng lẫn nhau, đồng thời một số âm cuối trong tiếng Việt lại là âm vô thanh. Do đó để chuẩn hoá nhãn thô, luận án áp dụng thêm một kỹ thuật phát hiện vùng hữu thanh và vô thanh lên đoạn dữ liệu đã được gán nhãn có thanh điệu. Nếu khung hiện thời mà được xác định là thuộc vùng vô thanh thì nhãn thanh điệu sẽ được thay thế lại thành notone, trái lại

thì giữ nguyên nhãn được gán ở bước gán thô. Hình 5-4 minh họa phương pháp gán nhãn thanh điệu như đã trình bày ở trên cho phát âm "tất".



Hình 5-4: Quy trình gán nhãn thanh điệu mức trạng thái HMM

5.3.2. Gán nhãn mức thanh điệu (Tone Labeling - TL)



Hình 5-5: Nhãn mức thanh điệu của phát âm "tất"

Nhãn mức thanh điệu (Tone label) được tạo ra bằng cách xoá bỏ các ký hiệu trạng thái (S2, S3, S4) trong bộ dữ liệu đã được gán nhãn mức trạng thái HMM của thanh điệu ở mục trước. Như vậy sẽ có 6 nhãn tương ứng với 6 thanh điệu, một nhãn cho trường hợp noTone. Tổng số nhãn khác biệt trong bộ dữ liệu gán nhãn ở mức thanh điệu là 7. Hình 5-5 minh họa nhãn thanh điệu của phát âm "tất" thu được sau khi xoá bỏ các ký hiệu trạng thái từ nhãn mức trạng thái HMM ở mục trên.

5.4. Lựa chọn cấu hình mạng MLP

5.4.1. Lựa chọn kích thước lớp ra của mạng MLP

Kích thước lớp ra của mạng MLP phụ thuộc vào số lớp mà mạng cần phân lớp hay chính là số nhãn khác biệt có trong cơ sở dữ liệu. Như mục 5.3 đã trình bày luận án sử dụng hai loại nhãn mức trạng thái HMM của thanh điệu (TSL) và mức thanh điệu (TL) để thử nghiệm. Với cơ sở dữ liệu sử dụng TSL thì số lớp đầu ra là 19 do vậy kích thước lớp đầu ra sẽ là 19, tương tự với cơ sở dữ liệu dùng TL thì kích thước lớp ra tương ứng là 7. Để tìm ra loại nhãn hay kích thước lớp ra tốt nhất luận án tiến hành thử nghiệm huấn luyện hai loại mạng trên bộ dữ liệu đã gán nhãn VOV để đánh giá chất lượng mạng. Tham số để đánh giá chất lượng mạng là độ chính xác đánh giá chéo (Cross Validation Accuracy - CV) trên tập dữ liệu thử nghiệm VOV-Test. Cả hai loại mạng đều có cấu trúc 5 lớp dạng L1-L2-L3-L4-L5. Trong đó kích thước của L5 là 19 hoặc 7 tương ứng với hai loại nhãn TSL và TL. Kích thước lớp L3 (BN) được chọn ban đầu là 9. Kích thước lớp đầu vào Input L1 là 45 tương ứng với kích thước của đặc trưng đầu vào như đã trình bày ở mục 5.2.2. Kích thước của hai lớp ẩn L2 và L4 sẽ được thay đổi trong các giá trị {100,200,300,400,500}. Tất cả các mạng MLP thử nghiệm đều được huấn luyện bằng công cụ Quicknet [Farber 1997]. Luận án sử dụng hàm Sigmoid như công thức (4.7) làm hàm kích hoạt ở lớp ẩn và hàm Softmax như công thức (4.8) làm hàm kích hoạt ở lớp ra. Tất cả các mạng đều được huấn luyện với hệ số học (learning rate) khởi đầu là 0.05. Các vòng lặp huấn luyện được thực hiện liên tiếp và dừng lại khi tham số CV trên tập VOV-test giữa hai vòng liên tiếp lệch nhau là 0.001. Bảng 5-1 trình bày kết quả huấn luyện mạng. Từ kết quả ở Bảng 5-1 cho thấy chất lượng phân lớp của các mạng MLP với kích thước lớp đầu ra là 7 tốt hơn rất nhiều so với loại mạng có kích thước lớp ra là 19. Điều này chứng tỏ loại nhãn mức thanh điệu (TL) cho chất lượng phân lớp tốt hơn TSL. Từ kết quả này luận án đi đến lựa chọn kích thước lớp ra cho tất cả các mạng MLP trong các thử nghiệm tiếp theo sẽ là 7. Hay nói cách khác là chỉ sử dụng cơ sở dữ liệu đã được gán nhãn ở mức thanh điệu (TL) cho việc huấn luyện mạng và trích chọn đặc trưng TBNF.

Bảng 5-1: Kết quả huấn luyện mạng MLP trên hai loại nhãn TSL và TL

TT	Loại nhãn	Ký hiệu	Cấu hình mạng MLP (L1-L2-L3-L4-L5)	CV (%)
1	TSL	TSL-50-50	45-500-9-500-19	28.82
2		TSL-40-40	45-400-9-400-19	29.00
3		TSL-30-20	45-300-9-200-19	29.77
4		TSL-20-10	45-200-9-100-19	29.56
5		**TSL-10-05**	**45-100-9-050-19**	**30.07**
6	TL	TL-50-50	45-500-9-500-07	50.20
7		TL-20-10	45-200-9-100-07	53.40
8		TL-20-05	45-200-9-050-07	53.27
9		**TL-10-50**	**45-100-9-050-07**	**54.39**

5.4.2. Lựa chọn kích thước lớp Bottleneck (BN)

Bảng 5-2: Kết quả thử nghiệm với kích thước lớp BN thay đổi

TT	Đặc trưng	Kích thước lớp BN	ACC(%)
1	$TBNF_2$+MFCC	2	76.34
2	**$TBNF_3$+MFCC**	**3**	**76.53**
3	$TBNF_5$+MFCC	5	75.73
4	$TBNF_5$+MFCC	7	73.15
5	$TBNF_9$+MFCC	9	70.68
6	$TBNF_{11}$+MFCC	11	70.54
7	$TBNF_{13}$+MFCC	13	70.28
9	$TBNF_{15}$+MFCC	15	70.13

Để tìm ra kích thước lớp BN tối ưu cho tính toán đặc trưng TBNF luận án tiến hành huấn luyện các mạng MLP với kích thước lớp BN khác nhau. Cụ thể cấu hình các mạng MLP bao gồm 5 lớp. Kích thước các lớp L1, L2, L4, L5 đều giống nhau và bằng 45, 100, 50 và 7 theo thứ tự, đây là bộ kích thước cho kết quả CV tốt nhất khi kích thước lớp BN=9 ở thử nghiệm trước. Kích thước lớp BN được chọn trong bộ kích thước sau BN={2,3,5,7,9,11,13,15}. Các mạng này được huấn luyện trên bộ dữ liệu được gán nhãn mức thanh điệu TL tương tự như các thử nghiệm ở mục 5.4.1. Sau đó các mạng này được sử dụng để tính toán đặc trưng TBNF. Sau bước này ta thu được các đặc trưng thanh điệu xác suất liên tục tương ứng với bộ kích thước BN đã chọn là {$TBNF_2$, $TBNF_3$, $TBNF_5$, $TBNF_7$, $TBNF_9$, $TBNF_{11}$, $TBNF_{13}$, $TBNF_{15}$}. Để xác định loại đặc trưng nào cho kết quả nhận dạng tốt nhất các đặc trưng này được tổ hợp với đặc trưng MFCC ($TNBF_i$+MFCC, với i=2, 3, 5, 7, 9, 11, 13, 15). Sau đó 8 hệ thống sử dụng mô hình HMM được tiến hành thử nghiệm, các hệ thống này sử dụng 8 loại đặc trưng $TNBF_i$+MFCC ở trên làm đầu vào. Tất các hệ thống đều sử dụng cơ sở dữ liệu,

từ điển, mô hình ngôn ngữ và các bước huấn luyện tương tự như hệ thống HMM-2. Kết quả nhận dạng trên tập VOV-Test được trình bày ở Bảng 5-2.

Từ kết quả thử nghiệm này cho thấy đặc trưng $TBNF_3$ cho kết quả nhận dạng tốt nhất. Vì vậy mô hình mạng MLP với kích thước lớp BN là 3 sẽ được chọn để tính toán được trưng TBNF-MSD ở bước sau.

