

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Chu Nguyên Đức - Bùi Chí Dũng

**NGHIÊN CỨU VÀ PHÁT TRIỂN HỆ
THÔNG XÁC NHẬN NGƯỜI NÓI TRÊN
THIẾT BỊ NHÚNG**

KHÓA LUẬN TỐT NGHIỆP CỦ NHÂN
CHƯƠNG TRÌNH CHÍNH QUY

Tp. Hồ Chí Minh, tháng 07/2021

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Chu Nguyên Đức - 1712352

Bùi Chí Dũng - 1712364

NGHIÊN CỨU VÀ PHÁT TRIỂN HỆ
THÔNG XÁC NHẬN NGƯỜI NÓI TRÊN
THIẾT BỊ NHÚNG

KHÓA LUẬN TỐT NGHIỆP CỦA NHÂN
CHƯƠNG TRÌNH CHÍNH QUY

GIÁO VIÊN HƯỚNG DẪN

TS. Nguyễn Đức Hoàng Hà

Tp. Hồ Chí Minh, tháng 07/2021

Lời cảm ơn

Khóa luận tốt nghiệp là nhiệm vụ cuối cùng trước khi sinh viên rời khỏi giảng đường đại học. Để trở thành một cử nhân đóng góp những kiến thức, kinh nghiệm mà giảng viên và nhà trường đã truyền đạt, cho sự phát triển của đất nước.

Trong suốt quá trình thực hiện đề tài này, nhóm đã nhận được sự giúp đỡ từ quý thầy cô và bạn bè, sự động viên từ gia đình, người thân. Nhờ đó mà nhóm đã đạt được kết quả là hoàn thành được luận văn như mong muốn. Vì vậy, nhóm chúng em xin gửi lời cảm ơn chân thành đến:

Các quý thầy cô công tác tại khoa Công nghệ thông tin, trường Đại học Khoa học tự nhiên, đã truyền đạt những kiến thức quý báu cho chúng em. Quý thầy cô công tác tại các phòng ban của trường Đại học Khoa học tự nhiên đã tạo điều kiện thuận lợi, giúp đỡ chúng em trong quá trình học tập tại trường và hoàn thành luận văn này.

Đặc biệt, nhóm chúng em xin gửi lời cảm ơn sâu sắc đến TS. Nguyễn Đức Hoàng Hạ, người đã hướng dẫn trực tiếp chúng em hoàn thành khóa luận này. Trong suốt quá trình thực hiện, thầy đã dành thời gian, công sức cho nhóm, tạo điều kiện tốt nhất để chúng em hoàn thành tốt khóa luận và có thể thực hiện đúng hướng và đúng kế hoạch đề ra.

Chúng em xin gửi lời cảm ơn chân thành đến quý thầy cô trong hội đồng chấm luận văn đã dành thời gian cho nhóm và đưa ra những góp ý, nhận xét chân thành. Những điều này sẽ vô cùng quý báu, giúp chúng em có thể cải thiện bản thân, cũng như có thể giúp luận văn có thể hoàn thiện hơn nữa.

Cuối cùng, nhóm xin gửi lời cảm ơn đến gia đình, người thân và bạn bè. Những người đã luôn động viên, ủng hộ, giúp đỡ chúng em bất cứ khi nào chúng em cần.

Đề cương chi tiết



fit@hcmus

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

ĐỀ CƯƠNG KHOÁ LUẬN TỐT NGHIỆP

NGHIÊN CỨU VÀ PHÁT TRIỂN HỆ THÔNG XÁC NHẬN NGƯỜI NÓI TRÊN THIẾT BỊ NHÚNG

(SPEAKER VERIFICATION ON EMBEDDED SYSTEM)

Thông tin chung

Người hướng dẫn:

- TS. Nguyễn Đức Hoàng Hạ (Khoa Công nghệ Thông tin)

Nhóm sinh viên thực hiện:

1. Chu Nguyên Đức (MSSV: 1712352)
2. Bùi Chí Dũng (MSSV: 1712364)

Loại đề tài: Ứng dụng

Thời gian thực hiện: Từ 01/2021 đến 06/2021

Nội dung thực hiện

Giới thiệu về đề tài

Nhận diện người nói là quá trình nhận diện người nói bằng cách căn cứ vào đặc trưng âm thanh của giọng nói người để xác minh danh tính của người truy cập hệ thống. Nó cho phép truy cập các dịch vụ cũng như hệ thống khác nhau bằng giọng nói. Một số ứng dụng có thể kể đến như giao dịch ngân hàng qua điện thoại, mua sắm trực tuyến, truy cập cơ sở dữ liệu, đặt chỗ tự động, hộp thư thoại, hệ thống an ninh hay thậm chí là công cụ truy tìm tội phạm, ...

Nhận diện người nói có thể phân thành hai loại đó là nhận diện và xác minh. Nhận diện người nói là quá trình xác định từ người nói đã đăng ký một nội dung giọng nói nhất định xuất phát từ đâu. Xác minh người nói là quá trình chấp nhận hoặc từ chối danh tính mà người nói đã xác nhận. Trong hầu hết các ứng dụng, giọng nói được sử dụng để xác nhận danh tính của người nói. Sự khác biệt cơ bản giữa xác định và xác minh người nói đó là số lượng lựa chọn. Nếu trong xác định người nói, số lượng lựa chọn bằng với số lượng người nói, thì trong xác minh chỉ có hai lựa chọn là chấp nhận hoặc từ chối, bất kể có bao nhiêu người nói. Vì vậy hiệu suất nhận diện người nói giảm khi số lượng người nói tăng, trong khi hiệu xuất xác minh gần như độc lập với số lượng người nói.

Đối với bất cứ phương pháp xác thực nào, người sử dụng có thể lưu trữ dữ liệu trực tiếp trên đám mây, sử dụng kết nối internet và xử lý các tín hiệu âm thanh thu được để xác thực giọng nói một cách trực tuyến. Một số ứng dụng có thể kể đến như các trợ lý ảo Siri hay Google Assistant. Tuy nhiên cách thực hiện này sẽ không khả thi đối với nhu cầu sử dụng hệ thống các thiết bị đòi hỏi tính độc lập với internet như hệ thống cửa ra vào, hệ thống điều khiển các thiết bị thông minh trong nhà hay thậm chí là thiết bị hỗ trợ dịch thuật. Các ứng dụng này cần hệ thống nhận diện giọng nói luôn hoạt động, kể cả khi không có kết nối internet hay thậm chí là mất điện. Một mô hình nhận diện sẽ được huấn luyện và cài đặt trực tiếp vào thiết bị nhúng, nhỏ gọn và tiện lợi. Đây là những thiết bị dễ dàng kết nối hay điều khiển với các thiết bị trong nhà bằng các loại sóng vô tuyến RF hay sóng hồng ngoại IR. Đây là những loại sóng gần gũi dễ sử dụng và cực kỳ tiện lợi, điều khiển trực tiếp không cần có kết nối

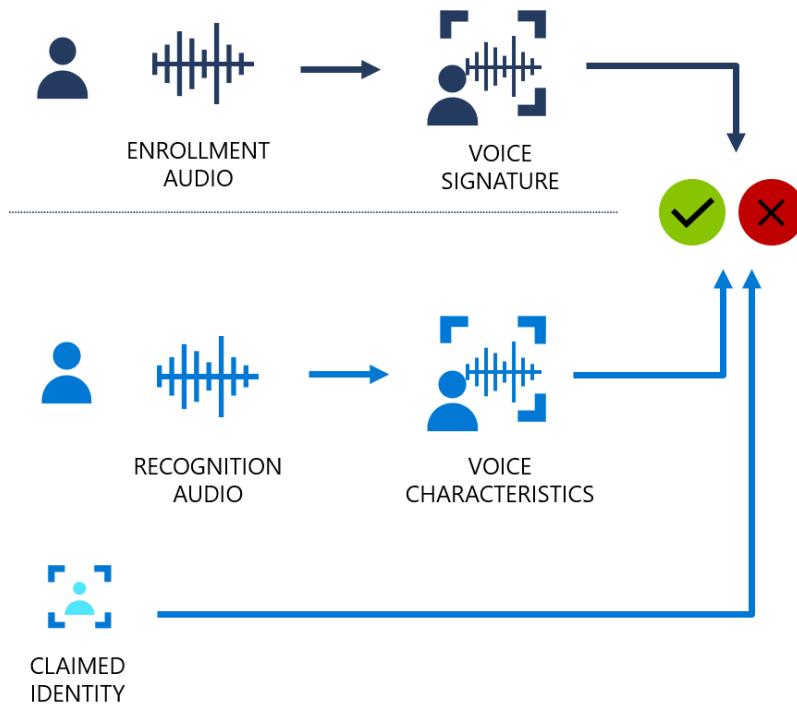
internet.

Từ những thực tiễn đó, nhóm sẽ xây dựng một thiết bị nhận diện người nói hỗ trợ việc xác minh danh tính người nói từ nguồn âm thanh đầu vào và so khớp với dữ liệu người nói đã đăng ký trong cơ sở dữ liệu để đưa ra kết quả chấp nhận (khi trùng khớp) hay từ chối (khi không trùng khớp). Dựa trên phương pháp xác minh người nói trên hình thức độc lập văn bản.

Mục tiêu đề tài

Mục tiêu chính của đề tài là nghiên cứu và phát triển một mô hình xác nhận người nói, sau đó tìm ra mô hình tốt nhất bằng cách kiểm thử trên thiết bị máy tính và triển khai trên thiết bị nhúng, phục vụ hướng đi xa hơn cho lĩnh vực này trong tương lai đó là dùng thiết bị nhúng này để kết nối vào các hệ thống ứng dụng thực tiễn nhận diện giọng nói như đã mô tả ở phần giới thiệu đề tài.

Hệ thống xác minh nói là hệ thống có thể xác minh danh tính của một người dựa vào giọng nói của người đó, so với giọng nói đã được đăng ký và lưu trữ. Phương pháp nhận diện người nói chia thành hai phương pháp chính đó là độc lập với văn bản (Text-Independent) và phụ thuộc với văn bản (Text-Dependent). Phương pháp độc lập văn bản, người nói được yêu cầu cung cấp nội dung văn bản, chúng được sử dụng cho cả đào tạo hay nhận diện, và chúng không dựa vào văn bản đang được nói. Đối với phương pháp phụ thuộc văn bản, thường căn cứ vào kỹ thuật đối sánh mẫu, căn cứ vào trực thời gian của giọng nói đầu vào và sự tương đồng giữa hai giọng nói đến từng âm vị, âm tiết từ đầu đến cuối lời nói. Phương pháp này mang hiệu suất nhận diện cao hơn so với phương pháp độc lập văn bản. Hệ thống xác minh người nói bao gồm 2 bước: đăng ký (enroll) giọng nói và danh tính người nói; xác minh (verify) so sánh giữa người nói hiện tại và dữ liệu đã có trong hệ thống để đưa ra kết luận chấp nhận hay từ chối. Các bước được thể hiện cụ thể như hình:



Hình 1: Các bước của hệ thống xác nhận người nói[10]

Hiện nay có khá nhiều phương pháp để giải quyết bài toán xác nhận người nói này, trong đó có thể kể đến những phương pháp sử dụng Deep Learning[15] như: x-vector[14], d-vector[16], j-vector[2], ... Mục tiêu của nhóm là áp dụng một trong những phương pháp Deep Learning này, sau đó có thể fine-tuning mô hình (tận dụng một phần các layer, thêm sửa xóa các tham số để tạo ra mô hình mới) để mang lại hiệu quả hơn cho việc xác nhận người nói.

Sau khi chọn được mô hình thích hợp thì nhóm sẽ triển khai trên thiết bị nhúng, có thể là Raspberry Pi hoặc ESP32. Raspberry Pi là chiếc máy tính kích thước nhỏ được tích hợp nhiều phần cứng mạnh mẽ đủ khả năng chạy hệ điều hành và cài đặt được nhiều ứng dụng trên nó. Với giá khoảng vài chục USD, Raspberry hiện đang là mini computer nổi bật nhất hiện nay.

Nếu triển khai được trên thiết bị Raspberry Pi và vẫn còn đủ thời gian thì nhóm sẽ tiếp tục đưa mô hình vào thiết bị ESP32. ESP32 là module MCU đa dụng, mạnh mẽ và được sử dụng rộng rãi trong nhiều ứng dụng về IoTs, với giá thành rẻ hơn cả Raspberry Pi. Tuy nhiên việc áp dụng trên ESP32 sẽ khó khăn vì nó không mạnh bằng Raspberry Pi, nên nhóm sẽ áp dụng khi có đủ điều kiện.



Hình 2: Raspberry Pi



Hình 3: ESP32

Phạm vi của đề tài

Trong khuôn khổ việc thực hiện luận văn này, công việc chính của nhóm sẽ tìm xây dựng một pre-trained model nhận diện người nói. Pre-trained model là mô hình được huấn luyện trước đó với một bộ dữ liệu lớn hoặc với các phương pháp tối tân giúp giảm công sức huấn luyện mô hình từ đầu. Mô hình sau đó có thể huấn luyện thêm hoặc tinh chỉnh để phù hợp với bộ dữ liệu thực tế hoặc sử dụng trực tiếp cho các bài toán học máy. Ở đây mô hình được dùng sẽ được thay đổi tham số cũng như bộ dữ liệu huấn luyện để phù hợp hơn, có thể triển khai trên thiết bị nhúng.

Cài đặt mô hình này trên thiết bị nhúng như Raspberry PI đi kèm với một số linh kiện phục vụ cho việc kiểm tra chức năng, độ chính xác của hệ thống.

Hệ thống này sẽ cho phép nhận đầu vào là một nguồn âm thanh giọng nói. Sau khi xử lý và so khớp với âm thanh giọng nói đã đăng ký trong cơ sở dữ liệu theo hình thức TI-SV, sẽ cho ra kết quả trùng khớp hoặc không. Đối với hình thức TD-SV sẽ là một đề tài mở rộng của nhóm, sẽ nghiên cứu trong thời gian sắp tới. Cùng với đó là việc áp dụng triển khai mô hình trên ESP32, vì thiết bị này nhỏ gọn hơn cả Raspberry PI và có phần phức tạp hơn cùng với một số lý do đã nêu ở phần mục tiêu.

Cách tiếp cận dự kiến

Cách tiếp cận cụ thể của nhóm bao gồm 3 bước:

- Tìm hiểu mô hình speaker verification: Sau quá trình tìm hiểu và thực hành thì nhóm chọn phương pháp sử dụng x-vector[14] để giải bài toán này. Nhóm sử dụng công cụ Kaldi[6] để thực hiện (Kaldi là một bộ công cụ mã nguồn mở được tạo ra để xử lý dữ liệu giọng nói, được viết chủ yếu bằng C/C++ và được đóng gói bằng các tập lệnh Bash và Python), cùng với pretrained model[7] đã được huấn luyện trên tập dữ liệu VoxCeleb[11]. Nhóm đã tiến hành kiểm thử mô hình trên máy và đạt được độ lỗi tốt (3.128%).
- Triển khai lên Raspberry Pi: Nhóm sẽ chuẩn bị cài đặt những phần cứng và phần mềm cần thiết cho Raspberry Pi, sau đó đưa mô hình đã được huấn luyện trên máy tính lên Raspberry Pi. Trong trường hợp mô hình quá lớn so với Raspberry Pi thì nhóm sẽ huấn luyện lại để ra được mô hình nhỏ hơn.
- Thực hiện phần demo speaker verification trên giọng nói thực tế: Sau khi hoàn tất kiểm thử mô hình, nhóm sẽ áp dụng mô hình vào thực tế. Nhóm sẽ chuẩn bị microphone để thu âm giọng nói cho quá trình enroll và verify. Khi mô hình dự đoán xong thì sẽ xuất tín hiệu ra đèn LED để hiển thị là có đúng người nói không.

Kết quả dự kiến của đề tài

- Luận văn, bao gồm báo cáo chi tiết và đầy đủ về quá trình hoạt động của nhóm để đạt được kết quả này, bảng đánh giá, so sánh kết quả đối với các mô hình tương tự.
- Dữ liệu đã thu thập được phục vụ cho quá trình thực hiện đề tài.
- Demo chi tiết dưới cùng hướng dẫn sử dụng thiết bị.
- Thiết bị nhận diện người nói đã được cài đặt mô hình nhận diện cùng các chức năng đăa nêu, có thẻ đăng ký và xác thực dữ liệu giọng nói đầu vào.

Kế hoạch thực hiện

STT	Thời gian	Công việc	Người thực hiện
1	02/01/2021 - 15/02/2021	Bổ sung tìm hiểu kiến thức nền tảng nhận diện người nói, tìm hiểu model thực hiện, tìm kiếm các đề tài nghiên cứu tương tự, chạy thử các mô hình nhận diện người nói.	Cả nhóm
2	15/02/2021 - 01/03/2021	Xem lại kiến thức nhận diện giọng nói, lựa chọn mô hình thích hợp để tìm hiểu và chuẩn bị cho bước tiền xử lý.	Cả nhóm
3	03/03/2021 - 08/03/2021	Viết đề cương	Cả nhóm
4	06/03/2021 - 20/03/2021	Thu thập dữ liệu, tìm hiểu tham số mô hình đã chọn	Dũng
5	06/03/2021 - 20/03/2021	Tìm hiểu tham số mô hình đã chọn, tìm hiểu cách triển khai mô hình trên thiết bị nhúng	Đức
6	20/03/2021 - 10/04/2021	Huấn luyện mô hình trên nền tảng máy tính, ghi nhận tham số tốt, đánh giá và ghi nhận kết quả thực hiện để đối chiếu các phiên bản của mô hình	Cả nhóm
7	10/04/2021 - 24/04/2021	Chạy thử và sửa lỗi, hoàn thiện mô hình, tối ưu kết quả	Dũng
8	10/04/2021 - 08/05/2021	Triển khai mô hình trên thiết bị nhúng	Đức
9	24/04/2021 - 08/05/2021	Triển khai mô hình trên thiết bị nhúng	Dũng
10	08/05/2021 - 15/05/2021	Hoàn thiện triển khai mô hình trên thiết bị nhúng, đánh giá và ghi nhận kết quả vào báo cáo	Cả nhóm
11	15/05/2021 - 29/05/2021	Tổng hợp báo cáo và hoàn chỉnh báo cáo	Cả nhóm
12	29/05/2021 - 12/06/2021	Backup time	Cả nhóm

Mục lục

Lời cảm ơn	i
Đề cương chi tiết	ii
Mục lục	xi
Danh sách hình vẽ	xiii
Danh sách bảng	xv
Tóm tắt	xv
1 Giới thiệu	1
1.1 Nhận diện người nói và một số ứng dụng	1
1.2 Lý do chọn đề tài	2
1.3 Mục tiêu luận văn	4
1.4 Bố cục	4
2 Cơ sở lý thuyết nhận diện người nói	5
2.1 Nguyên lý hoạt động hệ thống nhận diện giọng nói	5
2.2 Phương pháp nhận diện người nói độc lập văn bản (Text - Independent) và phụ thuộc văn bản (Text - Dependent)	7
2.3 Phương pháp rút trích đặc trưng	7
2.3.1 Phát hiện hoạt động người nói	8
2.3.2 Đặc trưng MFCC	10
2.3.3 GMM-UBM	18
2.3.4 i-vector	20

2.3.5	Deep Learning	21
3	Tổng quan về Raspberry Pi	25
3.1	Giới thiệu	25
3.2	Cấu trúc phần cứng	26
3.3	Hệ điều hành cho Raspberry Pi	30
3.4	Ứng dụng	31
4	Mô hình nhận diện người nói Kaldi	34
4.1	Giới thiệu	34
4.2	Tổng quan về Kaldi	36
4.3	Cấu trúc thư mục trong Kaldi	39
5	Hệ thống nhận diện người nói trên thiết bị nhúng Raspberry Pi	42
5.1	Chuẩn bị	42
5.1.1	Phần cứng	42
5.1.2	Phần mềm	43
5.2	Cài đặt	44
5.2.1	Cài đặt Kaldi trên Raspberry Pi	44
5.2.2	Cấu hình hệ thống xác nhận người nói	45
5.2.3	Chọn ngưỡng PLDA	51
5.3	Kết quả	52
5.3.1	Chuẩn bị dữ liệu	52
5.3.2	Tiến hành và kết quả	52
5.4	Kết luận và hướng phát triển	55
5.4.1	Kết luận	55
5.4.2	Hướng phát triển	56
	Tài liệu tham khảo	57

Danh sách hình vẽ

1	Các bước của hệ thống xác nhận người nói[10]	vi
2	Raspberry Pi	vii
3	ESP32	vii
2.1	Sơ đồ tổng quát của hệ thống xác nhận người nói	6
2.2	Biểu đồ thể hiện tín hiệu đầu vào,năng lượng và ước lượng giọng nói	9
2.3	Sơ đồ quá trình trích chọn đặc trưng MFCC	10
2.4	Nâng mức năng lượng của âm (hình bên trái là trước khi nâng và bên phải là sau khi nâng)	11
2.5	Quá trình Windowing	12
2.6	Spectrogram	14
2.7	Trước (trái) và sau (phải) khi biến đổi DFT	14
2.8	Các bộ lọc Mel	16
2.9	Hệ thống GMM-UBM	20
3.1	Raspberry Pi	26
3.2	Bảng so sánh thông số kỹ thuật các phiên bản Raspberry Pi .	28
3.3	Raspberry Pi 3 Model B+	29
3.4	Raspberry Pi ứng dụng trên xe điều khiển từ xa	32
3.5	Raspberry Pi ứng dụng trên máy pha cà phê	33
4.1	Kiến trúc bộ công cụ Kaldi	37
4.2	Cấu trúc thư mục egs - Kaldi	40
5.1	Cấu trúc thư mục	45
5.2	Cấu trúc thư mục db	46

5.3	Giao diện phiên bản command line	47
5.4	Giao diện phiên bản GUI	48
5.5	Chức năng Enrollment	48
5.6	Chức năng Delete Data	48
5.7	Chức năng Verification	49
5.8	Chức năng Configuration	49
5.9	Chức năng Format Data	50
5.10	Hình ảnh hệ thống trong thực tế. (1): USB Microphone; (2): Raspberry Pi 3; (3): Màn hình LCD; (4): Adapter 5V 2A . . .	50
5.11	Kết quả chọn ngưỡng	51
5.12	Biểu đồ thể hiện kết quả kiểm thử trên cùng người nói	53
5.13	Biểu đồ thể hiện kết quả kiểm thử khác người nói	54
5.14	Biểu đồ thể hiện kết quả đánh giá hệ thống	55

Danh sách bảng

2.1	Bảng thể hiện chi tiết các lớp của TDNN	23
5.1	Bảng kết quả kiểm thử trên cùng người nói	52
5.2	Bảng kết quả kiểm thử khác người nói	53
5.3	Bảng kết quả đánh giá hệ thống	54

Tóm tắt

Đề tài xây dựng một hệ thống nhận diện người nói theo cách thức xác minh người nói độc lập văn bản, chấp nhận hoặc từ chối danh tính của người nói đã cung cấp căn cứ trên thông tin người nói đã đăng ký trước đó. Sử dụng Raspberry Pi 3 như một đơn vị xử lý được cài đặt công cụ nhận diện người nói - Kaldi, tuy đã lâu đời nhưng Kaldi vẫn liên tục được cập nhật và phát triển bởi một cộng đồng khá lớn và được sử dụng rộng rãi. Hệ thống được phát triển nhằm ứng dụng công nghệ sinh trắc học để giải quyết vấn đề về bảo mật, hoàn toàn độc lập với internet (offline) tạo điều kiện sử dụng hệ thống này trong nhiều ứng dụng bảo mật khác nhau. Đề tài tạo tiền đề cho vấn đề bảo mật trong hệ thống bảo mật cửa ra vào, truy cập điện thoại, truy cập hệ thống dữ liệu cá nhân, kiểm soát an ninh, ... Phương pháp này cho thấy khả năng chạy mô hình Kaldi trên một thiết bị nhỏ như Raspberry Pi 3 nhưng vẫn giữ được độ chính xác cao. Bằng việc thu thập dữ liệu giọng nói từ các nguồn khác nhau với tổng cộng 150 giọng nói được thực hiện đánh giá. Cùng với đó, nhóm đã phân bố các giọng nói thu được vào các nhóm điều kiện nhất định về môi trường âm thanh. Sau khi tiến hành kiểm thử, hệ thống đã cho kết quả tốt với các tỷ lệ FAR đạt mức tốt tại 2% và FRR đạt mức vừa phải là 17.33%. Đề tài đóng góp một phần vào lĩnh vực nghiên cứu hệ thống nhận dạng được triển khai trên thiết bị nhúng. Đây sẽ là một trong những đề tài ngày càng được quan tâm nhiều hơn ở trong và ngoài nước.

Chương 1

Giới thiệu

Nhận diện người nói là một trong những lĩnh vực đang rất được quan tâm. Từ những ứng dụng mà nhận diện người nói mang lại, cùng với đó là những khó khăn, hạn chế khi nghiên cứu về lĩnh vực tại Việt Nam nói riêng. Nhóm sẽ đưa ra lý do, giải pháp, cũng như mục tiêu để thực hiện đề tài này.

1.1 Nhận diện người nói và một số ứng dụng

Nhận diện người nói là quá trình xác minh danh tính dựa trên thông tin đặc trưng giọng nói , có thể được ứng dụng trong việc kiểm soát quyền truy cập đối với các ứng dụng hay dịch vụ có sử dụng tiếng nói người. Về mặt ứng dụng, nhận diện người nói được phân thành hai loại đó là speaker identification (nhận diện người nói) và speaker verification (xác thực người nói). Trong khi nhận diện người nói là quá trình xác định nguồn phát hay người nói của một phát âm nhất định, thì xác minh người nói là quá trình chấp nhận hoặc từ chối danh tính mà người nói đã xác nhận.

Nếu phân tích về lợi ích của nhận diện người nói. Tốc độ là điều đầu tiên chúng ta cần nói tới. Thay vì phải xử lý các tác vụ như viết email, soạn văn bản, đặt lịch hẹn, ghi chú nội dung cuộc họp,... trên giấy, điện thoại hay máy tính. Các thao tác này vừa gây tốn thời gian và thậm chí còn dễ sai sót. Người nói trong tình huống này sẽ là một phương tiện nhập liệu, không chỉ nhanh

chóng, đơn giản, thuận tiện mà còn mang đến độ chính xác cao.

Cụ thể trong lĩnh vực chuyển đổi tín hiệu, hệ thống nhận diện một câu nói hoàn chỉnh từ người nói mà không cần phải ghi chú lại bài phỏng vấn của mình. Hay thậm chí là trong các buổi liên lạc, họp báo từ xa, văn bản cuộc họp sẽ được tự động lưu lại mà không cần đến thư ký. Đây là hệ thống nhận diện tiếng nói tự động chuyển đổi lời nói thành văn bản.

Nhận diện người nói còn được sử dụng trong hệ thống điều khiển IoT như Smarthome, giúp điều khiển mọi thiết bị trong nhà một cách đơn giản chỉ bằng giọng nói. Chúng ta hoàn toàn có thể ra lệnh điều khiển các thiết bị chiếu sáng, rèm cửa, điều hòa, âm thanh, truyền hình,... được lắp các thiết bị điều khiển điện tử để có thể kết nối với internet và điện thoại di động, cho phép chủ nhà có thể điều khiển bằng người nói của chính mình.

Trong lĩnh vực nhận diện nói riêng, nhận diện người nói cũng là công nghệ được sử dụng để xác minh danh tính khách hàng, đảm bảo người nói là chủ sở hữu tài khoản ngân hàng, mà không cần dùng phương pháp xác minh thông thường bằng mã PIN hay câu hỏi bảo mật. Ngoài ra, đây còn là phương pháp bảo mật khi truy cập thông tin đặc quyền, chuyển tiền, ủy quyền thẻ tín dụng, ngân hàng thoại hay các giao dịch tương tự.

1.2 Lý do chọn đề tài

Ngày nay, dưới sự phát triển nhanh chóng của khoa học công nghệ, ngành công nghệ thông tin đã và đang đón nhận sự quan tâm của giới khoa học, đặc biệt đối với lĩnh vực khoa học máy tính. Chúng ta đã đạt đường nhiều thành tựu trong nhiều hướng nghiên cứu khác nhau, trong điều khiển thông minh, tương tác người và máy, ... Để mang đến một sự tiện lợi và sử dụng công nghệ sinh trắc thể hiện sức mạnh điều khiển của con người, lĩnh vực nghiên cứu về giọng nói người đang được đặc biệt quan tâm. Tất cả các nhu cầu giao tiếp giữa người và máy hoàn toàn có thể được thực hiện bằng lời nói, và dần trở nên cần thiết, thậm chí cả trong việc bảo mật xác minh danh tính bằng giọng nói, đây là phương thức giao tiếp hiện tại và dễ dàng nhất.

Việc ứng dụng hệ thống nhận diện người nói cho đến nay vẫn còn nhiều bất

cập. Phần lớn xuất phát từ sự biến động của giọng nói trong phát âm. Tiếng nói sẽ thay đổi theo cả thời gian và độ tuổi. Tiếng nói của người khỏe mạnh sẽ phát âm khác hẳn với những người khi gặp phải tình trạng ốm, như cảm cúm, khàn tiếng, tắt tiếng, ... Không chỉ vậy, chúng ta còn phải quan tâm cả đến tốc độ nói, cường độ âm thanh, và với mỗi người nói, trong khoảng thời gian ngắn, việt phát âm một từ sẽ có thể khác nhau. Ngoài ra, giọng nói còn chịu ảnh hưởng của yếu tố ngoại cảnh. Tiếng ồn của âm thanh ngoại cảnh, độ nhiễu của môi trường xung quanh sẽ ảnh hưởng đến chất lượng âm thanh mà chúng ta phát ra, tạo ra. Một điều kiện lý tưởng cho việc thực hiện nhận diện tiếng nói là cần thiết cho cả quá trình huấn luyện hoặc nhận diện. Tiếng nói người là một trong những đặc điểm sinh trắc đặc biệt và là duy nhất của con người, khó có thể trùng lặp với người khác. Xét về góc độ bảo mật, các hệ thống xác minh danh tính có thể sẽ được làm giả, gian lận, ... Chính vì vậy mà đây là một đề tài không hề dễ dàng, cần được quan tâm và đầu tư hơn nữa.

Đối với bất cứ phương pháp xác thực nào, người sử dụng có thể lưu trữ dữ liệu trực tiếp trên đám mây, sử dụng kết nối internet và xử lý các tín hiệu âm thanh thu được để xác thực người nói một cách trực tuyến. Một số ứng dụng có thể kể đến như các trợ lý ảo Siri hay Google Assistant. Tuy nhiên cách thực hiện này sẽ không khả thi đối với nhu cầu sử dụng hệ thống các thiết bị đòi hỏi tính độc lập với internet như hệ thống cửa ra vào, hệ thống điều khiển các thiết bị thông minh trong nhà hay thậm chí là thiết bị hỗ trợ dịch thuật. Các ứng dụng này cần hệ thống nhận diện người nói luôn hoạt động, kể cả khi không có kết nối internet hay thậm chí là mất điện. Một mô hình nhận diện sẽ được huấn luyện và cài đặt trực tiếp vào thiết bị nhúng, nhỏ gọn và tiện lợi. Đây là những thiết bị dễ dàng kết nối hay điều khiển với các thiết bị trong nhà bằng các loại sóng vô tuyến RF hay sóng hồng ngoại IR. Đây là những loại sóng gần gũi dễ sử dụng và cực kỳ tiện lợi, điều khiển trực tiếp không cần có kết nối internet.