5.5. Thử nghiệm đặc trưng TBNF-MSD với mô hình MSD-HMM

5.5.1. Trích chọn đặc trưng TBNF-MSD

Sau hai bước thử nghiệm lựa chọn kích thước lớp Output và lớp BN ở các mục 5.4.1 và 5.4.2 luận án đã xác định được bộ kích thước tối ưu ban đầu của mạng MLP tương ứng với 5 lớp L1, L2, L3, L4, L5 là 45, 100, 3, 50 và 7. Mạng MLP này sẽ được sử dụng để tính toán đặc trưng TBNF-MSD_3 theo phương pháp đã đề xuất tại mục 5.2.3. Sau đó đặc trưng TBNF-MSD_3 này được tổ hợp với hai loại đặc trưng ngữ âm MFCC (MFCC+TBNF-MSD_3) và PLP (PLP+TBNF-MSD_3) để làm đầu vào cho mô hình MSD-HMM.

5.5.2. Dữ liệu, Từ điển, Mô hình ngôn ngữ

1. Dữ liệu huấn luyện: VOV.
2. Dữ liệu thử nghiệm: VOV-Test.
3. Mô hình ngôn ngữ: VOV-Bigram-LM.
4. Từ điển: Tonal-Dict.

5.5.3. Huấn luyện mô hình âm học MSD-HMM và kết quả thử nghiệm

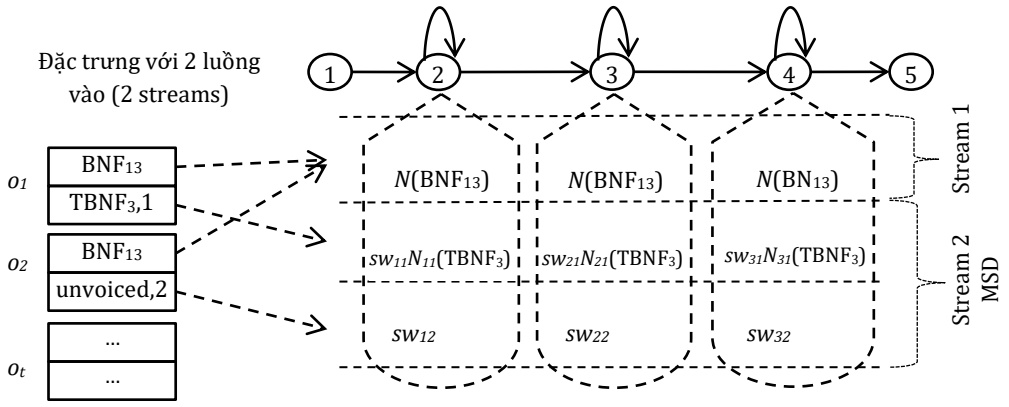
Bảng 5-3: Kết quả thử nghiệm TBNF-MSD với MSD-HMM
(Hệ thống MSD-HMM-4 đã được xây dựng ở mục 0 được đưa ra để so sánh)

TT	Hệ thống	Đặc trưng	ACC(%)
1	MSD-HMM-4	MFCC+AMDF	80.37
2	**TBNF-MSD-HMM-1**	**MFCC+TBNF-MSD_3**	**80.69**
3	TBNF-MSD-HMM-2	PLP+TBNF-MSD_3	80.23

Hai hệ thống được xây dựng tương ứng với hai loại đặc trưng đầu vào MFCC+TBNF-MSD_3 và PLP+TBNF-MSD_3. Các mô hình MSD-HMM của cả hai hệ thống được huấn luyện theo các bước và tham số tương tự như các hệ thống MSD-HMM ở mục 0. Hệ thống sau cùng được huấn luyện ở mức tri-phone với 2179 âm buộc, mỗi state sử dụng 16 thành phần trộn Gaussian. Kết quả thử nghiệm trên tập VOV-Test được trình bày ở Bảng 5-3.

Kết quả thử nghiệm cho thấy đặc trưng TBNF-MSD tương thích với mô hình MSD-HMM và cho chất lượng tốt hơn hai loại đặc trưng đã có là AMDF và NCC. Kết quả thanh điệu mới TBNF-MSD tăng chất lượng nhận dạng lên 0.32% tuyệt đối so với hệ thống sử dụng đặc trưng thanh điệu đã có AMDF, và thêm 2.99% tuyệt đối so với hệ thống cơ sở.

5.6. Mô hình tích hợp BNF, TBNF-MSD và MSD-HMM



Hình 5-6: Mô hình MSD-HMM cho đặc trưng kết hợp BNF_{13}+$TBNF\text{-}MSD_3$

Các thử nghiệm sử dụng đặc trưng BNF, TBNF-MSD và mô hình MSD-HMM trong luận án này đã cho thấy đặc trưng được trích chọn bởi mạng nơron đã làm tăng đáng kể chất lượng nhận dạng. Mô hình MSD-HMM hoàn toàn tương thích và có hiệu quả với nhận dạng tiếng Việt trên tập âm vị có thông tin thanh điệu. Ở thử nghiệm cuối cùng này luận án sẽ tiến hành tích hợp tất cả các kỹ thuật này vào một hệ thống duy nhất. Cụ thể như sau. Một hệ thống sử dụng mô hình MSD-HMM 5 trạng thái với hai luồng đầu vào, trong đó luồng thứ nhất dành cho đặc trưng BNF. Luận án sử dụng đặc trưng BNF_{13} được tính toán từ đặc trưng đầu vào là MFCC, đây là loại đặc trưng được trích chọn từ mô hình mạng MLP (ký hiệu là MFCC-4-3-13 ở Bảng 4-3) có kích thước lớp BN là 13 đã cho kết quả nhận dạng tốt nhất ở các thử nghiệm về BNF. Luồng thứ nhất này không áp dụng mô hình MSD do đặc trưng BNF là đặc trưng liên tục. Luồng thứ hai dành cho đặc trưng thanh điệu $TBNF\text{-}MSD_3$. Do $TBNF\text{-}MSD_3$ là dữ liệu chứa cả giá trị liên tục và rời rạc nên luồng thứ hai này sẽ được áp dụng mô hình MSD với hai không gian $\{\Omega_1, \Omega_2\}$. Trong đó Ω_1 là không gian số thực có số chiều là 3 tương ứng với kích thước của giá trị $TBNF_3$. Ω_1 là không gian rời rạc với số chiều là 0 chỉ có một giá trị duy nhất là "unvoiced" dành cho các giá trị "unvoiced" trong đặc trưng $TBNF\text{-}MSD_3$. Mô hình MSD-HMM sử dụng đặc trưng kết hợp BNF_{13}+$TBNF\text{-}MSD_3$ này được mô tả ở Hình 5-6.

Các mô mình MSD-HMM này được huấn luyện trên tập dữ liệu VOV sử dụng từ điển Tonal-Dict theo các bước và tham số tương tự như các hệ thống MSD-HMM ở Chương 3. Kết quả thử nghiệm trên tập VOV-Test với mô hình ngôn ngữ VOV-Bigram-LM được trình bày ở Bảng 5-4.

Bảng 5-4: Kết quả thử nghiệm MSD-HMM với đặc trưng BNF_{13}+TBNF-MSD_3
(Hệ thống 1 sử dụng đặc trưng BNF_{13} đã được xây dựng ở mục 0 sử dụng mạng MLP có cấu hình 585x4000x13x3000x463 với đặc trưng đầu vào của mạng là MFCC)

TT	Đặc trưng	ACC(%)
1	BNF_{13}	84.18
2	BNF_{13}+TBN-MSD_{13}	**84.54 (+0.36)**

5.7. Kết luận chương

1. Các kết quả thử nghiệm ở chương này cho thấy phương pháp đã đề xuất để việc tính toán đặc trưng thanh điệu cho mô hình MSD-HMM sử dụng mạng nơron đã làm tăng chất lượng nhận dạng. Cụ thể loại đặc trưng thanh điệu này tốt hơn khoảng 0.3% tuyệt đối (khoảng 2% tương đối) so với các đặc trưng thanh điệu đã có như AMDF và NCC. Kết quả này cho thấy đặc trưng thanh điệu rõ ràng là một trong các nhân tố cùng với mô hình thanh điệu để tối ưu mô hình nhận dạng cho tiếng Việt. Và việc TBNF_MSD cho kết quả tốt hơn AMDF và NCC cho thấy cần thiết phải nghiên cứu các biện pháp nâng cao chất lượng cho đặc trưng thanh điệu.
2. Mô hình tích hợp BNF, TBNF với MSD-HMM đã cho chất lượng tốt nhất so với mô hình HMM sử dụng đặc trưng MFCC/PLP+AMDF/NCC. Cụ thể qua thử nghiệm kết hợp đặc trưng BNF và đặc trưng thanh điệu TBNF-MSD với mô hình MSD-HMM đã làm tăng chất lượng nhận dạng lên 6.23% tuyệt đối so với hệ thống cơ sở, và 4.17% tuyệt đối so với hệ thống sử dụng đặc trưng chưa tăng cường MFCC+AMDF.