Từ những thực tiễn đó, nhóm sẽ thực hiện đề tài "Nghiên cứu và phát triển hệ thống xác nhận người nói trên thiết bị nhúng", nhằm xây dựng một thiết bị nhận diện người nói hỗ trợ việc xác minh danh tính người nói từ nguồn âm thanh đầu vào và so khớp với dữ liệu người nói đã đăng ký trong cơ sở dữ liệu để đưa ra kết quả chấp nhận (khi trùng khớp) hay từ chối (khi không trùng

khớp). Dựa trên phương pháp xác nhận người nói độc lập với văn bản.

1.3 Mục tiêu luận văn

Sử dụng mô hình nhận diện người nói Kaldi, được tinh chỉnh tham số mô hình phù hợp với tình hình thực tiễn của hệ thống, cài đặt lên thiết bị nhúng Raspberry Pi 3 được kết nối với các linh kiện đi kèm như LCD và LED, phục vụ cho công việc kiểm tra chức năng, độ chính xác của hệ thống và hiển thị kết quả nhận diện. Bên cạnh đó, việc thực hiện luận văn sẽ giúp các thành viên hiểu rõ về cơ sở lý thuyết lĩnh vực nhận diện nói chung, và nhận diện người nói riêng. Từ những kiến thức đó, kết hợp với kiến thức về việc sử dụng thiết bị nhúng như Raspberry Pi, có thể ứng dụng cài đặt mô hình người nói trên thiết bị để tạo ra một hệ thống nhận diện người nói. Cuối cùng, có thể đánh giá khách quan khả năng tìm hiểu, nghiên cứu và thực hiện luận văn của các thành viên trong nhóm.

1.4 Bố cục

Bố cục báo cáo gồm có năm phần chính. Phần giới thiệu sẽ thông qua một số ứng dụng cùng với thông tin về nhận diện người nói, để từ đó đưa ra được lý do chọn đề tài và mục tiêu luận văn của nhóm. Tiếp theo là cơ sở lý thuyết nhận diện người nói. Các kiến thức cơ sở về nhận diện người nói được đề cập, những phương pháp, kỹ thuật sẽ được vận dụng trong luận văn cũng được đề cập trong chương này. Với thiết bị nhúng Raspberry Pi được sử dụng trong đề tài này, chương ba sẽ làm rõ những thông tin cơ bản và cấu trúc phần cứng của thiết bị. Mô hình nhận diện người nói Kaldi là thư viện được sử dụng chính trong đề tài, một số giới thiệu cơ bản cùng kiến thức tổng quan về Kaldi mà nhóm đã tìm hiểu được sẽ được nêu trong chương bốn. Chương cuối cùng sẽ là quá trình xây dựng hệ thống nhận diện người nói trên thiết bị nhúng Raspberry Pi, bao gồm các quá trình chuẩn bị, cài đặt và sau cùng là kết quả thực hiện, cũng như kết luận và hướng phát triển cho hệ thống.

Chương 2

Cơ sở lý thuyết nhận diện người nói

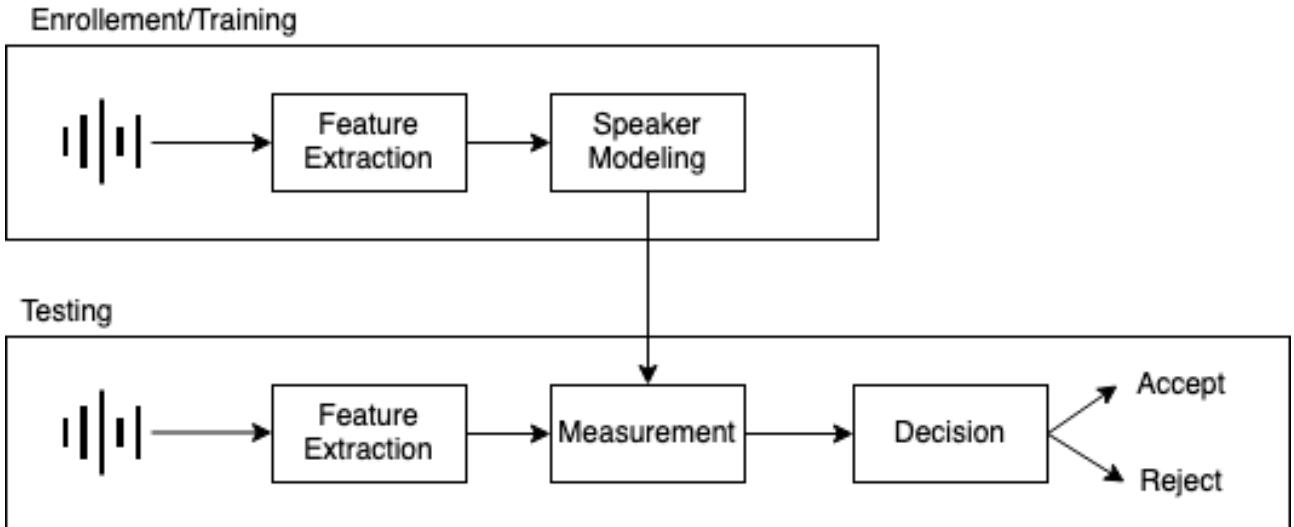
Chương này sẽ làm rõ bản chất, lý thuyết về nhận diện người nói. Đặc biệt là quá trình, nguyên lý hoạt động của một hệ thống nhận diện người nói, cùng các phương pháp, kỹ thuật sẽ được vận dụng.

2.1 Nguyên lý hoạt động hệ thống nhận diện giọng nói

Nhận diện người nói có thể được phân thành hai loại: định danh người nói (Speaker Identification) và xác minh người nói (Speaker Verification).

- Hệ thống định danh người nói là hệ thống đưa ra quyết định người nào trong số những người đã huấn luyện hệ thống đang giao tiếp với hệ thống.
- Hệ thống xác nhận người nói là hệ thống có thể xác minh danh tính của một người dựa vào người nói của người đó, so với người nói đã được đăng ký và lưu trữ.

Trong phạm vi luận văn này, nhóm sẽ tập trung nói về hệ thống xác nhận người nói.



Hình 2.1: Sơ đồ tổng quát của hệ thống xác nhận người nói

Như ở hình 2.1, một hệ thống xác nhận người nói bao gồm 2 giai đoạn:

- Giai đoạn đăng ký (training/enrollment): Người dùng sẽ đăng ký người nói và danh tính vào hệ thống .
- Giai đoạn xác minh (testing/verification): Hệ thống sẽ so sánh giữa người nói hiện tại và dữ liệu đã có trong hệ thống để đưa ra kết luận chấp nhận hay từ chối.

Chất lượng của hệ thống xác nhận người nói có thể bị giảm bởi những biến đổi của kênh và phiên giữa lúc đăng ký và lúc xác minh. Những yếu tố có thể làm biến đổi phiên/kênh bao gồm:

1. Kênh không phù hợp giữa tín hiệu âm thanh lúc đăng ký và xác minh như là sử dụng microphone khác nhau cho 2 hai giai đoạn.
2. Tiếng ồn của môi trường và âm thanh bị vang.
3. Sự khác nhau trong giọng của người nói như tuổi tác, sức khoẻ, cách ăn nói và cảm xúc.
4. Những kênh vận chuyển như điện thoại, microphone và nói chuyện qua mạng.

2.2 Phương pháp nhận diện người nói độc lập văn bản (Text - Independent) và phụ thuộc văn bản (Text - Dependent)

Phương pháp xác nhận người nói chia thành hai phương pháp chính đó là độc lập với văn bản (Text-Independent) và phụ thuộc với văn bản (Text-Dependent).

Phương pháp độc lập văn bản, người nói được yêu cầu cung cấp nội dung văn bản, chúng được sử dụng cho cả đào tạo hay nhận diện, và chúng không dựa vào văn bản đang được nói.

Đối với phương pháp phụ thuộc văn bản, thường căn cứ vào kỹ thuật đối sánh mẫu, căn cứ vào trực thời gian của người nói đầu vào và sự tương đồng giữa hai người nói đến từng âm vị, âm tiết từ đầu đến cuối lời nói. Phương pháp này mang hiệu suất nhận diện cao hơn so với phương pháp độc lập văn bản.

Trong phạm vi luận văn này, nhóm sẽ tìm hiểu và cài đặt hệ thống xác nhận người nói theo phương pháp độc lập văn bản.

2.3 Phương pháp rút trích đặc trưng

Như ở hình 2.1, tín hiệu âm thanh ở cả hai giai đoạn sẽ đều đi qua bước rút trích đặc trưng (feature extraction). Rút trích đặc trưng được sử dụng để biến đổi tín hiệu âm thanh thành các vector đặc trưng. Những vector đặc trưng này có thể đại diện cho các đặc tính cần thiết của tín hiệu âm thanh. Mục tiêu của rút trích đặc trưng là làm giảm số chiều của tín hiệu âm thanh bằng cách bỏ qua những thông tin không cần thiết mà chỉ tập trung vào những thông tin có ích cho hệ thống xác nhận người nói.

2.3.1 Phát hiện hoạt động người nói

Phát hiện hoạt động người nói (VAD) đề cập đến vấn đề xác định xem một tín hiệu âm thanh có chứa lời nói hay không. Do đó, nó là một bài toán phân lớp nhị phân.

Mô tả vấn đề

Đối với tín hiệu đầu vào x , mục tiêu của chúng ta là xác định xem đó có phải là lời nói hay không. Thuật toán VAD được biểu diễn dưới dạng một hàm $y = VAD(x)$, trong đó mục tiêu đầu ra mục tiêu mong muốn là:

$$y^* := \begin{cases} 0, & x \text{ is not speech}, \\ 1, & x \text{ is speech}. \end{cases}$$

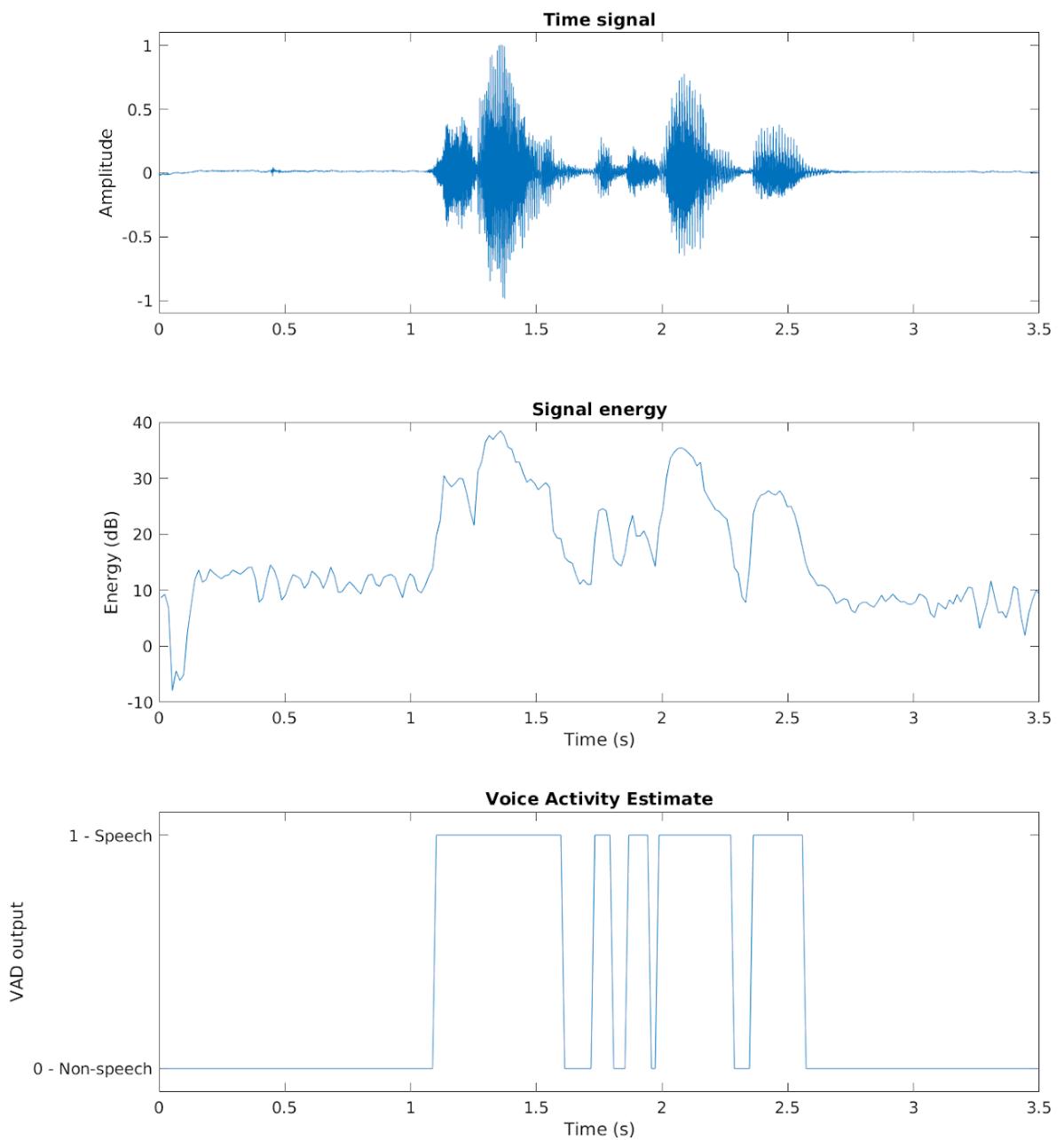
Hướng tiếp cận đơn giản nhất là VAD dựa trên năng lượng (energy-based VAD). Để thực hiện điều này, đầu tiên ta sẽ áp dụng windowing trên tín hiệu đầu vào. Với mỗi cửa sổ, ta tính được năng lượng của tín hiệu:

$$\sigma^2(x) = \|x\|^2 = \sum_{k=0}^{N-1} x_k^2$$

Tiếp theo, ta đặt ngưỡng $\theta_{SILENCE}$ sao cho khi năng lượng của tín hiệu $\sigma^2(x)$ vượt qua ngưỡng , VAD cho biết tín hiệu có giọng nói, và ngược lại không có giọng nói khi năng lượng nhỏ hơn ngưỡng.

$$VAD(x) := \begin{cases} 0, & \sigma^2(x) < \theta_{SILENCE} \\ 1, & \sigma^2(x) \geq \theta_{SILENCE}. \end{cases}$$

Để tìm được ngưỡng thích hợp, ta có thể nhiều giá trị ngưỡng cho đến khi đạt được độ chính xác cao nhất. Hoặc như hình *, ta có thể vẽ biểu đồ năng lượng của một tín hiệu mẫu.



Hình 2.2: Biểu đồ thể hiện tín hiệu đầu vào, năng lượng và ước lượng giọng nói

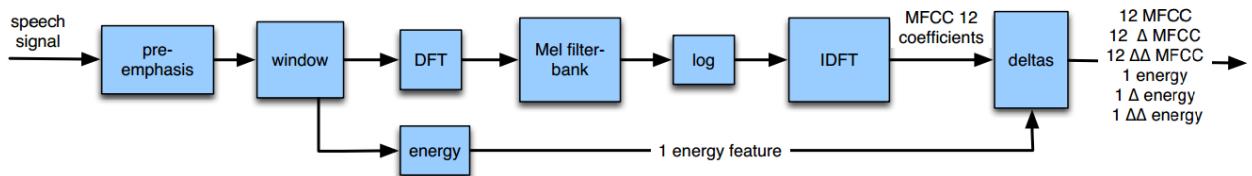
Từ biểu đồ, ta có thể thấy rằng những khoảng ít hoạt động thì có năng lượng dưới 17 dB, theo đó chúng ta có thể đặt ngưỡng ở $\theta_{SILENCE}$ là 17dB. Kết quả của VAD được mô tả ở biểu đồ cuối cùng .Ta có thể thấy kết quả là hợp lý.