5.8. Các bài báo đã công bố liên quan đến nội dung của chương

1. Nguyen Van Huy, Luong Chi Mai, Vu Tat Thang, *Adapting bottle neck feature to multi space distribution for Vietnamese speech recognition*, Conference of the Oriental chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (OCOCOSDA), Phuket-Thailan, Oct-2014.

Kết luận

❖ ***Các công việc chính đã thực hiện của luận án***

- Đã nghiên cứu tổng quan về tình hình nghiên cứu nhận dạng tiếng nói và nhận dạng tiếng Việt. Từ kết quả nghiên cứu này luận án đã xác định được các vấn đề còn tồn tại trong nhận dạng tiếng Việt từ vựng lớn.
- Đã nghiên cứu và trình bày tổng quan về các thành phần chính của một hệ thống nhận dạng tiếng nói. Nội dung nghiên cứu chính của luận án tập trung vào việc cải tiến các phần liên quan đến trích chọn đặc trưng và mô hình âm học trong hệ thống nhận dạng tiếng nói.
- Đã nghiên cứu cơ bản về đặc tính ngữ âm tiếng Việt. Luận án đã trình bày cấu trúc ngữ âm, tập âm vị, tập thanh điệu của tiếng Việt. Từ kết quả này luận án đã đề xuất phương pháp xây dựng mô hình nhận dạng tiếng Việt từ vựng lớn phát âm liên tục bằng cách sử dụng tập âm vị có thanh điệu làm bộ đơn vị nhận dạng. Với cách tiếp cận này mô hình nhận dạng của luận án có thể nhận dạng tất cả các từ có thể có của tiếng Việt, mặc dù từ đó có thể không cần có trong dữ liệu huấn luyện. Đồng thời luận án cũng đã đề xuất một giải thuật tạo từ điển âm vị tự động cho tiếng Việt áp dụng cho nhận dạng tiếng Việt từ vựng lớn. Giải thuật VN-G2P này có thể được áp dụng để tạo từ điển ngữ âm cho bất kỳ tập từ vựng tiếng Việt đầu vào nào.
- Nghiên cứu đưa ra được mô hình thanh điệu cho nhận dạng tiếng Việt từ vựng lớn phát âm liên tục theo phương pháp tích hợp nhận dạng thanh điệu và nhận dạng âm vị trong cùng một pha. Việc tích hợp này được thực hiện bằng cách tổ hợp âm chính trong các âm tiết của tiếng Việt với thông tin thanh điệu để tạo thành âm chính có thanh điệu.
- Đã nghiên cứu và trình bày lý thuyết về mô hình phân bố đa không gian MSD-HMM. Đã nghiên cứu và đề xuất loại mô hình MSD-HMM áp dụng cho nhận dạng tiếng Việt có thanh điệu. Bao gồm quy trình và phương pháp cấu hình mô hình MSD-HMM, phương pháp huấn luyện mô hình. Luận án cũng đã trình bày các phương pháp trích chọn đặc trưng thanh điệu cho loại mô hình MSD-HMM này, từ đó đã tìm loại đặc trưng thanh điệu tương thích nhất.
- Đã nghiên cứu và trình bày lý thuyết về phương pháp trích chọn đặc trưng Bottleneck và áp dụng cho nhận dạng tiếng Việt. Luận án đã trình bày quy trình và phương pháp tính toán đặc trưng BNF, phương pháp gán nhãn và huấn luyện mạng MLP, phương pháp chuẩn hóa và tối ưu đặc trưng BNF cho tiếng Việt. Kết quả của nghiên cứu này cũng được áp dụng để xây dựng

module nhận dạng tiếng Việt trong dự án quốc tế VoiceTra4U[7] về phát triển ứng dụng dịch tiếng nói tự động của 32 quốc gia mà Viện công nghệ thông tin (IOIT) là một đại diện của Việt Nam đang tham gia.

- Đã nghiên cứu và đề xuất phương pháp trích chọn đặc trưng thanh điệu cho mô hình MSD-HMM sử dụng mạng nơron MLP. Luận án đã trình bày phương pháp trích chọn đặc trưng, tối ưu hóa đặc trưng, kỹ thuật gán nhãn dữ liệu, phương pháp chuẩn hóa và tích hợp đặc trưng này với mô hình MSD-HMM cho tiếng Việt.
- Đã nghiên cứu và đề xuất kết hợp các kỹ thuật trích chọn đặc trưng BNF và đặc trưng thanh điệu TBNF sử dụng mạng nơron MLP với mô hình MSD-HMM vào một hệ thống duy nhất cho nhận dạng tiếng Việt.

Với các công việc đã thực hiện ở trên thì luận án đã hoàn thành các mục tiêu chính đã đặt ra ở Chương 1. Cụ thể là:

1. Đã đưa ra được mô hình cho hệ thống nhận dạng tiếng Việt từ vựng lớn phát âm liên tục. Từ kết quả thí nghiệm cho thấy mô hình này cho kết quả tốt hơn mô hình không có thanh điệu.
2. Đã đưa ra được phương pháp áp dụng mô hình MSD-HMM trong việc mô hình hóa đặc trưng thanh điệu tiếng Việt theo đúng bản chất đứt gãy. Và việc áp dụng mô hình này cũng đã cho kết quả tốt hơn mô hình HMM truyền thống.
3. Đã đưa ra được phương pháp áp dụng mạng nơron để tính toán Bottleneck cho tiếng Việt, đồng thời dựa vào kết quả này luận án cũng đã đề xuất một phương pháp tính toán đặc trưng cải tiếng mới TBNF cho tiếng Việt. TBNF đã cho kết quả tốt hơn các phương pháp AMDF, NCC đã có.
4. Đã đưa ra được mô hình tích hợp BNF, TBNF và MSD-HMM cho tiếng Việt.

❖ ***Các kết luận và thảo luận từ các kết quả thử nghiệm của luận án***

- Đặc trưng thanh điệu và tập âm vị có thông tin thanh điệu là các thành phần quan trọng ảnh hưởng đến chất lượng của mô hình nhận dạng tiếng Việt có thanh điệu. Qua các thử nghiệm trên bộ dữ liệu kích thước lớn cũng như trung bình và trên các bộ công cụ khác nhau là HTK và Kaldi đều cho thấy đặc trưng thanh điệu giúp làm tăng chất lượng nhận dạng thêm khoảng trên 3% tuyệt đối và tập âm vị có thông tin thanh điệu làm tăng chất lượng nhận dạng thêm khoảng trên 1.5% tuyệt đối. Tương tự như các nghiên cứu trên

[7]http://www.ustar-consortium.com/app/app.html

các ngôn ngữ Mandarin, Cantonese, Thai cho thấy rõ ràng thanh điệu là yếu tố quan trọng trong việc tối ưu mô hình nhận dạng. Tuy nhiên trong phương pháp xây dựng bộ đơn vị cho mô hình thanh điệu mà luận án đã thực hiện thì mới có 6 thanh điệu của tiếng Việt được sử dụng. Trong phạm vi luận án này chưa xét đến sự biến đổi của thanh điệu khi đi cùng với các phụ âm cuối đóng (stop consonant) như /p/, /t/, /k/, trong trường hợp này sẽ có 8 thanh điệu. Việc bổ sung thông tin thanh điệu vào tập âm vị và bổ sung đặc trưng thanh điệu cùng với đặc trưng ngữ âm làm đặc trưng đầu vào đã làm tăng độ phức pháp tính toán cho hệ thống. Cụ thể ở đây tập âm vị tăng từ 45 lên 154 và cần có thêm một khâu tính toán đặc trưng thanh điệu. Nếu hệ thống nhận dạng tính đến tốc độ và không yêu cầu về chất lượng tối ưu thì có thể bỏ qua thông tin thanh điệu ở mô hình âm học và đặc trưng đầu vào nếu chấp nhận độ chính xác giảm đi khoảng 5%. Khi đó việc xây dựng mô hình nhận dạng cho tiếng Việt hoàn toàn có thể áp dụng các mô hình đã có trên các ngôn ngữ phổ dụng không có thanh điệu như tiếng Anh, Đức mà không cần quan tâm đến đặc tính thanh điệu của tiếng Việt. Việc nhận dạng thanh điệu có thể chuyển sang mô hình ngôn ngữ.

- Mô hình MSD-HMM có hiệu quả với tiếng Việt. Mô hình MSD-HMM có khả năng mô tả đúng đặc tính vật lý của đặc trưng thanh điệu đó là liên tục trong vùng hữu thanh và đứt gãy trong vùng vô thanh. Mô hình này đã giúp làm tăng chất lượng nhận dạng thêm khoảng 15% tương đối so với mô hình HMM truyền thống. Kết quả này tương đồng với nghiên cứu trên ngôn ngữ Mandarin [Y. a. Qian 2009] [Chong-Jia 2011] (khoảng 17%). Như vậy việc nghiên cứu tìm ra loại mô hình có khả năng mô hình hóa thông tin thanh điệu là một yếu tố quan trọng trong việc nâng cao chất lượng nhận dạng cho tiếng Việt. Đồng thời cùng với kết quả nghiên cứu trên tiếng Mandarin cho thấy việc mô hình hóa đúng bản chất đứt gãy của đặc trưng thanh điệu cho kết quả tốt hơn loại đặc trưng được bổ sung các giá trị "nhận tạo" vào vùng vô thanh.
- Phương pháp tăng cường đặc trưng sử dụng mạng nơron có hiệu quả với tiếng Việt. Phương pháp tính toán đặc trưng này đã giúp tăng chất lượng cho cả hai loại đặc trưng ngữ âm và đặc trưng thanh điệu. Với đặc trưng ngữ âm BNF đã giúp tăng thêm khoảng 29% tương đối so với hai loại đặc trưng đã có MFCC và PLP, và đặc trưng thanh điệu TBNF cải tiến mới đã giúp tăng thêm khoảng 2% tương đối so với hai loại đặc trưng thanh điệu đã có AMDF và NCC. Cả BNF và TBNF được trích chọn dựa theo đặc tích ngữ âm của tiếng Việt. Cụ thể BNF được tính toán thông qua mạng nơron đã

được huấn luyện để phân lớp các âm vị đã tích hợp 6 thanh điệu tiếng Việt, TBNF sử dụng mạng nơron đã được huấn luyện để phân lớp 6 thanh điệu tiếng Việt. Từ kết quả thử nghiệm cho thấy rõ ràng là mạng nơron không chỉ có hiệu quả trong việc phân lớp mà còn có hiệu quả như một mô hình biến đổi đặc trưng. Tuy nhiên việc áp dụng BNF, hoặc TBNF cũng làm gia tăng độ phức tạp tính toán cho hệ thống. Nhưng với 29% tăng chất lượng trong nghiên cứu này, và khoảng 10% tăng chất lượng trên các công bố trên các ngôn ngữ khác như tiếng Anh, Đức cho thấy đây là một mô hình quan trọng để tối ưu đặc trưng. Tham số của mạng tính toán BNF và TBNF tùy thuộc vào từng ngôn ngữ vào kích thước bộ dữ liệu huấn luyện cụ thể. Hai yếu tố quan trọng ảnh hưởng đến chất lượng đặc trưng BNF và TBNF là cấu hình mạng MLP và chất lượng của việc gán nhãn dữ liệu để huấn luyện mạng.

- Mô hình tích hợp BNF, TBNF với MSD-HMM cho kết quả tối ưu nhất so với các mô hình khác mà luận án đã xây dựng. Kết quả này cho thấy mô hình MSD-HMM thực sự hiệu quả hơn mô hình HMM khi sử dụng với đặc trưng thanh điệu đứt gãy. Các đặc trưng tăng cường BNF và đặc trưng cải tiến TBNF đã giúp cho mô hình MSD-HMM đạt chất lượng tốt hơn so với việc sử dụng các đặc trưng chưa tăng cường như MFCC, PLP, AMDF và NCC (tốt khoảng 19% tương đối). Như vậy việc nghiên cứu để tìm ra các mô hình tăng cường chất lượng đặc trưng, tối ưu cho MSD-HMM là đúng đắn và rất cần thiết.

❖ ***Hướng phát triển***

- Việc sử dụng tập âm vị có thông tin thanh điệu làm gia tăng kích thước của hệ thống từ 54 âm vị đơn lên 154 âm vị đơn. Và việc bổ sung thông tin thanh điệu mới chỉ được áp dụng trên âm chính của âm tiết. Cần có các nghiên cứu tiếp theo để tìm ra tập âm vị tối ưu, vị trí bổ sung thông tin thanh điệu tối ưu cho tiếng Việt, hoặc các phương pháp làm giảm kích thước tập âm vị thông qua các kỹ thuật phân cụm.
- Đặc trưng thanh điệu TBNF hiện tại cho chất lượng tăng còn thấp, chỉ khoảng 2% tương đối. Nên cần tiếp tục được nghiên cứu để nâng cao chất lượng. Một số kỹ thuật biến đổi đặc trưng như LDA, MLLT có thể được áp dụng trước khi áp dụng phương pháp này để nâng cao chất lượng.
- Trong luận án này đặc trưng BNF và TBNF mới chỉ được trích chọn từ các mạng MLP 5 lớp. Trong khi hiện nay các kỹ thuật mạng MLP học sâu với nhiều lớp ẩn hơn đã mang lại nhiều kết quả tích cực trong nhiều lĩnh vực khác nhau. Trong các nghiên cứu tiếp theo thì công nghệ mạng học sâu

(Deep Learning) cần được áp dụng để nâng cao chất lượng cho đặc trưng BNF và TBNF.

- Mô hình MSD-HMM trong nghiên cứu này chưa áp dụng các kỹ thuật tối ưu tham số. Vì vậy cần nghiên cứu và thử nghiệm áp dụng các kỹ thuật huấn luyện tối ưu như ước lượng tham số phụ thuộc người nói (SAT), tối đa tính tự tương quan giữa các đặc trưng thuộc cùng một lớp (Maximum Likelihood),…

Các đóng góp chính luận án

Đã đề xuất kiến trúc hệ thống nhận dạng tiếng Việt liên tục từ vựng lớn có thể tích hợp thông tin thanh điệu.

1) Đưa ra phương pháp áp dụng mô hình MSD-HMM để mô hình hóa tập âm vị tiếng Việt có thông tin thanh điệu dựa trên đặc trưng thanh điệu đầu vào vẫn giữ nguyên đặc tính đứt gãy của nó.
2) Đề xuất phương pháp cải tiến đặc trưng thanh điệu mới (TBNF) sử dụng mạng nơron MLP. TBNF biểu diễn đúng đặc tính đứt gãy của đặc trưng thanh điệu và tương thích với mô hình MSD-HMM.
3) Đưa ra mô hình kết hợp giữa MSD-HMM với đặc trưng BNF và đặc trưng thanh điệu TBNF cho nhận dạng tiếng Việt.

Một số đóng góp khác của luận án

1) Đề xuất giải thuật tạo từ điển ngữ âm có thông tin thanh điệu tự động cho tập dữ liệu đầu vào tiếng Việt bất kỳ.
2) Đề xuất thuật toán gán nhãn thanh điệu cho dữ liệu dựa trên kỹ thuật gán nhãn âm vị kết hợp với kỹ thuật phát hiện vùng hữu thanh và vô thanh.

Danh mục các công trình khoa học đã công bố của tác giả và cộng sự

A. Tạp chí quốc gia

1. Công bố nghiên cứu áp dụng đặc trưng Bottleneck cho nhận dạng tiếng Việt trên tạp chí Tin học & Điều khiển năm 2013.

   Nguyen Van Huy, Luong Chi Mai, Vu Tat Thang, *Áp dụng Bottle neck Feature cho nhận dạng tiếng Việt*, Journal of Computer Science and Cybernetics, Vietnam, ISSN 1813-9663, Vol 29, No 4, Oct-2013.

2. Công bố nghiên cứu áp dụng mô hình MSD-HMM cho nhận dạng tiếng Việt trên tạp chí Tin học & Điều khiển năm 2014.

   Nguyen Van Huy, Luong Chi Mai, Vu Tat Thang, Do Quoc Truong, *Vietnamese recognition using tonal phoneme based on multi space distribution*, Journal of Computer Science and Cybernetics, Vietnam, ISSN 1813-9663, Vol 30, No 1, Jan-2014.

3. Công bố nghiên cứu phương pháp tối ưu đặc trưng Bottleneck áp dụng cho nhận dạng tiếng Việt trên tạp chí Khoa học Công nghệ - ĐH Thái Nguyên năm 2015.

   Nguyễn Văn Huy, *Nâng cao chất lượng đặc trưng bottle neck cho nhận dạng tiếng Việt*, Tạp chí Khoa học và Công nghệ Đại học Thái Nguyên, ISSN 1859-2171, Tập 137, Số 07, 2015.

B. Hội thảo quốc tế

1. Công bố nghiên cứu áp dụng đặc trưng Bottleneck cho nhận dạng tiếng Anh tại cuộc thi về các hệ thống nhận dạng và dịch tiếng nói tự động quốc tế đươc tổ chức tại Đức năm 2013. Đây là nghiên cứu thử nghiệm đầu tiên của NCS về Bottleneck trước khi nghiên cứu áp dụng cho tiếng Việt.