Phát hiện hoạt động giọng nói dựa trên năng lượng là phương pháp đơn

giản nhưng hiệu quả cho việc rút trích đặc trưng. Ngoài phương pháp dựa trên năng lượng còn có phương pháp dựa trên mô hình và phương pháp hỗn hợp. Phương pháp dựa trên mô hình sẽ áp dụng các thuật toán máy học hiện đại, còn phương pháp hỗn hợp thì sẽ áp dụng cả hai phương pháp được nêu. Trong luận văn này, nhóm sẽ áp dụng phương pháp phát hiện hoạt động giọng nói dựa trên năng lượng.

2.3.2 Đặc trưng MFCC

MFCC[9] là viết tắt của Mel-frequency cepstral coefficients, kỹ thuật tính toán dựa trên phân tích phổ ngắn hạn của tín hiệu. Đây là các hệ số giúp mã hóa sóng âm dựa trên phương trình biến đổi chuỗi Fourier Transform. m thanh sẽ được đặc trưng bởi hai đại lượng đó là amplitude (độ lớn) và frequency (tần suất). Mỗi quan hệ giữa hai đại lượng này có dạng đồ thị sóng hình sin, biểu diễn của chúng sẽ thu được một đồ thị sóng. MFCC dựa trên việc thực hiện biến đổi để chuyển dữ liệu âm thanh đầu vào (đã qua thực hiện Fourier Transform) về thang đo tần số Mel, đây là một thang đo diễn tả tốt hơn sự nhạy cảm của tai người với âm thanh. Kỹ thuật này sẽ cho ra kết quả là các hệ số (coefficients) của cepstral từ Mel filter trên phổ lấy được từ các file âm thanh chứa giọng nói. Quá trình này bao gồm các bước biến đổi liên tiếp, đầu ra của bước biến đổi trước sẽ là đầu vào của bước biến đổi sau. Một đoạn tín hiệu giọng nói sẽ làm đầu vào cho quá trình trích chọn đặc trưng. Tín hiệu âm thanh sau khi được đưa vào máy tính sẽ được rời rạc hóa, vì vậy đoạn tín hiệu giọng nói này bao gồm các mẫu giá trị thực liên tục nhau, thể hiện giá trị biên độ âm thanh tại một thời điểm.



Hình 2.3: Sơ đồ quá trình trích chọn đặc trưng MFCC

Hình trên là sơ đồ minh họa quá trình trích chọn đặc trưng MFCC gồm 6 bước và kết quả là một tập 39 đặc trưng cho mỗi khung (frame) của tín hiệu

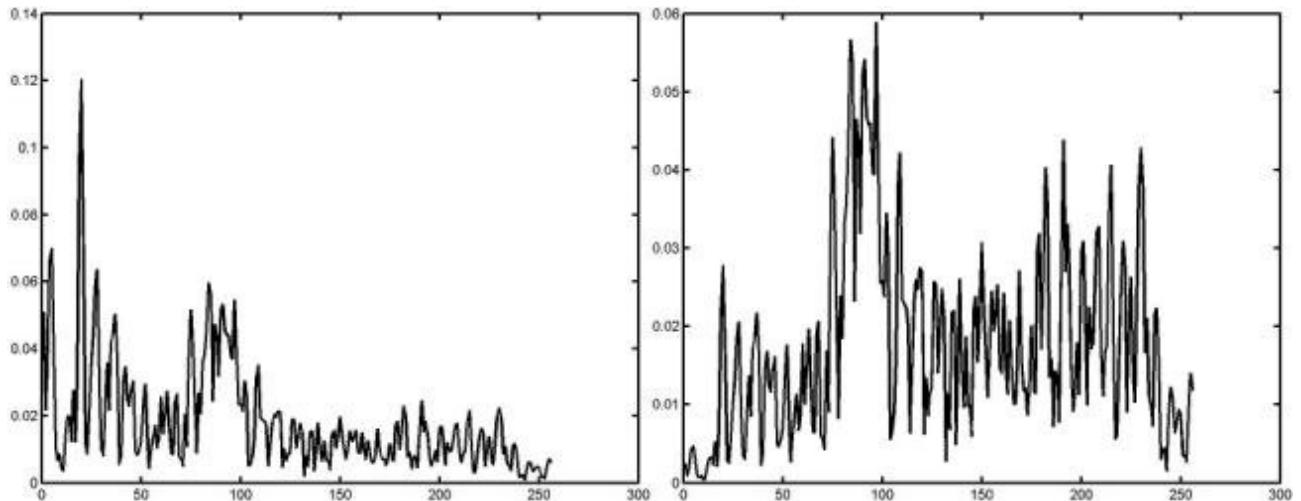
giọng nói.

Bước pre-emphasis (tiền nhấn mạnh)

Cấu trúc đặc biệt ở môi trường thanh quản cho thấy mức năng lượng của các âm hữu thanh ở tần số cao suy giảm hơn so với tần số thấp. Tuy nhiên, ở tần số cao vẫn có nhiều thông tin về formant có giá trị cho mô hình âm học (Acoustic model). Ta cần tăng mức năng lượng của tín hiệu ở tần số cao để khai thác được nhiều thông tin hơn. Bước đầu trong quá trình trích chọn đặc trưng MFCC sẽ xử lý vấn đề này thông qua bộ lọc sau thỏa công thức sau:

$$H(z) = 1 - a^{z-1}, 0.9 < \alpha < 1$$

Hình dưới đây thể hiện mức năng lượng của một âm trước và sau khi cho qua bộ lọc, ta thấy được rằng mức năng lượng ở các tần số cao được nâng lên đáng kể sau khi qua bộ lọc.



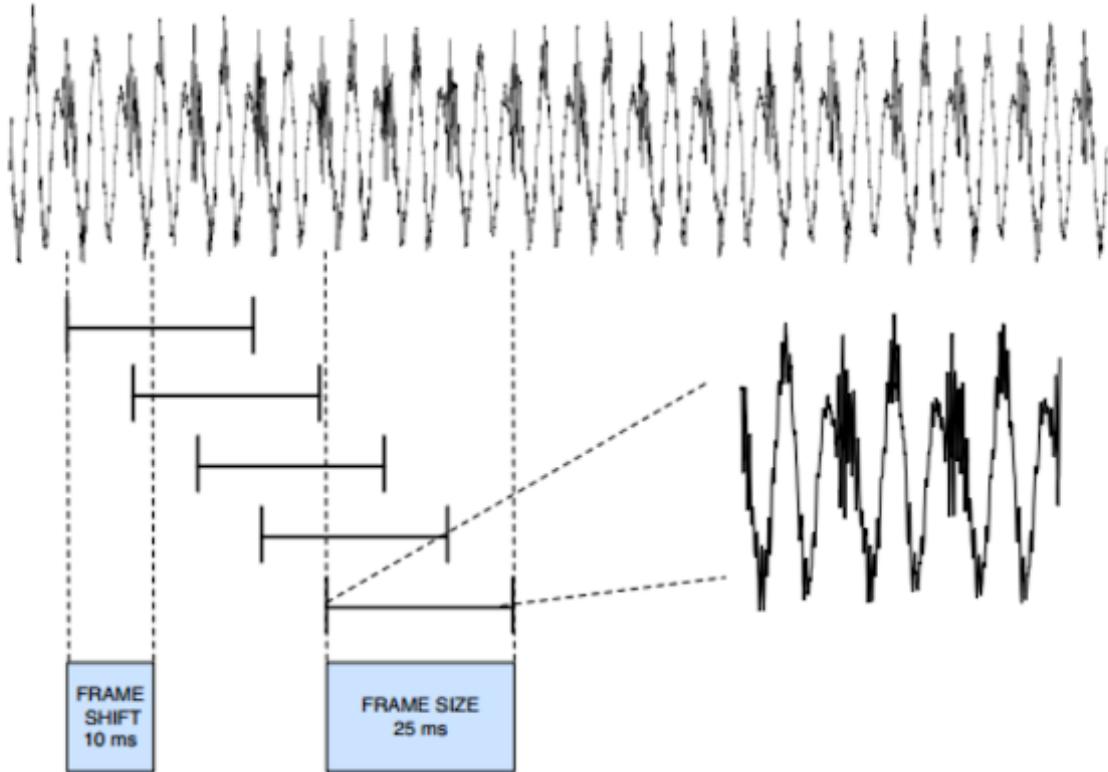
Hình 2.4: Nâng mức năng lượng của âm (hình bên trái là trước khi nâng và bên phải là sau khi nâng)

Bước Windowing

Tín hiệu giọng nói sẽ thay đổi liên tục theo thời gian, tương ứng thông tin theo đó cũng biến đổi nhiều và nếu trích chọn thông tin trên toàn bộ tín hiệu thì sẽ thiếu tính ổn định. Vì vậy ta cần sử dụng một cửa sổ nhỏ nhằm có được các đặc trưng thống kê được xem như không đổi, bằng cách trích tín hiệu trong

khoảng thời gian đủ ngắn. Thông thường của ô Hamming được sử dụng trong trường hợp này để trượt lên toàn bộ tín hiệu, trích xuất ra một loạt các frame, trong mỗi frame thông tin đặc trưng được xem là tĩnh. Quá trình này gọi là Windowing, và sau đây là cửa sổ Hamming:

$$w[n] = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N-1$$



Hình 2.5: Quá trình Windowing

Một cửa sổ (window) chạy dọc theo tín hiệu âm thanh và cắt tín hiệu nằm trong cửa sổ đó. Mỗi một cửa sổ được định nghĩa bằng các thông số sau:

- Frame shift: bước nhảy của cửa sổ, là độ dài mà cửa sổ sẽ trượt để cắt ra frame tiếp theo.
- Frame size: độ rộng của cửa sổ, quy định độ lớn của khung tín hiệu sẽ được cắt ra.

Trong loại cửa sổ này, giá trị của tín hiệu sẽ giảm dần về 0 khi tiến dần ra hai biên của frame. Nếu sử dụng cửa sổ Hamming để lấy ra các frame, năng

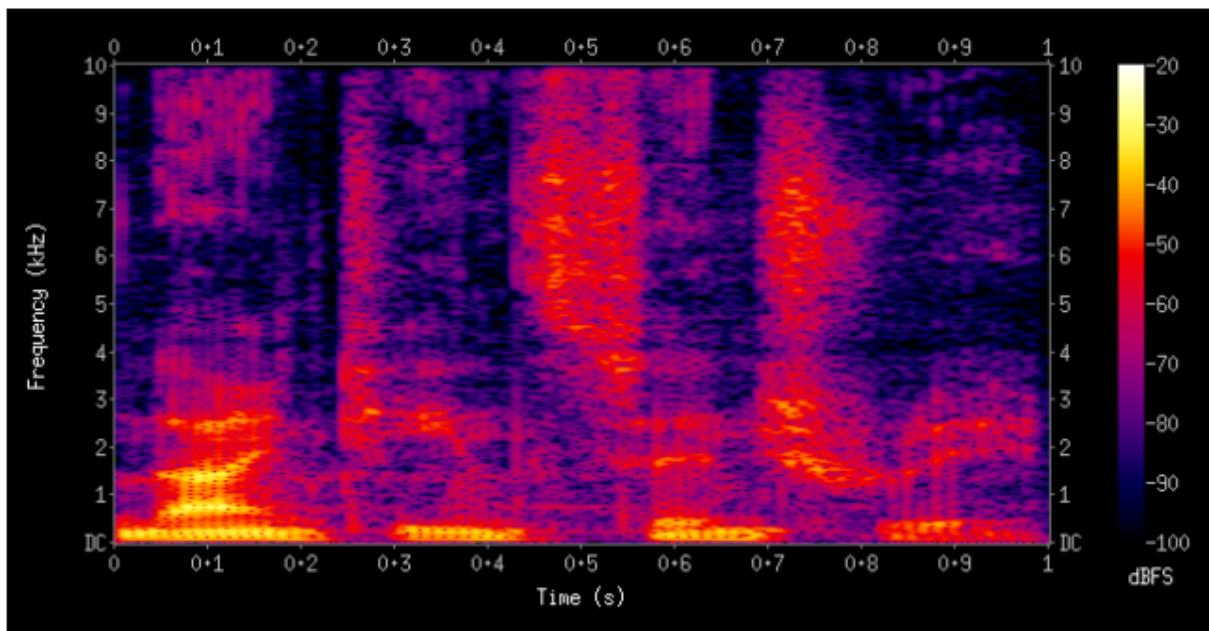
lượng của mỗi frame sẽ tập trung ở giữa frame, không những thế, các giá trị biên của cửa sổ Hamming tiến dần về 0 sẽ làm bước biến đổi Fourier ngay sau đó trở nên dễ dàng hơn.

Bước DFT (Discrete Fourier Transform)

Phổ tín hiệu sau khi nhân với cửa sổ Hamming sẽ sử dụng phép biến đổi Fourier nhanh ta thu được biên độ phổ chứa các thông tin có ích của tín hiệu giọng nói, sau quá trình này tín hiệu mỗi khung với N mẫu từ miền thời gian được chuyển sang miền tần số. FFT (Fast Fourier Transform) là thuật toán hiệu quả để tính DFT, do làm giảm độ phức tạp và thời gian tính toán của một chuỗi số tín hiệu, thích hợp trong xử lý tín hiệu thời gian thực của âm thanh.

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-\frac{j2\pi nk}{N}}, 0 \leq k \leq N - 1$$

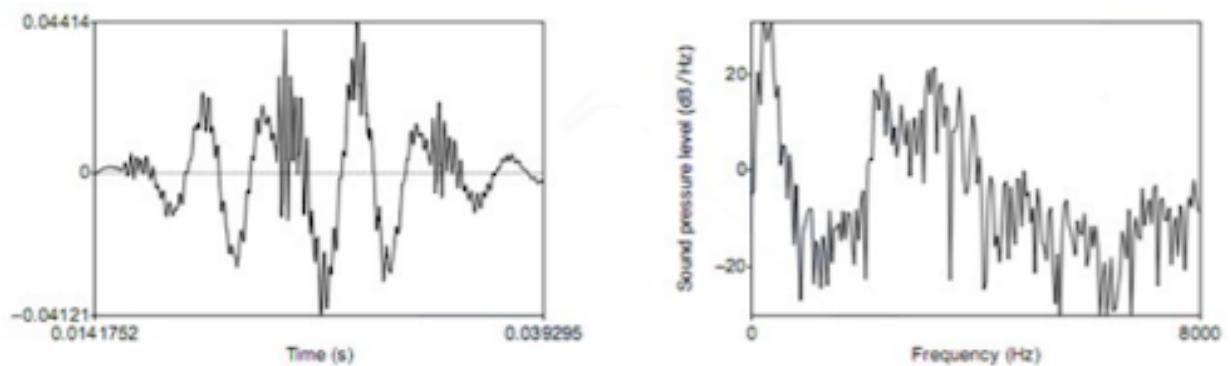
Trong đó $x(n)$ là giá trị của mẫu thứ n trong frame, $X(k)$ là một số phức biểu diễn cường độ và pha của một thành phần tần số trong tín hiệu gốc, N là số mẫu trong một frame. Mỗi frame thu được 1 danh sách các giá trị độ lớn (magnitude) tương ứng với từng tần số từ 0 đến N . Áp dụng trên tất cả các frame, ta đã thu được 1 Spectrogram như hình bên dưới. Trục x là trục thời gian (tương ứng với thứ tự các frame), trục y thể hiện dải tần số từ - đến 10000Hz, giá trị magnitude tại từng tần số được thể hiện bằng màu sắc. Qua quan sát spectrogram này, ta nhận thấy tại các tần số thấp thường có magnitude cao, tần số cao thường có magnitude thấp.



Hình 2.6: Spectrogram

Dó là cách spectrogram được tạo ra. Tuy nhiên trong nhiều bài toán, spectrogram không phải là sự lựa chọn hoàn hảo. Vì vậy ta cần thêm vài bước tính nữa để thu được dạng MFCC, tốt hơn, phổ biến và hiệu quả hơn spectrogram.

Thông thường người ta sử dụng biến đổi FFT (Fast Fourier Transform) thay vì DFT. Biến đổi FFT nhanh hơn nhiều so với biến đổi DFT, tuy nhiên thuật toán này đòi hỏi giá trị N phải là một lũy thừa của 2. Hình sau mô tả trước và sau khi biến đổi DFT của một cửa sổ:



Hình 2.7: Trước (trái) và sau (phải) khi biến đổi DFT

Bước Mel Filter-bank

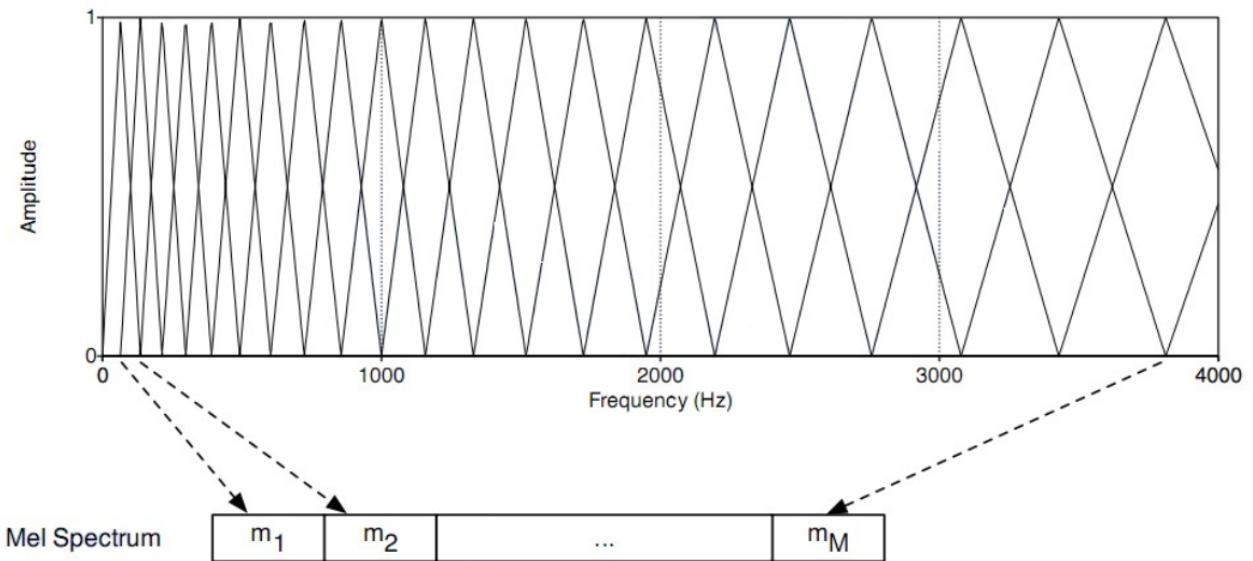
Kết quả của quá trình biến đổi Fourier thể hiện năng lượng của tín hiệu ở những dải tần số khác nhau. Tuy nhiên, tai của người lại không có sự nhạy cảm như nhau đối với mọi dải tần số. Do đó việc mô hình hóa tính chất này của tai người trong quá trình trích chọn đặc trưng làm tăng khả năng nhận diện của hệ thống.

Dầu tiên, ta bình phương các giá trị trong spectrogram thu được DFT power spectrum (phổ công suất). Kế đến áp dụng 1 tập các bộ lọc thông dải Mel-scale filter trên từng khoảng tần số (mỗi filter áp dụng trên 1 dải tần số xác định). Giá trị đầu ra của từng filter là năng lượng dải tần số mà filter đó bao phủ được, ta sẽ thu được Mel-scale power spectrum. Ngoài ra, các filter dùng cho dải tần thấp thường hẹp hơn các filter dùng cho dải tần cao.

Trong mô hình trích chọn đặc trưng MFCC, các phương trình sau dùng để chuyển đổi giữa Hert (f) và Mel (m):

$$f_{Mel} = 2595 \log_{10} \left(1 + \frac{f}{700} \right)$$

Trong đó f là tần số ở thang đo Hz, f_{Mel} là tần số ở thang đo mel. Người ta sử dụng các băng lọc để tính các hệ số mel. Sử dụng bao nhiêu băng lọc thì sẽ cho ra bấy nhiêu hệ số mel, và các hệ số mel này sẽ là đầu vào cho quá trình tiếp theo của trích chọn đặc trưng MFCC.



Hình 2.8: Các bộ lọc Mel

Cuối cùng của giai đoạn này, ta lấy logarit cơ số tự nhiên của phổ tính theo thang đo Mel, thao tác này có 2 nguyên nhân, một là do tai người nhạy cảm với âm thanh cường độ thấp hơn, hai là làm các giá trị đặc trưng nhỏ đi, tiện cho việc tính toán.

Bước biến đổi DFT ngược (Inverse Discrete Fourier Transform)

Bước tiếp theo của việc trích chọn đặc trưng MFCC là biến đổi fourier ngược với đầu vào là các hệ số phổ mel của bước trước, đầu ra sẽ là các hệ số cepstrum (MFCC – Mel Frequency Cepstrum Coefficients). Sau khi thực hiện biến đổi Fourier thì dãy tín hiệu theo thời gian đã được chuyển thành phổ tần số, và việc áp dụng các băng lọc tần số mel giúp cô đọng phổ tần số về một số hệ số nhất định (bằng với số băng lọc). Các hệ số này thể hiện các đặc trưng của nguồn âm thanh như tần số cơ bản, xung âm thanh... Tuy nhiên, các đặc trưng này không quan trọng đối với việc phân biệt các âm khác nhau. Thay vào đó, các đặc trưng về bộ máy phát âm (khoang miệng, khoang mũi, thanh quản, hầu) rất cần thiết cho việc nhận diện các âm. Việc thực hiện biến đổi fourier ngược sẽ giúp tách biệt các đặc trưng về nguồn âm và bộ máy phát âm từ các hệ số (các đặc trưng về bộ máy phát âm là các hệ số đầu tiên). Quá trình biến đổi DFT ngược sẽ tính các hệ số cepstral với công thức sau:

$$C(i) = \sum_{j=1}^K \log(|m_i|^2) \cos\left(i(j - \frac{1}{2}) \frac{\pi}{K}\right)$$

Vì mục đích cuối cùng là phân loại các âm vị trong bài toán nhận diện tiếng nói, nên ta chỉ lấy 12 hệ số cepstral đầu tiên chứa thông tin về bộ lọc đại diện cho cơ quan phát âm, và độc lập với thông tin về nguồn âm.

Phép biến đổi IDFT tương tự một phép biến đổi DCT (Discrete Cosine Transform). DCT là một phép biến đổi trực giao. Về mặt toán học, phép biến đổi này tạo ra các đặc trưng độc lập có độ tương quan kém với nhau được gọi là uncorrelated features. Sau bước biến đổi này, ta sẽ thu được 12 đặc trưng cepstral.

Năng lượng và các đặc trưng động

Từ các hệ số mel thu được từ quá trình trước, thông thường chúng ta chỉ lấy ra 12 hệ số đầu tiên để chọn làm đặc trưng. 12 hệ số này chỉ đặc trưng cho các bộ phận của bộ máy phát âm. Như vậy, chúng ta đã có 12 đặc trưng đầu tiên.

Đặc trưng thứ 13 là đặc trưng về năng lượng khung trong khoảng thời gian t được tính sau khi cửa sổ hóa Hamming được cho bởi công thức:

$$Energy = \sum_{t=t_1}^{t_2} x^2[t]$$

Trong mỗi đoạn tín hiệu biểu diễn một âm vị thì thông tin sẽ động khi chuyển từ frame này sang frame khác. Như trong một phụ âm tắc (stop consonant) khi chuyển từ trạng thái đóng sang bật âm chứa nhiều thông tin hữu ích cho việc định danh âm vị. Vì vậy mà các đặc trưng động từ các hệ số cepstral để biểu diễn thông tin từ sự thay đổi. Với 13 đặc trưng đó, chúng ta thêm vào 13 đặc trưng delta thể hiện tốc độ thay đổi của âm giữa các khung tín hiệu, được tính bằng công thức:

$$\Delta C(t) = \frac{\sum_{n=1}^N n(C_{t+n} - C_{t-n})}{2 \sum_{n=1}^N n^2}$$

Giá trị N thường được chọn là 2, là số frame trước và sau khi frame ‘t’ hiện tại. $C(t)$ là véc tơ đặc trưng của frame hiện tại (bao gồm cả năng lượng). Tiếp theo, thêm 13 đặc trưng double delta thể hiện sự thay đổi gia tốc của âm giữa các khung tín hiệu, công thức tính giống với delta với $c(t)$ là giá trị của các đặc trưng delta.

Sau cùng, ta nhận được một véc tơ đặc trưng MFCC có 39 đặc trưng bao gồm 12 hệ số cepstral, 12 delta của hệ số cepstral, 12 delta-delta của hệ số cepstral, 1 năng lượng của khung tín hiệu, 1 delta năng lượng, 1 delta-delta năng lượng.

2.3.3 GMM-UBM

Phương pháp nhận diện người nói đầu tiên được dựa trên mô hình Gaussian hỗn hợp (Gaussian Mixture Model - GMM) [12]. GMM là tổ hợp các hàm mật độ xác suất có phân phối Gauss thường được sử dụng cho việc mô hình các dữ liệu đa biến. Không chỉ gom cụm dữ liệu một cách không giám sát, GMM còn đưa ra được hàm mật độ xác suất của nó. Áp dụng GMM vào việc huấn luyện mô hình có thể tìm ra được hàm mật độ xác suất xuất của người nói, từ đó thu được xác suất của người nói. Khi kiểm thử một giọng nói chưa biết người nói, dựa trên xác suất thu được từ GMM của các người nói, có thể quyết định được giọng nói đó thuộc về ai.

GMM là một tổ hợp tuyến tính các phân phối Gauss được tham số hóa bởi các vector trung bình, ma trận hiệp phương sai và trọng số:

$$f(x; \mu, \Sigma) = \sum_{i=1}^k w_i \cdot N(x; \mu_i, \Sigma_i)$$

Trong đó w_i, μ_i, Σ_i lần lượt là trọng số, vector trung bình và ma trận hiệp phương sai của thành phần $i = 1, 2, \dots, M$. Tổng của các trọng số này phải thoả $\sum_{i=1}^M w_i = 1$. Hàm mật độ xác suất $N(x; \mu_i, \Sigma_i)$ có dạng:

$$N(x; \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{|x|/2} \sqrt{|\Sigma_i|}} \exp \left(-\frac{1}{2}(x - \mu_i)\Sigma_i^{-1}(x - \mu_i) \right)$$

Một mô hình GMM đầy đủ được tham số hóa bởi vector trung bình, ma

trận hiệp phương sai và các trọng số từ các thành phần. Các tham số này được đại diện bởi ký hiệu $\lambda = \{w_i, \mu_i, \Sigma_i\}, i = 1, 2, \dots, M$

Trong bài toán nhận diện người nói, mỗi người nói sẽ được biểu diễn bằng 1 phân phối GMM, tức là mỗi speaker sẽ có 1 model lambda riêng. Giả thiết T là số lượng vector đặc trưng MFCC của tín hiệu tiếng nói, M là số thành phần Gauss:

$$X = \{x_1, x_2, x_3, \dots, x_T\}$$

Để định danh giọng nói đã được mô hình hóa bởi λ , cần xác định khả hiện cực đại (maximum likelihood):

$$p(X|\lambda) = \prod_{n=1}^T p(x_n|\lambda)$$

Trên thực tế, lambda là hàm phi tuyến nên cần dùng thuật giải EM (Expectation Maximization) để xác định sao cho $\log p(X|\lambda)$ đạt cực đại.

Về bài toán xác nhận người nói, một hướng tiếp cận khác được phát triển. Bản chất của việc quyết định xác nhận người nói là tỉ số khả dĩ (likelihood ratio). Giả sử như chúng ta muốn xác định xem mẫu giọng nói Y có được nói bởi người S không. Nên việc xác nhận là một kiểm định giả thiết:

H_0 : Y là của người nói S

H_1 : Y không phải của người nói S

Quyết định chấp nhận H_0 dựa vào likelihood ratio (LR). Nếu LR lớn hơn ngưỡng θ , chúng ta chấp nhận H_0 , ngược lại chấp nhận H_1 .

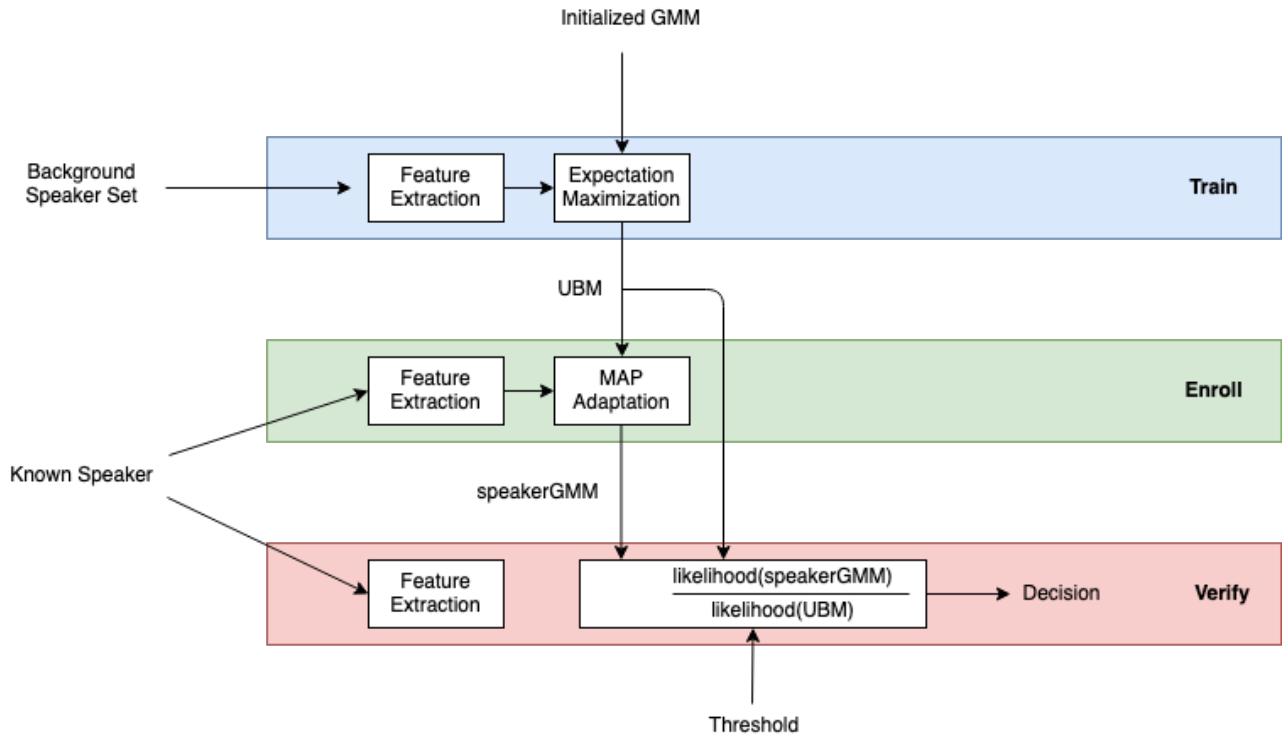
$$LR = \frac{p(Y|H_0)}{p(Y|H_1)}$$

Để thu được mô hình cho giả thiết H_1 , mô hình UBM (Universal Background Model)[1] được sử dụng. UBM là một GMM bậc cao (thường có 512 đến 2048 thành phần với 24 chiều) được huấn luyện trên một số lượng lớn giọng nói, từ tập hợp nhiều người khác nhau. Điểm lợi của UBM là có thể đại diện cho tất cả, có thể được sử dụng cho bất kì người nói nào mà không cần phải huấn luyện lại.