   Kevin Kilgour, Christian Mohr, Michael Heck, Quoc Bao Nguyen, **Van Huy Nguyen**, Evgeniy Shin, Igor Tseyzer, Jonas Gehring, Markus Muller, Matthias Sperber, Sebastian Stuker and Alex Waibel , *The 2013 KIT IWSLT Speech-to-Text Systems for German and English*, International Workshop on Spoken Language Translation (IWSLT), Germany, Dec-2013.

2. Công bố nghiên cứu tập âm vị có thông tin thanh điệu áp dụng cho nhận dạng tiếng Việt tại hội thảo "Automatic Speech Recognition and Understanding (ASRU)" tại Czech năm 2013.

   Jonas Gehring, Kevin Kilgour, Quoc Bao Nguyen, **Van Huy Nguyen**, Florian Metze, Zaid A. W. Sheikh, Alex Waibel , *Models of tone for tonal and non-tonal languages*, IEEE Automatic Speech Recognition and Understanding (ASRU), Czech Republic, Dec-2013.

3. Công bố nghiên cứu cải tiến phương pháp Bottleneck để trích trọn đặc trưng thanh điệu cho mô hình MSD-HMM áp dụng cho tiếng Việt tại hội thảo “Conference of the Oriental chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (OCOCOSDA)” ở Thái Lan năm 2014.

   Nguyen Van Huy, Luong Chi Mai, Vu Tat Thang, *Adapting bottle neck feature to multi space distribution for Vietnamese speech recognition*, Conference of the Oriental chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (OCOCOSDA), Phuket-Thailan, Oct-2014.

4. Công bố phương pháp tạo từ điển âm vị tự động cho tiếng Việt từ dữ liệu văn bản đầu vào (Graphphem to Phoneme) tại hội thảo “Conference of the Oriental chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (OCOCOSDA)" ở Thượng Hải năm 2015.

   Van Huy Nguyen, Chi Mai Luong, Tat Thang Vu, *Tonal phoneme based model for Vietnamese LVCSR*, IEEE Conference of the Oriental chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (OCOCOSDA), Shanghai-China, Oct-2015.

5. Công bố nghiên cứu thử nghiệm áp dụng đặc trưng thanh điệu để xây dựng hệ thống nhận dạng tiếng Anh cho cuộc thi về các hệ thống nhận dạng và dịch tiếng nói tự động quốc tế được tổ chức tại Việt Nam năm 2015. Nghiên cứu này thử nghiệm đầu tiên cho dự kiến ap dụng mô hình MSD-HMM cho tiếng Anh của NCS.

   Van Huy Nguyen, Quoc Bao Nguyen, Tat Thang Vu, Chi Mai Luong, The IOIT English ASR system for IWSLT 2015, International Workshop on Spoken Language Translation (IWSLT), Da Nang, Vietnam, Dec-2015.

Tài liệu tham khảo

Tiếng Việt

Chừ, Mai Ngọc and Nghiệu, Vũ Đức and Phiến, Hoàng Trọng. *Cơ sở ngôn ngữ học và tiếng Việt.* Việt Nam: NXB Giáo Dục, 1997.

Đức, Đặng Ngọc. *Mạng nơron và mô hình Markov ẩn trong nhận dạng tiếng Việt.* Hà Nội: Luấn án tiến sỹ, Trường ĐH Khoa học tự nhiên – ĐH Quốc gia hà Nội, 2003.

Khang, Bạch Hưng. *Tổng Hợp và Nhận dạng tiếng Việt - Đề tài cấp nhà nước.* Hà Nội: Viện Công Nghệ Thông Tin, 2004.

Tiếng Anh

Ambra, N. and Catia, C. and Wilhelmus, S. "Automatic Speech Recognition for second language learning: How and why it actually works." *International Congress of Phonetic Sciences (ICPhS).* Barcelona, 2003.

Anastasakos, T. and McDonough, J. and Makhoul, J. "Speaker adaptive training: a maximum likelihood approach to speaker normalization." *Acoustics, Speech and Signal Processing (ICASSP).* Munich, 1997. 1043 – 1046.

Bengio, Yoshua and Rejean, Ducharme and Pascal, Vincent and Christian, Jauvin. "A neural probabilistic language." *Machine Learning Research*, 2003: 1137–1155.

Chen, C.J. and Haiping Li and Liqin Shen and Guokang Fu. "Recognize tone languages using pitch information on the main vowel of each syllable." *Acoustics, Speech, and Signal Processing (ICASSP).* Salt Lake City, UT: IEEE, 2001. 61-64.

Chong-Jia, Ni and Wen-Ju, Liu and Bo, Xu. "Prosody Dependent Mandarin Speech Recognition." *International Joint Conference on Neural Networks.* California, USA: IEEE, 2011. 197-201.

Christian, Plahl and Ralf, Schluter and Hermann, Ney. "Cross-lingual Portability of Chinese and English Neural Network Features for French and German LVCSR." *Automatic Speech Recognition & Understanding (ASRU).* Waikoloa, HI, USA: IEEE, 2011. 371-376.

Chuong, Nguyen Thien. *Automatic speech recognition of Vietnamese.* PhD Thesis, Technical University of Liberec, Czech Republic, 2014.

Chừ, Mai Ngọc and Nghiệu, Vũ Đức and Phiến, Hoàng Trọng. *Cơ sở ngôn ngữ học và tiếng Việt.* Việt Nam: NXB Giáo Dục, 1997.

Daniel, Povey and Arnab, Ghoshal and Gilles, Boulianne and Lukas, Burget and Ondrej, Glembek and Nagendra, Goel and Mirko, Hannemann and Petr, Motlicek and Yanmin, Qian and Petr, Schwarz and Jan, Silovsky and Georg, Stemmer and Karel, Vesely. "The Kaldi Speech Recognition Toolkit." *Automatic Speech Recognition and Understanding.* Hawaii, US, 2011.

Daniel, Povey and Lukas, Burget and Mohit, Agarwal and et. "Subspace Gaussian Mixture Models for Speech Recognition." *Acoustics Speech and Signal Processing (ICASSP).* Texas, USA: IEEE, 2010.

Dixon, P.R. and Hori, C. and Kashioka, H. "Development of the SprinTra WFST Speech Decoder." *NICT Research Journal*, 2012: Journal.

Đức, Đặng Ngọc. *Mạng nơron và mô hình Markov ẩn trong nhận dạng tiếng Việt.* Hà Nội: Luấn án tiến sỹ, Trường ĐH Khoa học tự nhiên – ĐH Quốc gia hà Nội, 2003.

Farber, P. *Quicknet on multispert: fast parallel neural network training.* TR-97-047, ICSI, 1997.

Fatemeh, Sadat Saleh and Boshra, Shams and Hossein, Sameti and Soheil, Khorram. "An Automatic Prosodic Event Detector Using MSD HMMs for Persian Language." *Artificial Intelligence and Signal Processing, ISBN 978-3-319-10848-3*, 2013: 234-240.

Ferreira, E. and Nocera, P. and Goudi, M. and Thi, N.D.D. "YAST: A Scalable ASR Toolkit Especially Designed for Under-Resourced Languages." *Asian Language Processing (IALP).* Hanoi: IEEE, 2012. 141 - 144.

Florian, Honig and Georg, Stemmer and Christian, Hacker and Fabio, Brugnara. "Revising Perceptual Linear Prediction (PLP)." *INTERSPEECH.* Lisbon, Portugal, 2005.

Frederick, Jelinek and Robert, L. Mercer. "Interpolated Estimation of Markov Source Parameters from Sparse Data." *Pattern Recognition in .* The Netherlands: North-Holland, 1980. 381-397.

Gales, M. and Young, S. "The Application of Hidden Markov Models in Speech Recognition." *Signal Processing*, 2007: 195-304.

Gehring, J. and Miao, Y. and Metze, F. and Waibel, A. "Extracting deep bottleneck features using stacked auto-encoders." *Acoustics, Speech and Signal Processing (ICASSP).* Vancouver, 2013. 3377 – 3381.

Good, I. J. "The population frequencies of species and the estimation of population." *Biometrika, Vol. 40, No. 3/4*, 1953: 237-264.

Grézl, Frantisek and Fousek, Petr. "Optimizing Bottel-neck features for LVCSR." *ICASSP.* Las Vegas: IEEE, 2008. 4729-4732.

Grézl, Frantisek and Karafiát, Martin and Kontár, Stanislav and Cernocký, Jan. "Probabilistic and Bottle-Neck Features for LVCSR of Meetings." *ICASSP.* Honolulu: IEEE, 2007. IV-757- IV-760.