Sau khi có được mô hình UBM, ứng với mỗi người nói, mô hình đại diện cho

người đó sẽ được thích nghi theo mô hình UBM dựa vào thuật toán Maximum a Posteriori Adaptation (MAP). Sau đó, từ các vector trung bình của các thành phần, ta có thể xây dựng supervector bằng cách kết hợp những vector này lại với nhau tạo thành supervector:

Mô tả của hệ thống GMM-UBM như sau:



Hình 2.9: Hệ thống GMM-UBM

2.3.4 i-vector

Lấy động lực từ cách sử dụng các thành phần người nói được lấy từ JFA (Joint Factor Analysis[8]) như một đặc trưng cho thuật toán phân lớp SVM (Support Vector Machine), Dehak et al. đã đưa ra hướng tiếp cận mới gọi là i-vector[4].

Trong JFA, một câu nói của người nói được mô tả bởi một supervector, bao gồm các thành phần độc lập người nói, các thành phần phụ thuộc người nói, các kênh phụ và các yếu tố còn lại. Mỗi thành phần được đại diện bởi một tập các nhân tố có số chiều thấp.

Một mô hình đơn giản hơn đã được đề xuất, mô hình này đã loại bỏ được

sự tách biệt giữa các thành phần về người nói và kênh, sau đó mô hình hóa hai yếu tố này trong một không gian có số chiều thấp, còn gọi là không gian toàn bộ sự biến thiên (total variability space). Sự thay đổi của người nói hoặc phiên là sự thay đổi được thể hiện bởi một người nói nhất định từ phiên ghi âm này sang phiên ghi âm khác. Loại biến thiên này thường được cho là do hiệu ứng kênh mặc dù điều này không hoàn toàn chính xác vì biến thể theo người nói và biến thể ngữ âm cũng có liên quan. Trong hướng tiếp cận này, một đoạn giọng nói sẽ được biểu diễn bằng một "vector nhận diện" có số chiều thấp, gọi là i-vector.

Ý tưởng chính của hướng tiếp cận này là các thành phần phụ thuộc vào phiên và kênh của supervector GMM có thể được mô hình hóa bởi:

$$s = m + Tw$$

Trong đó m là thành phần độc lập với phiên và kênh của supervector, T là ma trận cơ sở kéo dài không gian con bao hàm sự biến thiên quan trọng (cả người nói và phiên cụ thể) trong không gian của supervector và w là một biến ẩn được phân phối theo phân phối chuẩn. Giá trị trung bình của phân phối này tương ứng chính xác với i-vector cần tìm. Thống kê Baum-Welch được sử dụng để trích xuất các i-vector này. Đây có thể coi là hướng tiếp cận giúp làm giảm số chiều của GMM supervector.

Các i-vector sau đó sẽ được đưa vào mô hình phân lớp như SVM (Support Vector Machine) hoặc dùng các độ đo như khoảng cách cosine, PLDA (Probabilistic Linear Discriminant Analysis) để so sánh sự trùng khớp của giọng nói đầu vào và giọng nói đăng ký.

2.3.5 Deep Learning

Đa phần, deep learning trong xác nhận người nói đi theo hai hướng. Một hướng là thay thế i-vector bằng các phương pháp deep learning như một bước trích xuất đặc trưng. Những nghiên cứu của hướng đi này sẽ huấn luyện một mạng lưới nơron trên các mẫu giọng nói bằng cách sử dụng các đặc trưng âm thanh như MFCC làm đầu vào và ID của người nói làm mục tiêu và sử dụng kết quả đầu ra của lớp ẩn bên trong để thay thế cho i-vector và áp dụng khoảng

cách cosine hoặc PLDA để đưa ra quyết định. Hướng đi còn lại là sử dụng deep learning để phân loại và ra quyết định, thay thế khoảng cách cosine và PLDA bằng một mạng nơron riêng biệt. Trong phạm vi luận văn này, nhóm sẽ thực hiện theo hướng đi thứ nhất. Cụ thể là sử dụng x-vector[14] để trích xuất đặc trưng và dùng PLDA để đưa ra quyết định.

X-vector

Trong hệ thống i-vector, một mạng lưới nơron hoãn thời gian (TDNN - Time-delay Neural Network)[13] được sử dụng để tính toán các “speaker embedding” từ các giọng nói có độ dài khác nhau. Speaker embedding là một không gian vector dùng để biểu diễn dữ liệu có khả năng miêu tả được mối liên hệ, sự tương đồng ứng với đặc điểm của một người nói. Sau khi có được các speaker embedding (x-vector) từ mẫu giọng nói, PLDA được sử dụng để phân lớp người nói.

Mạng nơron TDNN được huấn luyện sử dụng số lượng lớn dữ liệu để phân biệt giữa các người nói. Bảng 1 cho ta thấy kiến trúc mạng được đề xuất. Bốn lớp đầu tiên của mạng hoạt động trên các khung của mẫu giọng nói. Nếu t là mốc thời gian hiện tại, các khung $t - 2, t - 1, t, t + 1, t + 2$ được nối với nhau và tạo thành đầu vào cho các lớp sau. Đầu vào là đặc trưng MFCC 24 chiều với độ dài khung 25ms, được chuẩn hoá trên cửa sổ trượt lên tới 3ms. Hệ thống phát hiện hoạt động giọng nói (VAD) như đã nêu trên sẽ lọc ra những khung không có giọng nói.

Kích thước đầu ra của lớp thứ nhất là 512. Trong lớp thứ hai, các khung $t - 2, t, t + 2$ được nối lại nên kích thước đầu vào là $1536 (512 \times 3)$. Lớp thứ ba, các khung $t - 3, t, t + 3$ được nối lại nên kích thước đầu vào là $1536 (512 \times 3)$. Tương tự như vậy với các lớp thứ tư và thứ năm. Sau đó các đặc trưng sẽ đi qua lớp tổng hợp thống kê (statistics pooling layer). Lớp này sẽ tập hợp lại tất cả các kết quả đầu ra từ lớp thứ năm sau đó tính trung bình và độ lệch chuẩn. Quá trình này tổng hợp thông tin trên các chiều thời gian để các lớp tiếp theo hoạt động trên toàn bộ phân đoạn giọng nói. Ở bảng 1, điều này được ký hiệu bởi ngữ cảnh lớp (layer context) là 0 và tổng hợp ngữ cảnh (total context) là T. Giá trị trung bình và độ lệch chuẩn được nối lại với nhau và đi qua các lớp

còn lại cho đến lớp cuối cùng là lớp softmax.

DNN được huấn luyện để phân lớp N người nói trong tập dữ liệu huấn luyện. Một mẫu dữ liệu huấn luyện bao gồm một đoạn giọng nói (trung bình khoảng 3s) và thông tin về người nói tương ứng. Sau khi huấn luyện, các speaker embedding được trích xuất từ lớp segment6. Bỏ qua lớp softmax và lớp segment7 (vì không cần dùng nữa sau khi huấn luyện) có tổng cộng là 4.2 triệu tham số.

Layer	Layer context	Total context	Input x output
frame1	$[t - 2, t + 2]$	5	120x512
frame2	$t - 2, t, t + 2$	9	1536x512
frame3	$t - 3, t, t + 3$	15	1536x512
frame4	t	15	512x512
frame5	t	15	512x1500
stats pooling	$[0, T)$	T	$1500T \times 3000$
segment6	0	T	3000×512
segment7	0	T	512x512
softmax	0	T	512x N

Bảng 2.1: Bảng thể hiện chi tiết các lớp của TDNN

PLDA

Phân tích phân biệt tuyến tính (LDA - Linear Discriminant Analysis)[5] là một kỹ thuật giảm kích thước có giám sát. LDA chiếu dữ liệu đến một không gian con có chiều thấp hơn sao cho trong không gian con được chiếu, các điểm thuộc các lớp khác nhau được trải rộng hơn (tối đa hóa hiệp phương sai giữa các lớp) so với trải trong mỗi lớp (giảm thiểu hiệp phương sai bên trong lớp). Phân tích phân biệt tuyến tính xác suất (PLDA - Probabilistic Linear Discriminant Analysis) là phiên bản xác suất của LDA với khả năng xử lý độ phức tạp hơn trong dữ liệu.

Ban đầu, PLDA được áp dụng cho i-vector, nên sau này PLDA vẫn có thể được áp dụng cho x-vector và các loại vector thay thế khác. Cho tập hợp các i-vector d chiều được chuẩn hoá theo độ dài $X = \{x_{ij}; i = 1, \dots, N; j = 1, \dots, H_i\}$ có được từ N người nói (mỗi người có H_i i-vector), i-vector có thể biểu diễn theo dạng sau:

$$x_{ij} = \mu + Wz_i + \epsilon_{ij}$$

$$x_{ij}, \mu \in R^D; W \in R^{D \times M}; z_i \in R^M; \epsilon_{ij} \in R^D$$

$Z = \{z_i, i = 1, \dots, N\}$ là các biến ẩn, $\omega = \{\mu, W, \Sigma\}$ là tham số của mô hình PLDA, ϵ_{ij} là yếu tố nhiễu theo phân phối chuẩn với trung bình bằng 0 và hiệp phương sai Σ . Cho một i-vector kiểm thử x_t và một i-vector kết quả x_s , tỷ lệ LR có thể được tính:

$$\begin{aligned} S_{LR}\{x_t, x_s\} &= \frac{P(x_t, x_s | \text{same speaker})}{P(x_t, x_s | \text{different speaker})} \\ &= \frac{N([x_s^T x_s^T] | [\mu^T \mu^T], \hat{W} \hat{W}^T + \hat{\Sigma})}{N([x_s^T x_s^T] | [\mu^T \mu^T], \text{diag}\{WW^T + \Sigma\}, \text{diag}\{WW^T + \Sigma\})} \end{aligned}$$

Với $\hat{W} = [W^T W^T]$ và $\hat{\Sigma} = \text{diag}\{\Sigma, \Sigma\}$. Sử dụng công thức ma trận khôi nghịch đảo, tỷ lệ log-likelihood được tính bằng:

$$S_{LR}\{x_t, x_s\} = \text{const} + x_s^T Q x_s + x_t^T Q x_t + 2x_s^T P x_t$$

Trong đó $P = \Lambda^{-1} \Gamma (\Lambda - \Gamma \Lambda^{-1} \Gamma)^{-1}$; $\Lambda = WW^T + \Sigma$

$$Q = \Lambda^{-1} - (\Lambda - \Gamma \Lambda^{-1} \Gamma)^{-1}; \Gamma = WW^T$$

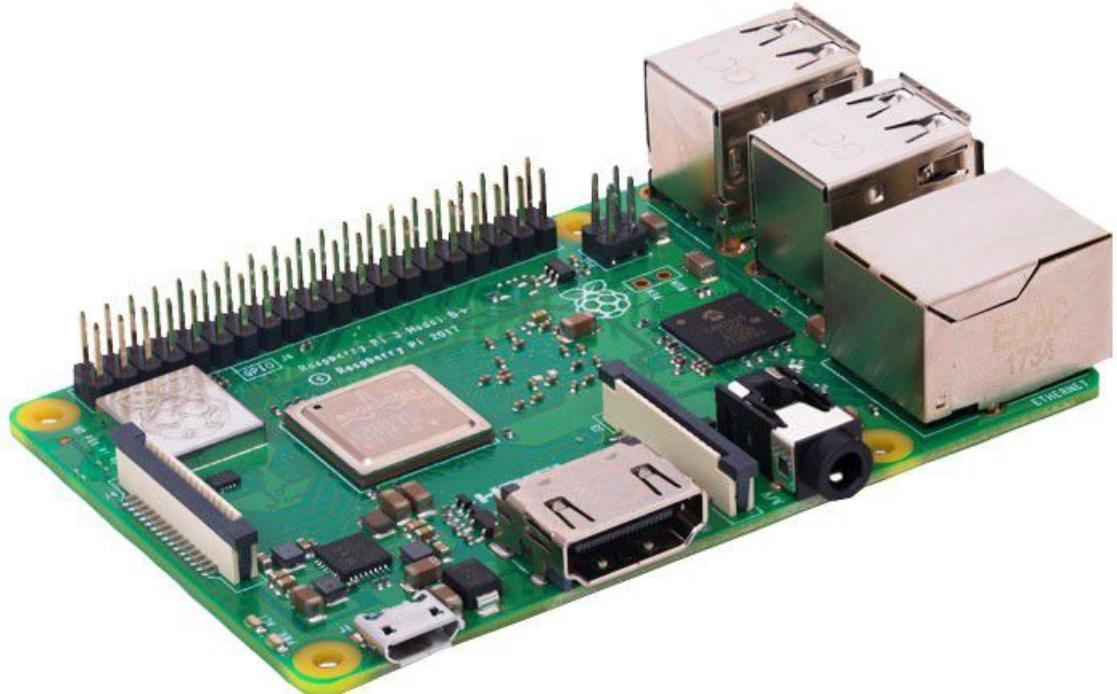
Chương 3

Tổng quan về Raspberry Pi

Raspberry Pi là một thiết bị nhúng được sử dụng trong đề tài này, mang thế mạnh là một máy tính, giúp cài đặt mô hình nhận diện và xử lý các tác vụ cần thiết như một trung tâm điều khiển.

3.1 Giới thiệu

Raspberry Pi là máy tính kích thước nhỏ được tích hợp nhiều phần cứng mạnh mẽ, đủ khả năng chạy hệ điều hành và cài đặt được nhiều ứng dụng. Với giá thành khá thấp, Raspberry hiện tại đang là mini computer nổi bật nhất, được nhiều giới công nghệ tin dùng. Raspberry Pi được phát triển dựa trên ý tưởng của tiến sĩ Eben Upton tại đại học Cambridge nhằm tạo ra chiếc máy tính giá rẻ để học sinh, sinh viên có thể dễ dàng tiếp cận và khám phá thế giới tin học. Đặc tính của Raspberry xây dựng xoay quanh bộ xử lý SoC Broadcom BCM2835 (đây là chip xử lý mobile mạnh mẽ có kích thước nhỏ, hay được dùng trong điện thoại di động) bao gồm CPU, GPU, bộ xử lý âm thanh/video và các tính năng tiện ích khác, tất cả đều được tích hợp bên trong một thiết bị nhỏ gọn.



Hình 3.1: Raspberry Pi

3.2 Cấu trúc phần cứng

Raspberry Pi[17] sản xuất bởi 3 OEM (Original equipment manufacturer): Sony Qsida, Egoman. Được phân phối bởi Element14, RS Components và Egomani.

Bộ xử lý trung tâm của Raspberry Pi là chip SoC (System On Chip) của Broadcom. Ram và Chip của Raspberry Pi đến từ Samsung và Hynix. Chip SoC

tích hợp các thành phần cần thiết bao gồm: CPU, GPU, RAM trên duy nhất 1 đế chip, tạo điều kiện cho việc thiết kế các hệ thống chạy ổn định nhưng lại yêu cầu kích thước nhỏ. SoC này khác với CPU trên máy tính, vì nó được chế tạo dựa trên kiến trúc tập lệnh (Instruction Set Architecture - ISA) là ARM, khác với kiến trúc x86 của Intel. ARM có ISA dạng rút gọn RISC và tiêu thụ điện năng rất thấp nên phù hợp với thiết bị di động. Ngược lại kiến trúc x86 có ISA dạng CISC và hoạt động với công suất cao nên dễ dàng xử lý các tác vụ phức tạp trên máy tính.

Cấu tạo chung của một Raspberry Pi gồm 10 phần chính:

1. Chip SoC (System On Chip) Broadcom BCM2837 là trái tim của Raspberry Pi. Chip tương đương với nhiều loại được sử dụng trong smartphone phổ thông, có thể chạy hệ điều hành Linux. Tích hợp trên chip là nhân đồ họa (GPU) Broadcom VideoCore IV, đủ mạnh để có thể chơi game chuẩn full HD.
2. 8 ngõ GPIO (General Purpose Input Output): các cổng kết nối có thể liên kết với rất nhiều thiết bị khác.
3. Ngõ HDMI: Giúp Raspberry Pi kết nối với màn hình máy tính hoặc TV có hỗ trợ cổng kết nối này
4. Ngõ RCA Video (analog): Giúp kết nối với các TV thế hệ cũ.
5. Ngõ audio 3.5mm: kết nối với loa và headphone.
6. Cổng USB: Cổng USB 2.0, 3.0 (ở Raspberry Pi 4) sử dụng để kết nối bàn phím, chuột, webcam, GPS, ... hay các phụ kiện máy tính khác mà không cần cài đặt driver.
7. Cổng Ethernet: Cho phép kết nối Internet thông qua mạng dây.
8. Khe cắm thẻ SD: Thẻ SD được sử dụng như ổ cứng, nơi để lưu trữ tài liệu cho Raspberry Pi, toàn bộ hệ điều hành Linux sẽ hoạt động trên thẻ SD này.

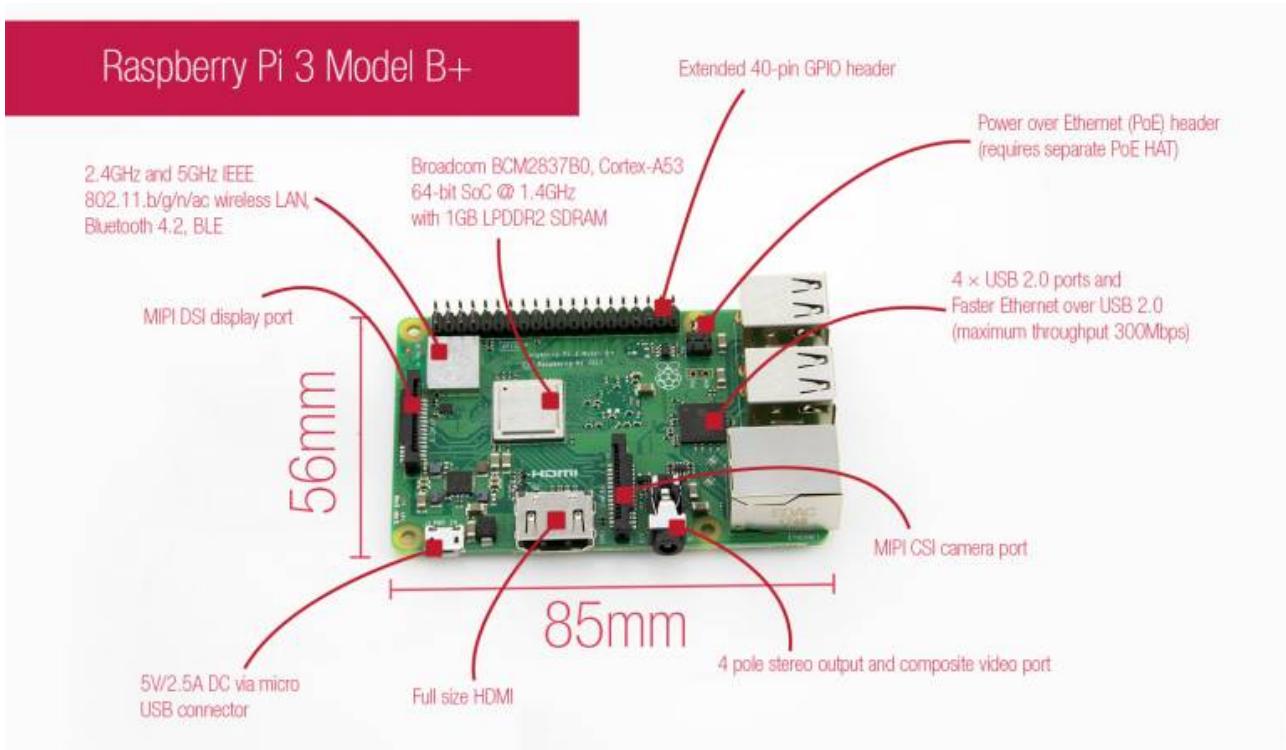
9. Đèn LED: Hỗ trợ 5 đèn LED để hiển thị tình trạng hoạt động.
10. Jack nguồn micro USB 5V, có thể dùng cáp sạc di động để kết nối và cung cấp nguồn cho Raspberry Pi.

So sánh các phiên bản Raspberry Pi:

	Raspberry Pi 4 B	Raspberry Pi 3 Model A+	Raspberry Pi 3 B+	Raspberry Pi Zero WH
Image				
Release date	2019 Jun 24	2018 Nov 15	2018 Mar 14	2018 Jan 12
Description				Same as Raspberry Pi Zero W with header already soldered
Product details				
Price	US\$35.00	US\$25.00	US\$35.00	US\$15.00
SOC				
SOC Type	Broadcom BCM2711	Broadcom BCM2837B0	Broadcom BCM2837B0	Broadcom BCM2835
Core Type	Cortex-A72 (ARM v8) 64-bit	Cortex-A53 64-bit	Cortex-A53 64-bit	ARM1176JZF-S
No. Of Cores	4	4	4	1
GPU		VideoCore IV	VideoCore IV	VideoCore IV
CPU Clock	1.5 GHz	1.4 GHz	1.4 GHz	1 GHz
RAM	1 GB, 2 GB, 4 GB	512 MB DDR2	1 GB DDR2	512 MB
Wired Connectivity				
USB	✓ 2x USB3.0 + 2x USB2.0	✓ 1xUSB 2.0	✓ 4x USB2.0	✓ micro & micro OTG
Ethernet	✓ Gigabit	✗	✓ Gigabit - Over USB 2.0	✗
SATA Ports	✗	✗	✗	✗
HDMI port	✓ 2x micro HDMI	✓	✓	✓ mini
Analog Video Out	✓ shared with audio jack	✓ shared with audio jack	✓ shared with audio jack	✓ via unpopulated pin
Analog Audio Out	✓ 3.5mm jack	✓ 3.5mm jack	✓ 3.5mm jack	✗ HDMI audio
Analog Audio In	✗	✗	✗	✗
SPI	✓	✓	✓	✓
I2C	✓	✓	✓	✓
GPIO	✓	✓	✓	✓
LCD Panel	✓	✓	✓	✗
Camera	✓	✓	✓	✓
SD/MMC	✓ microSD	✓ microSD	✓ microSD	✓ microSD
Serial	✗	✗	✗ RX/TX UART	✗
Wireless Connectivity (On-Board)				
Wi-Fi	✓ 2.4GHz and 5GHz 802.11 b/g/n/ac	✓ 2.4GHz and 5GHz 802.11 b/g/n/ac	✓ 2.4GHz and 5GHz 802.11 b/g/n/ac	✓ 802.11n
Bluetooth®	✓ 5.0	✓ 4.2, BLE	✓ 4.2, BLE	✓ 4.1
Dimensions				
Height	3.37 in (85.6 mm)	2.55 in (65 mm)	3.37 in (85.6 mm)	1.18 in (30 mm)
Width	2.22 in (56.5 mm)	2.20 in (56 mm)	2.22 in (56.5 mm)	2.55 in (65 mm)
Depth	0.43307 in (11 mm)	0.43307 in (11 mm)	0.66929 in (17 mm)	0.511181 in (13 mm)
Weight		1.02 oz (29 g)	1.58 oz (45 g)	0.42328 oz (12 g)
Power				
Power ratings			1.13 A @5V	180 mA
Power sources	USB-C	microUSB, GPIO	microUSB, GPIO	microUSB or GPIO
Power Over Ethernet	✗ with PoE Hat	✗	✗ with PoE Hat	✗

Hình 3.2: Bảng so sánh thông số kỹ thuật các phiên bản Raspberry Pi

Hiện nay, các phiên bản Raspberry được sử dụng nhiều nhất đó là:



Hình 3.3: Raspberry Pi 3 Model B+

Được ra mắt vào năm 2018, sau đây là thông số kỹ thuật chi tiết của phiên bản này:

- Vi xử lý: Broadcom BCM2837B0, quad-core A53 (ARMv8) 64-bit SoC @1.4Ghz
- RAM: 1GB LPDDR2 SDRAM
- Kết nối: 2.4GHz and 5GHz IEEE 802.11 b/g/n/ac wireless LAN, Bluetooth 4.2, BLE, Gigabit Ethernet over USB 2.0 (Tối đa 300Mbps).
- Cổng USB: 4 x 2.0
- Mở rộng: 40-pin GPIO
- Video và âm thanh: 1 cổng full-sized HDMI, Cổng MIPI DSI Display, cổng MIPI CSI Camera, cổng stereo output và composite video 4 chân.
- Multimedia: H.264, MPEG-4 decode (1080p30), H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics

- Lưu trữ: MicroSD
- Nguồn điện sử dụng: 5V/2.5A DC cổng microUSB, 5V DC trên chân GPIO, Power over Ethernet (PoE) (yêu cầu thêm PoE HAT).

Đây cũng là phiên bản được áp dụng trong đề tài này.

3.3 Hệ điều hành cho Raspberry Pi

Raspberry Pi được xem như là một máy tính mini, vì vậy để nó hoạt động cần phải được cài đặt hệ điều hành. Raspberry sẽ chạy hệ điều hành Linux, gần như tất cả những tác vụ có thể thực hiện trên Window đều có thể thực hiện trên Linux. Raspberry Pi không chạy Windows bởi vì cấu tạo kiến trúc Raspberry sử dụng chip ARM, tuy nhiên vì mục đích sử dụng, ta có thể dùng máy ảo trên Raspberry Pi. Đối với mã nguồn mở như Linux, sẽ có rất nhiều hệ điều hành tùy biến (distro) khác nhau. Tùy theo mục đích sử dụng mà chúng ta sẽ chọn distro phù hợp. Có 5 phiên bản hệ điều hành được cung cấp chính thức cho Raspberry Pi như sau:

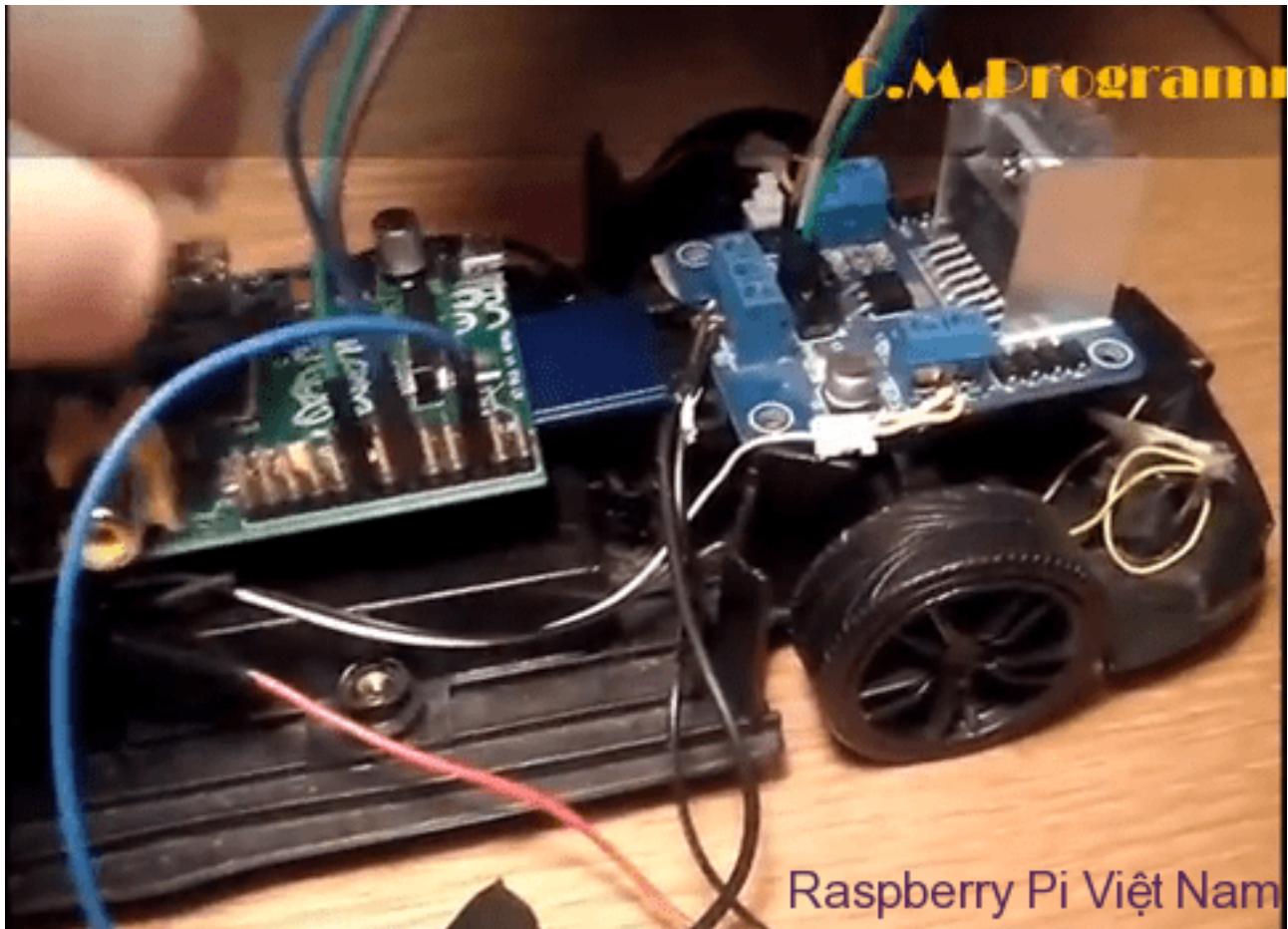
1. Raspbian (được khuyên dùng) Đây là distro dựa trên Debian wheezy, sử dụng hard-float ABI (tính toán dấu chấm động bằng phần cứng) cho thời gian chạy các ứng dụng nhanh hơn và có sẵn giao diện đồ họa. Đây cũng là hệ điều hành được sử dụng trong đề tài này.
2. Soft-float Vẫn được xây dựng dựa trên Debian wheezy, tuy nhiên việc xử lý dấu chấm động được thực hiện bằng phần mềm. Giúp chúng ta có thể sử dụng máy ảo Java (Oracle JVM) trên Raspberry.
3. Arch Linux Phiên bản dành cho ARM, đảm bảo thời gian khởi động trong 10 giây. Chỉ khởi động và load các gói cần thiết. Tuy nhiên sẽ cần kiến thức cơ bản về Linux để có thể sử dụng Arch Linux.
4. Pidora Là phiên bản của Fedora được tối ưu cho Raspberry Pi, có sẵn giao diện đồ họa.