Haeb-Umbach, R. and Ney, H. "Linear discriminant analysis for improved large vocabulary continuous speech recognition." *Acoustics, Speech, and Signal Processing (ICASSP).* California, USA, 1992. 13-16.

Hary, Myron and. "Average Magnitude Difference Function Pitch Extractor." *IEEE transactions on Acoustic, Speech, and Signal processing*, 1974.

Hermansky, H. and Daniel, P.W. Ellis and Sangita, Sharma. "Tandem connectionist feature extraction for conventional HMM systems." *Acoustics, Speech, and Signal Processing (ICASSP).* Istanbul: IEEE, 2000. 1635-1638.

Hermansky, H. "Perceptual linear predictive (PLP) analysis of speech." *Acoustical Society of America Journal*, 1990: 1738–1752.

Hong Quang, Nguyen and Nocera, P. and Castelli, E. and Van Loan, T. "Tone recognition of Vietnamese continuous speech using hidden Markov model." *Communications and Electronics - ICCE.* Hoi an: IEEE, 2008. 235 - 239.

Janin, A. and Andreas, Stolcke and Xavier, Anguera and Kofi, Boakye and Özgür, Çetin and Joe, Frankel and Jing, Zheng. "Machine Learning for Multimodal Interaction." *The ICSI-SRI Spring 2006 meeting recognition system, Lecture Notes in Computer Science*, 2006: 444-456.

Jonas, G. and Kevin, K. and Quoc Bao, N. and Van Huy, N. and Florian, M. and Zaid, A. W. and Alex, W. *Models of tone for tonal and non-tonal languages.* Czech republic: Automatic Speech Recognition and Understanding (ASRU), IEEE, 2013.

Juang, B. H. and Rabiner, L. R. "Hidden Markov Models for Speech Recognition,." *Technometrics*, 1991: 251-272.

Jurafsky, Daniel and Martin, James H. *Speech and Language Processing - 2nd Edition.* Prentice Hall, ISBN-13: 978-0131873216, ISBN-10: 0131873210, 2008.

Kasi, K. and Zahorian, S. A. "Yet another algorithm for pitch tracking." *IEEE International Symposium on Circuits and Systems.* Arizona: IEEE, 2002. 361-364.

Katz, S. "Estimation of probabilities from sparse data for the language model component of a speech recognizer." *Acoustics, Speech and Signal Processing.* IEEE, 1987. 400 - 410.

Kevin, K. and Christian, M, and Michael, H., Quoc Bao, N. and Van Huy, N. and Evgeniy, S. and Igor, T. and Jonas, G. and Markus, M. and Matthias, S. and Sebastian, S. and Alex, W.l. "The 2013 KIT IWSLT Speech-to-Text Systems for German and English." *International Workshop on Spoken Language Translation (IWSLT).* Germany, 2013.

Kevin, K. and Heck, M. and Muller, Markus and Sperber, Matthias and Stuker, Sebastian and Waibe, Alex. "The 2014 KIT IWSLT Speech-to-Text Systems for English, German and Italian." *The International Workshop on Spoken Language Translation (IWSLT).* Lake Tahoe, USA, 2014.

Kevin, Kilgour and Saam, C. and Mohr, C. and Stuker, S. and Waibel, A. "The 2011 KIT Quaero Speech-to-text system for Spanish." *International Workshop on Spoken Language Translation (IWSLT).* San Francisco, 2011.

Kriesel, D. *A Brief Introduction to Neural Networks.* University of Bonn in Germany, 2005.

Kunikoshi, A. and Yao, Qian and Soong, F. and Minematsu, N. "F0 modeling and generation in voice conversion." *Acoustics, Speech and Signal Processing (ICASSP).* Prague, 2011. 4568 – 4571.

Kwanchiva, Thangthai and Ananlada, Chotimongkol and Chai, Wutiwiwatchai. "A Hybrid Language Model for Open-Vocabulary Thai LVCSR." *INTERSPEECH.* Lyon, France: IEEE, 2013.

Khang, Bạch Hưng. *Tổng Hợp và Nhận dạng tiếng Việt - Đề tài cấp nhà nước.* Hà Nội: Viện Công Nghệ Thông Tin, 2004.

Lei, Xin. *Modeling Lexical Tones for Mandarin Large Vocabulary Continuous Speech Recognition.* USA: University of Washington, 2006.

Levinson, N. "The Wiener RMS error criterion in filter design and prediction." *J. Math. Physics*, 1947: 261–278.

Martin, Karafiat and Lukas, Burget and Pavel, Matejka and Ondrej, Glembek. "iVector-Based Discriminative Adaptation for Automatic Speech Recognition." *Automatic Speech Recognition and Understanding (ASRU).* Waikoloa: IEEE, 2011. 152-157.

Matsuda, S. and Xinhui Hu and Shiga, Y. and Kashioka, H. and Hori, C. and Yasuda, K. and Okuma, H. and Uchiyama, M. and Sumita, E. and Kawai, H. and Nakamura, S. "Multilingual Speech-to-Speech Translation System: VoiceTra." *Mobile Data Management (MDM).* Milan: IEEE, 2013. 229 - 233.

Miyajima, C. and Hattori Y. and Tokuda, K. and Masuko and Takashi and Kobayashi, T. and Kitamura, T. "Speaker identification using Gaussian mixture models based on multi-space probability distribution." *Acoustics, Speech, and Signal Processing (ICASSP).* Salt Lake City, UT, 2001. 433 – 436.

Muda, Lindasalwa and Begam, Mumtaj and Elamvazuthi, I. "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques." *journal of computing, V.2, No.2, ISSN 2151-9617*, 2010.

Ney, R. K. and Hermann. "Improved backing-off for n-gram language modeling." *Acoustics, Speech and Signal Processing.* Detroit: IEEE, 1995. 181-184.

Ney, Reinhard Kneser and Hermann. *The IEEE International Conference on Acoustics, Speech and Signal Processing.* n.d.

Ochiai, T. and Matsuda, S. and Lu, Xugang and Hori, C. and Katagiri, S. "Speaker Adaptive Training using Deep Neural Networks." *Acoustics, Speech and Signal Processing (ICASSP).* Florence, 2014. 6349 – 6353.

Oura, Keiichiro. *List of modifications made in HTS (forversion2.2).* Japan: Nagoya Institute of Technology, 2011.

Plahl, Christian and Schluter, Ralf and Ney, Hermann. "Improved Acoustic Feature Combination for LVCSR by Neural Networks." *INTERSPEECH.* Italy: IEEE, 2011.

Psutka, Josef V. "Benefit of Maximum Likelihood Linear Transform (MLLT) Used at Different Levels of Covariance Matrices Clustering in ASR Systems." *Text, Speech and Dialogue, 10th International Conference (TSD).* Czech Republic, 2007.

Qian, Y. and Soong Frank, K. "A Multi-Space Distribution (MSD) and two-stream tone modeling approach to Mandarin speech recognition." *Speech Communication.* Beijing China, 2009. 1169 - 1179.

Qian, Yao and Frank, K. Soong. "A Multi-Space Distribution (MSD) and two-stream tone modeling approach to Mandarin speech recognition." *Speech Communication, Vol 51*, 2009: 1169–1179.

Qian, Yao and Frank, Soong and Yining ,Chen and Min, Chu. "An HMM-Based Mandarin Chinese Text-To-Speech System." *Computer Science, Volume 4274* , 2006: 223-232.

Quoc Cuong, Nguyen and Yen, Pham Thi Ngoc and Castelli, E. "Shape vector characterization of Vietnamese tones and application to automatic recognition." *Automatic Speech Recognition and Understanding - ASRU.* Italy: IEEE, 2001. 437 - 440.

Rabiner, L. and Juang, B. "An introdution to Hidden Markov Models." *IEEE, V.77, No.2*, 1989: 257-286.

Ravanelli, M. and Do, Van Hai and Janin, A. "TANDEM-bottleneck feature combination using hierarchical Deep Neural Networks." *Chinese Spoken Language Processing (ISCSLP).* Singapore, 2014. 113 – 117.

Sakai, M.,Denso Corp. "Generalization of Linear Discriminant Analysis used in Segmental Unit Input HMM for Speech Recognition." *Acoustics, Speech and Signal Processing (ICASSP).* Honolulu, 2007. IV-333 - IV-336.

Saon, G. "Speaker adaptation of neural network acoustic models using i-vectors." *Automatic Speech Recognition and Understanding (ASRU).* Olomouc, 2013. 55 – 59.

Schwenk, Holger. "Continuous space language models." *Computer Speech and Language, Vol 21*, 2007: 492-518.

Sethserey, Sam and Eric, Castelli and Laurent, Besacier. "Unsupervised acoustic model adaptation for multi-origin non native." *INTERSPEECH .* Japan: IEEE, 2010.

Shen, Peng and Lu, Xugang and Hu, Xinhui and Kanda, Naoyuki and Saiko, Masahiro and Hori, Chiori. "The NICT ASR System for IWSLT 2014." *The International Workshop on Spoken Language Translation (IWSLT).* Lake Tahoe, USA, 2014.

Sinaporn, Suebvisai and Paisarn, Charoenpornsawat and et. "Thai Automatic Speech Recognition." *Acoustics, Speech, and Signal Processing (ICASSP).* Philadlnphia, USA: IEEE, 2005. 857-860.

Snack. 2004. http://www.speech.kth.se/snack/.

SPTK. 2014. http://sp-tk.sourceforge.net.

SRI, International. *SRILM - The SRI Language Modeling Toolkit.* 2011. http://www.speech.sri.com/projects/srilm/.

Stolcke, Andreas. "Entropy-based Pruning of Backoff Language Models." *DARPA Broadcast News Transcription and Understanding.* Virginia, 1998. 270-274.

Stuker, S. and Kilgour, K. and Saam, C. and Waibel, A. "The 2011 kit english asr system for the iwslt evaluation." *International Workshop on Spoken Language Translation (IWSLT).* SanFrancisco, 2011.

Suphattharachai, Chomphan. "Analysis of Decision Trees in Context Clustering of Hidden Markov Model Based Thai Speech Synthesis." *Computer Science , Vol 7, ISSN 1549-3636*, 2011: 359-365.

Takashi, Masuko and Keiichi, Tokuda and Noboru, Miyazak and Takao, Kobayashi. "Pitch pattern generation using multispace probability distribution HMM." *Systems and Computers in Japan, Vol 33, No 6*, 2002: 62-72.

Talkin, D. "A Robust Algorithm For Pitch Tracking." In *Speech coding and synthesis*, 495-518. USA: Elsevier, 1995.

Tebelskis, Joe. *Speech Recognition using Neural Networks.* USA: Carnegie Mellon University, 1995.

Tokuda, K. and Masuko, Takashi and Miyazaki, Noboru and Kobayashi, Takao. "Hidden Markov models based on multi-space probability distribution for pitch pattern modeling." *Acoustics, Speech, and Signal Processing (ICASSP).* Phoenix, USA, 1999. 229-232.

Tong, Rong and Lim, Boon Pang and Chen, N.F. and Ma, Bin and Li, Haizhou. "Subspace Gaussian mixture model for computer-assisted language learning." *Acoustics, Speech and Signal Processing (ICASSP).* Florence, 2014. 5347 – 5351.

Tuan, Nguyen and Hai Quan, Vu. "Advances in Acoustic Modeling for Vietnamese LVCSR." *Asian Language Processing.* Singapore: IEEE, 2009. 280 - 284.

Tuerxun, M. and Zhang, Shiliang and Bao, Yebo and Dai, Lirong. "Improvements on bottleneck feature for large vocabulary continuous speech recognition." *Signal Processing (ICSP).* Hangzhou, 2014. 516 – 520.

Thang, Vu Tat and Tang, Khanh Nguyen and Le, Son Hai and Luong, Mai Chi. "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.* Kyoto,, 2008. 253–256.

Thắng, Vũ Ngọc. *Automatic Speech Recognition for Low-resource Languages and Accents Using Multilingual and Crosslingual Information.* Karlsruher - Germany: Karlsruher Instituts of Technologie - KIT, 2014.

Van Huy, N. and Chi Mai, L. and Tat Thang, V. "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-China: IEEE, 2015.

Vesely, K. and Karafiat, M. and Grezl, F. "Convolutive Bottleneck Network features for LVCSR." *ASRU.* Waikoloa: IEEE, 2011. 42-47.

Vu, Ngoc Thang and Schultz, Tanja. "Vietnamese Large Vocabulary Continuous Speech Recognition." *Automatic Speech Recognition & Understanding - ASRU.* Merano: IEEE, 2009. 333 - 338.

Vu, Thang Tat and Nguyen, Dung Tien and Luong, Mai Chi and Hosom, John Paul. "Vietnamese large vocabulary continuous speech recognition." *INTERSPEECH.* Lisbon, 2005. 1172-1175.

Wang, Huanliang and et. "A Multi-Space Distribution (MSD) Approach to speech recognition of tonal languages." *INTERSPEECH.* Pittsburgh, USA: IEEE, 2006.

Womak, B.D. "Improved speech recognition via speaker stress directed classification." *Acoustics, Speech, and Signal Processing (ICASSP).* Atlanta-GA: IEEE, 1996. 53-56.

Young, Steve. *The HTK Book.* UK: Cambridge University Engineering Department, 2009.

Yu, Kai and Young, S. "Continuous F0 Modeling for HMM Based Statistical Parametric Speech Synthesis." *Audio, Speech, and Language Processing, IEEE, V. 19, Issue 5, ISSN:1558-7916* [IEEE], 2010: 1071 – 1079.

Online

Snack. 2004. http://www.speech.kth.se/snack/.

SPTK. 2014. http://sp-tk.sourceforge.net.

SRI, International. *SRILM - The SRI Language Modeling Toolkit.* 2011. http://www.speech.sri.com/projects/srilm/.

Phụ lục

1. TCL Script tạo từ điển ngữ âm cho một tập văn bản tiếng Việt đầu vào bất kỳ

1) Nội dung các file "BphonemeVN.txt", "MphonemeVN.txt", "EphonemeVN.txt" để ánh xạ các âm đầu, âm chính, âm cuối sang biểu diễn phoneme tương ứng

BphonemeVN.txt	***MphonemeVN.txt***	***EphonemeVN.txt***
ph ph	oai w a iz	ch kc
f ph	oao w a uz	ng ngz
th th	oay w aw iz	c kc
tr tr	uây w aa iz	t tc
gi d	oeo w e uz	n nc
d d	iêu ie uz	x kc
ch ch	yêu ie uz	nh ngz
nh nh	uya w ie	p pc
ng ng	uyu w i uz	m mc
ngh ng	uôi uo iz	
kh kh	ươi wa iz	
g g	ươu wa uz	
gh g	uyê w ie	
c k	ia ie	
qu k	iê ie	
k k	ua uo	
t t	uô uo	
r r	ưa wa	
h h	ươ wa	
b b	ai a iz	
m m	ay aw iz	
v v	ây aa iz	
đ dd	oi o iz	
n n	ôi oo iz	
l l	ơi ow iz	
	ui u iz	
	ưi uw iz	
	ao a uz	
	au a uz	
	âu aa uz	
	eo e uz	
	êu ee uz	
	iu i uz	
	ưu uw uz	
	oa w a	
	oă w aw	
	uâ w aa	
	oe w e	
	uê w ee	
	uơ w ow	

	uy w i yê w ie oo o ôô oo a a	

2) Script

```
# making Vietnamese phoneme Dict
# Author: Nguyen Van Huy, huynguyen@tnut.edu.vn
# 2014-Step-12

#!/bin/tclsh
encoding system utf-8
# vietnmaese triphone and tiphone
if { $argc != 1 } {
	puts "Requite an input Word-List"
	exit 0
}
#load middel phone
set lcuda "oai oao oay uây oeo iêu yêu uya uyu uôi ươi ươu uyê oài oào oày uầy oèo iều yều uỳa uỳu uồi ười ườu uyề oái oáo oáy uấy oéo iếu yếu uýa uýu uối ướI ướu uyế oải oảo oảy uẩy oẻo iểu yểu uỷa uỷu uổi ưởi ưởu uyể oãi oão oãy uẫy oẽo iễu yễu uỹa uỹu uỗi ưỡi ưỡu uyễ oại oạo oạy uậy oẹo iệu yệu uỵa uỵu uội ượi ượu uyệ ia ua ưa iê ươ ai ay ây oi ôi ơi ui ưi ao au âu eo êu iu ưu oa oa oă uâ oe uê uô uơ uy yê oo ôô ìa ùa ừa iề ườ ài ày ầy òi ồi ời ùi ừi ào àu ầu èo ều ìu ừu oà òa oằ uầ oè uề uồ uờ uỳ yề oò ôồ ía úa ứa iế ướ ái áy ấy ói ối ới úi ứi áo áu ấu éo ếu íu ứu oá óa oắ uấ oé uế uố uớ uý yế oó ôố ỉa ủa ửa iể ưở ải ảy ẩy ỏi ổi ởi ủi ửi ảo ảu ẩu ẻo ểu ỉu ửu oả ỏa oẳ uẩ oẻ uể uổ uở uỷ yể oỏ ôổ ĩa ũa ữa iễ ưỡ ãi ãy ẫy õi ỗi ỡi ũi ữi ão ãu ẫu ẽo ễu ĩu ữu oã õa oẵ uẫ oẽ uễ uỗ ưỡ uỹ yễ oõ oỗ ịa ụa ựa iệ ượ ại ạy ậy ọi ội ợi ụi ựi ạo ạu ậu ẹo ệu ịu ựu oạ ọa oặ uậ oẹ uệ uộ uợ uỵ yệ oọ oộ a ă â e ê i o ô ơ u ư y à ằ ầ è ề ì ò ồ ờ ù ừ ỳ á ắ ấ é ế í ó ố ớ ú ứ ý ả ẳ ẩ ẻ ể ỉ ỏ ổ ở ủ ử ỷ ã ẵ ẫ ẽ ễ ĩ õ ỗ ỡ ũ ữ ỹ ạ ặ ậ ẹ ệ ị ọ ộ ợ ụ ự ỵ"

# Load Begining phoneme
set Bphonemef [open "BphonemeVN.txt" r]
set LBphoneme [split [read $Bphonemef] "\n"]
close $Bphonemef

# Load Ending phoneme
set Ephonemef [open "EphonemeVN.txt" r]
set LEphoneme [split [read $Ephonemef] "\n"]
close $Ephonemef
# Load Middle phoneme
set Mphonemef [open "MphonemeVN.txt" r]
set LMphoneme [split [read $Mphonemef] "\n"]
close $Mphonemef
####
# input word list to make phoneme Dict
set inf [open [lindex $argv 0] r]
```

```
#==============================
set wlist [split [read $inf ] "\n"]
close $inf
set count [expr -1]
set wdone ""
foreach phone $lcuda {
	incr count
	#puts "working on phone: $phone"
	set outsearch [lsearch -inline -all $wlist "*$phone*"]
	if {$outsearch!=-1} {
		foreach word $outsearch {
			if {$count < 78 } {
				set Mphone [lindex $lcuda [expr $count % 13]]
				set toneP [expr $count/13]
			} elseif {$count > 77 && $count < 270} {
				set Mphone [lindex $lcuda [expr (($count-77) % 32)+77]]
				set toneP [expr  ($count-77)/32]
			} else {
				set Mphone [lindex $lcuda [expr (($count-270) % 12)+270]]
				set toneP [expr  ($count-270)/12]
			}
			switch $toneP {
				0 {set tone 1}
				1 {set tone 2}
				2 {set tone 3}
				3 {set tone 4}
				4 {set tone 5}
				5 {set tone 6}
			}
			set start [string first $phone $word]
			set end [expr $start + [string length $phone] -1]
			if {$start!=0} {
				set Bphone [string range $word 0 [expr $start-1]]
			} else {
				set Bphone ""
			}
			if {$end!=[expr [string length $word]-1]} {
				set Ephone [string range $word [expr $end+1] end]
			} else {
				set Ephone ""
			}
			if {$Bphone=="q" && [string index $Mphone 0]=="u" && [string
length $Mphone]>1} {
				set Bphone "qu"
				set Mphone [string range $Mphone 1 end]
			}
			if {$Bphone=="g" && [string index $Mphone 0]=="i" && [string
length $Mphone]>1} {
```

```
                        set Bphone "gi"
                        set Mphone [string range $Mphone 1 end]
                                } else {
                        set Bphone ""
                    }
                    if {$end!=[expr [string length $word]-1]} {
                        set Ephone [string range $word [expr $end+1] end]
                    } else {
                        set Ephone ""
                    }
                    if {$Bphone=="q" && [string index $Mphone 0]=="u" && [string
length $Mphone]>1} {
                        set Bphone "qu"
                        set Mphone [string range $Mphone 1 end]
                    }
                    if {$Bphone=="g" && [string index $Mphone 0]=="i" && [string
length $Mphone]>1} {
                        set Bphone "gi"
                        set Mphone [string range $Mphone 1 end]
                    }
                    # convert phone to phoneme
                    set Bphoneme [lindex [lsearch -inline $LBphoneme "${Bphone} *"] 1]
                    #set Bphoneme [lsearch -inline $LBphoneme "${Bphone} *"]
                    if {$Bphoneme==-1} {set Bphoneme $Bphone}
                    set Ephoneme [lindex [lsearch -inline $LEphoneme "${Ephone} *"] 1]
                    #set Ephoneme [lsearch -inline $LEphoneme "${Ephone} *"]
                    set Mphoneme [lsearch -inline $LMphoneme "${Mphone} *"]
                    if {[llength $Mphoneme]>2} {
                        set tmpstr ""
                        set Mphoneme [lrange $Mphoneme 1 end]
                        foreach ph $Mphoneme {
                            set tmpstr "$tmpstr ${ph}${tone}"
                        }
                    } else {
                        set tmpstr "[lindex $Mphoneme 1]${tone}"
                    }
                    set tmpstr [string trim $tmpstr]
                    set tmpstr [string trim "$Bphoneme $tmpstr $Ephoneme"]
                    lappend tmpdict "$word  $tmpstr"
                    set wlist [lsearch -inline -all -not -exact $wlist $word]
                }
            }

}
set outdict [lsort $tmpdict]
foreach tmp $outdict {puts $tmp}
set errf [open OVV.err w]
puts $errf $wlist
```

```
close $errf
```

2. File cấu hình mô hình MSD-HMM

```
~o <VecSize> 16 <USER><DIAGC><MSDInfo> 2  0  1 <StreamInfo> 2 13 3
<BeginHMM>
<NumStates> 5
<State> 2
<SWeights> 2 1.0 1.0
<Stream> 1
<Mean> 13
    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
<Variance> 13
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0
<Stream> 2
<NumMixes> 4
<Mixture> 1 0.25000
<Mean> 3
    0.0 0.0 0.0
<Variance> 3
    1.0 1.0 1.0
<Mixture> 2 0.25000
<Mean> 3
    0.0 0.0 0.0
<Variance> 3
    1.0 1.0 1.0
<Mixture> 3 0.25000
<Mean> 3
    0.0 0.0 0.0
<Variance> 3
    1.0 1.0 1.0
<Mixture> 4 0.25000
<Mean> 0
<Variance> 0
<State> 3
<SWeights> 2 1.0 1.0
<Stream> 1
<Mean> 13
    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
<Variance> 13
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0
<Stream> 2
<NumMixes> 4
<Mixture> 1 0.25000
<Mean> 3
```

0.0 0.0 0.0
<Variance> 3
1.0 1.0 1.0
<Mixture> 2 0.25000
<Mean> 3
0.0 0.0 0.0
<Variance> 3
1.0 1.0 1.0
<Mixture> 3 0.25000
<Mean> 3
0.0 0.0 0.0
<Variance> 3
1.0 1.0 1.0
<Mixture> 4 0.25000
<Mean> 0
<Variance> 0
<State> 4
<SWeights> 2 1.0 1.0
<Stream> 1
<Mean> 13
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
<Variance> 13
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0
<Stream> 2
<NumMixes> 4
<Mixture> 1 0.25000
<Mean> 3
0.0 0.0 0.0
<Variance> 3
1.0 1.0 1.0
<Mixture> 2 0.25000
<Mean> 3
0.0 0.0 0.0
<Variance> 3
1.0 1.0 1.0
<Mixture> 3 0.25000
<Mean> 3
0.0 0.0 0.0
<Variance> 3
1.0 1.0 1.0
<Mixture> 4 0.25000
<Mean> 0
<Variance> 0
1.0 1.0 1.0
<Mixture> 3 0.25000
<Mean> 3

```
    0.0 0.0 0.0
<Variance> 3
    1.0 1.0 1.0
<Mixture> 4 0.25000
<Mean> 0
<Variance> 0
<State> 4
<SWeights> 2 1.0 1.0
<Stream> 1
<Mean> 13
    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
<Variance> 13
    1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
    1.0 1.0 1.0
<Stream> 2
<NumMixes> 4
<Mixture> 1 0.25000
<Mean> 3
    0.0 0.0 0.0
<Variance> 3
    1.0 1.0 1.0
<Mixture> 2 0.25000
<Mean> 3
    0.0 0.0 0.0
<Variance> 3
    1.0 1.0 1.0
<Mixture> 3 0.25000
<Mean> 3
    0.0 0.0 0.0
<Variance> 3
    1.0 1.0 1.0
<Mixture> 4 0.25000
<Mean> 0
<Variance> 0
<TransP> 5
     0.0 1.0 0.0 0.0 0.0
     0.0 0.6 0.4 0.0 0.0
     0.0 0.0 0.6 0.4 0.0
     0.0 0.0 0.0 0.7 0.3
     0.0 0.0 0.0 0.0 0.0
<EndHMM>
```

===Hết===