5. RISC OS Đây là hệ điều hành được ARM thiết kế riêng.

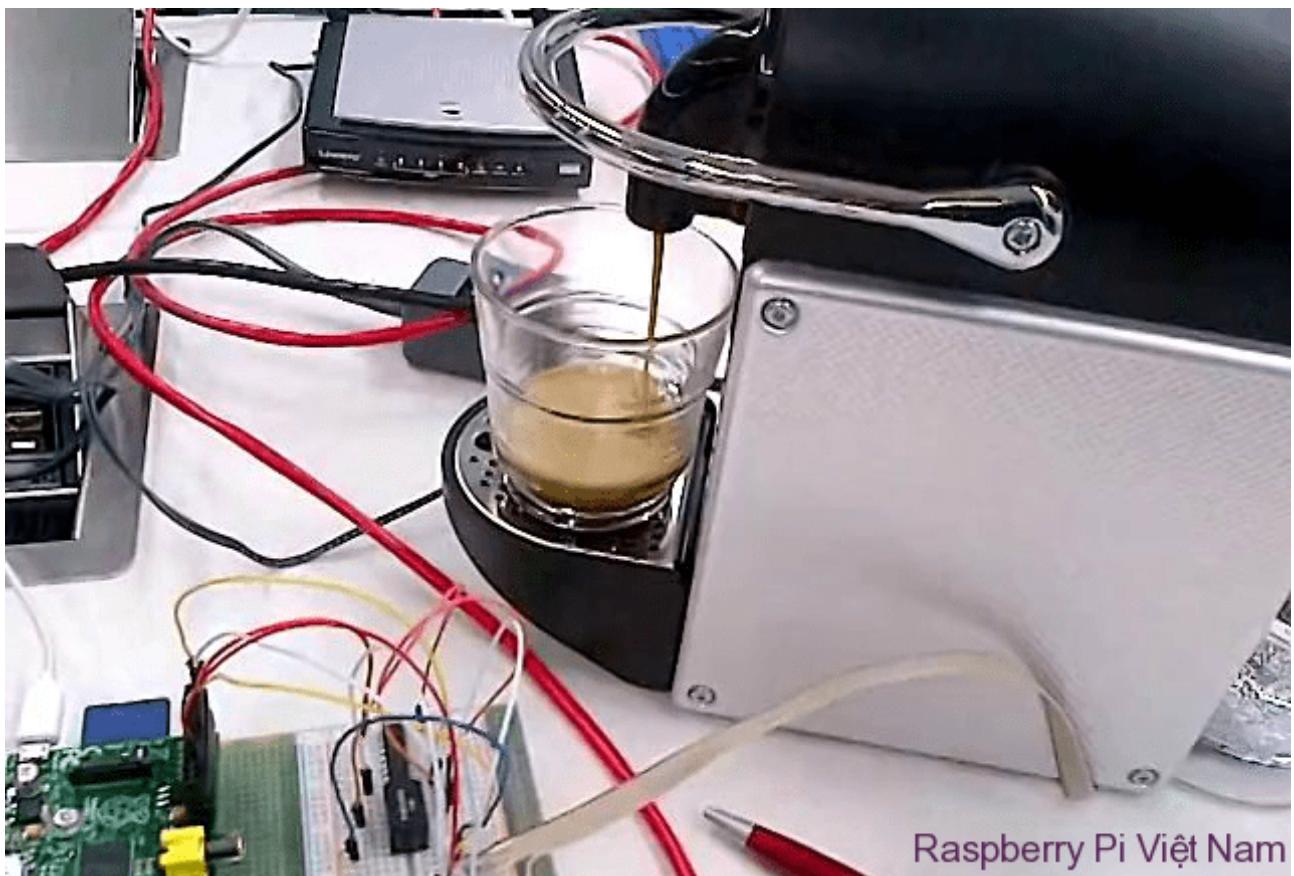
Ngoài ra sẽ còn một số hệ điều hành khác được người dùng và nhà phát triển tùy biến.

3.4 Ứng dụng

Raspberry Pi được sử dụng như một máy tính để bàn. Raspberry Pi hỗ trợ khe cắm thẻ nhớ SD, với nguồn cung cấp điện, kết nối thông qua cáp HDMI với màn hình hiển thị phù hợp, cùng với các linh kiện cơ bản của máy tính như chuột, bàn phím. Raspberry Pi cũng có sẵn tính năng Wifi, Bluetooth cùng với cổng kết nối Internet. Khi đã cài đặt xong hệ điều hành, tất cả các công cụ cần thiết để chạy Raspberry Pi như một máy tính đã được cài sẵn trên thiết bị. Hơn thế nữa, chúng ta hoàn toàn có thể cài đặt bất kỳ công cụ nào khác thông qua kho lưu trữ hoặc tải xuống qua trình duyệt. Ngoài ra có thể kể đến một số ứng dụng tiêu biểu của Raspberry Pi như: máy chủ in không dây; hệ thống media center Kodi; máy chủ game Minecraft; bộ điều khiển Robot; hoạt động như một camera; máy chủ Web; xây dựng hệ thống an ninh; ; hệ thống tự động hóa với Arduino; một thiết bị hỗ trợ việc học code, ...



Hình 3.4: Raspberry Pi ứng dụng trên xe điều khiển từ xa



Raspberry Pi Việt Nam

Hình 3.5: Raspberry Pi ứng dụng trên máy pha cà phê

Chương 4

Mô hình nhận diện người nói Kaldi

Kaldi là một công cụ được ưa chuộng trong các bài toán xử lý dữ liệu giọng nói. Được ứng dụng trong nhận dạng giọng nói, nhận dạng người nói, tăng âm, ... Đây cũng là mô hình sử dụng trong đề tài này. Tổng quan về Kaldi cũng như các cấu trúc thư mục và cách mà nhóm sử dụng sẽ được trình bày ở các phần tiếp theo.

4.1 Giới thiệu

Kaldi[3] là một bộ công cụ được phát triển vào năm 2009. Được giới thiệu tại hội thảo diễn ra tại trường Đại học Johns Hopkins University với tiêu đề (“Low Development Cost, High Quality Speech Recognition for New Languages and Domains”). Đây là bộ công cụ nhận diện người nói được viết bằng C++, được cấp phép theo giấy phép Apache 2.0. Kaldi được thiết kế cho các nhà nghiên cứu về nhận diện người nói. Mục đích là để có mã nguồn hiện đại và linh hoạt được viết bằng C++, có thể dễ dàng sửa đổi và mở rộng. Kaldi có các tính năng quan trọng: hỗ trợ số học tuyến tính mở rộng gồm một thư viện ma trận với gói BLAS và các chương trình con LAPACK; thiết kế mở rộng, bộ giải mã có thể làm việc với các mô hình khác, ví dụ như mạng nơ ron; giấy phép mở cho phép sử dụng thuận tiện. Các nhà nghiên cứu về nhận diện người nói đã

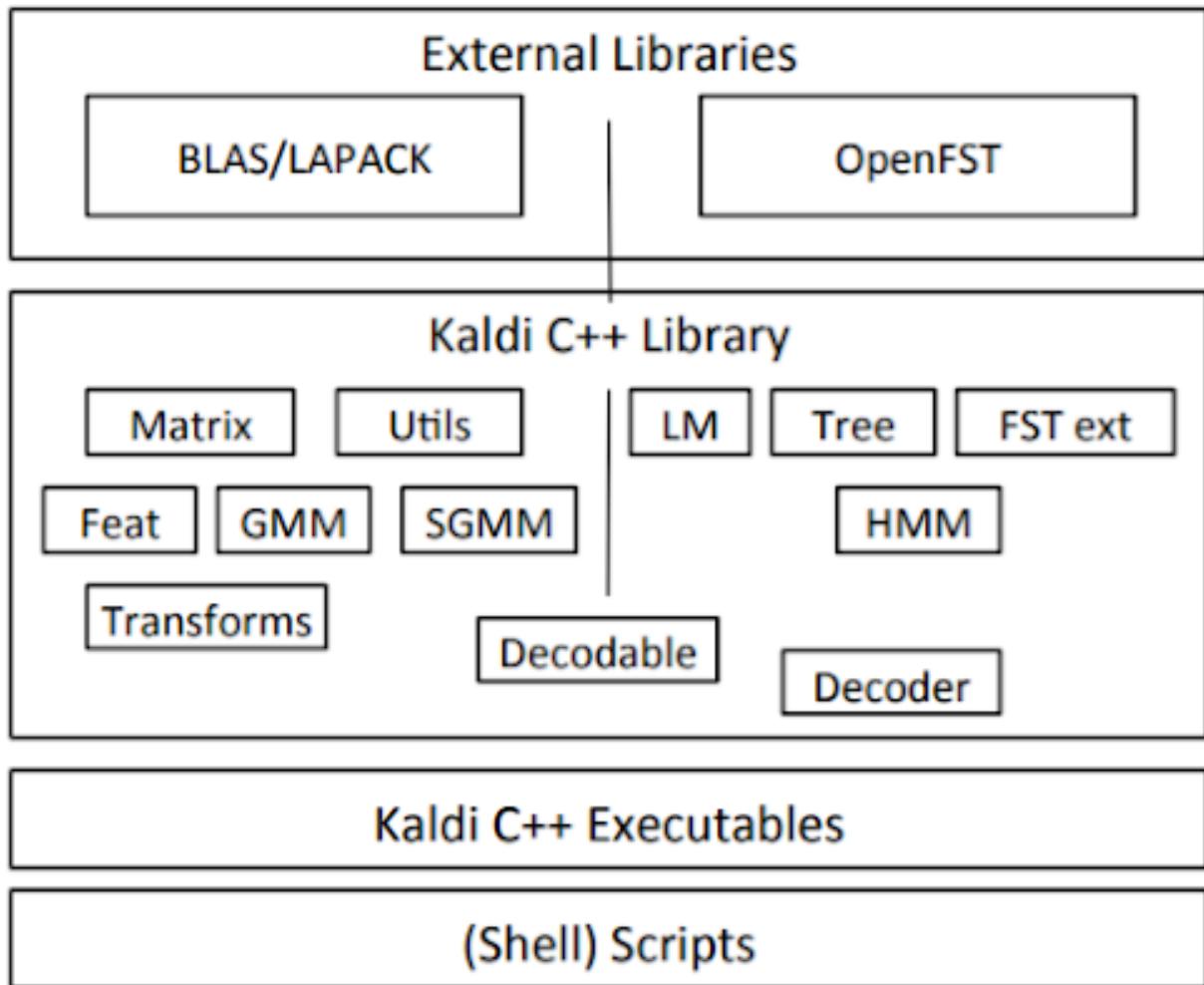
đưa ra một số lựa chọn tiềm năng về bộ công cụ mã nguồn mở để xây dựng hệ thống nhận diện. Đáng chú ý là HTK, Julius, cả hai được viết bằng C, Sphinx-4 được viết bằng Java hay bộ công cụ RWTH ASR được viết bằng C++. Tuy nhiên các yêu cầu về khung (frame) dựa trên bộ chuyển đổi trạng thái hữu hạn (FST), hỗ trợ đại số tuyến tính mở rộng, và một bộ công cụ không bị hạn chế về giấy phép đã dẫn đến sự ra đời của Kaldi cùng với những đặc điểm nổi trội:

- Tích hợp bộ chuyển đổi trạng thái hữu hạn: Biên dịch dựa trên công cụ OpenFst và sử dụng nó như một thư viện.
- Hỗ trợ đại số tuyến tính mở rộng: Bao gồm một thư viện ma trận bao bọc các quy trình BLAS và LAPACK tiêu chuẩn.
- Mở rộng thiết kế: Cung cấp thuật toán ở dạng tổng thể nhất có thể. Ví dụ, bộ giải mã hoạt động với một giao diện cung cấp điểm số cho một khung (frame) cụ thể và ký hiệu đầu vào FST. Vì vậy bộ giải mã có thể hoạt động thích hợp với bất kỳ nguồn điểm số nào.
- Giấy phép mở (open license): Được cấp phép theo Apache v2.0, đây là một trong những giấy phép có ít hạn chế nhất hiện có.
- Độ hoàn chỉnh: Cung cấp các công thức hoàn chỉnh để xây dựng hệ thống nhận diện người nói từ cơ sở dữ liệu có sẵn rộng rãi, chẳng hạn như các cơ sở dữ liệu do Linguistic Data Consortium (LDC) cung cấp.
- Được kiểm tra kỹ lưỡng: Mục tiêu là để tất cả các mã có quy trình kiểm tra tương ứng.

Mục đích chính của Kaldi là nghiên cứu mô hình âm thanh, do đó các đối thủ cạnh tranh gần nhất là HTK và bộ công cụ RWTH ASR (RASR). Lợi thế chính so với HTK là mã hiện đại, linh hoạt có cấu trúc rõ ràng, WFST và hỗ trợ toán học tốt hơn. Ngoài ra, các điều khoản cấp phép cởi mở hơn so với HTK hoặc RASR.

4.2 Tổng quan về Kaldi

Kaldi gồm một thư viện là OpenFst, các bộ chương trình dòng lệnh và kịch bản cho các mô hình âm học. Kaldi triển khai nhiều bộ giải mã để đánh giá các mô hình âm học, sử dụng huấn luyện Viterbi cho việc ước lượng mô hình âm học. Chỉ trong trường hợp đặc biệt của huấn luyện discriminative thí nghi người nói thì được mở rộng sử dụng thuật toán Baum-Welsh. Các kiến trúc của bộ công cụ Kaldi có thể được tách thành các thư viện Kaldi và các kịch bản huấn luyện. Các kịch bản này truy cập vào các hàm của thư viện Kaldi qua các chương trình dòng lệnh. Thư viện Kaldi C++ được xây dựng dựa trên thư viện OpenFST. Các hàm này có liên quan đến nhau và thường được nhóm trong một tên miền trong mã nguồn C++ mà tương ứng với một thư mục trên một hệ thống tập tin. Các ví dụ của tên miền và các thư mục được thể hiện trong hình bên dưới:



Hình 4.1: Kiến trúc bộ công cụ Kaldi

Thư viện các mô-đun có thể được nhóm thành những mô-đun phụ thuộc vào thư viện đại số tuyến tính BLAS/LAPACK và những mô-đun phụ thuộc vào OpenFst. Trong đó Decodable là lớp cầu nối giữa hai nhóm này. Các mô-đun thấp hơn trong hình sẽ phụ thuộc vào một hoặc nhiều mô-đun cao hơn.

Các mô hình xử lý dữ liệu âm thanh hầu hết hoạt động với một số biểu diễn dựa trên pixel của dữ liệu đó. Khi muốn trích xuất đại diện như vậy, chúng ta quan tâm đến hai tính năng chính, đó là xác định âm thanh lời nói của con người và loại bỏ nhiễu hay tiếng ồn không cần thiết.

MFCC (Mel-frequency cepstral coefficients) là phương pháp trích chọn đặc trưng sử dụng bộ lọc và thang tần số Mel được sử dụng rộng rãi trong ngành công nghiệp. Kaldi thực thi bằng cách tải đầu vào từ các tập tin và lưu trữ kết quả tới các tập tin một lần nữa. Ngoài ra, đầu ra của một chương trình Kaldi

có thể được đưa vào lệnh kế tiếp, sử dụng hệ thống đường ống (pipe). Thường có nhiều sự lựa chọn thay thế cho mỗi tác vụ nhận diện người nói sẽ được thể hiện trong danh sách các tập tin thực thi như sau:

- Tham số hóa người nói
 - apply-mfcc
 - compute-mfcc-feats
 - compute-plp-feats
 - ...
- Biến đổi các tham số
 - apply-cmvn
 - compute-cmvn-stats
 - fmpe-apply-transform
 - ...
- Các bộ giải mã
 - gmm-latgen-faster
 - gmm-latgen-faster-parallel
 - gmm-latgen-biglm-faster
 - ...
- Đánh giá và các tiện ích
 - compute-wer
 - show-alignments
 - ...

Kaldi cũng cung cấp kịch bản chuẩn hoặc các hàm thêm mới tiện ích. Các kịch bản được đặt tại thư mục /utils và /steps được sử dụng trong kịch bản huấn luyện và các công thức cho các dữ liệu khác nhau.

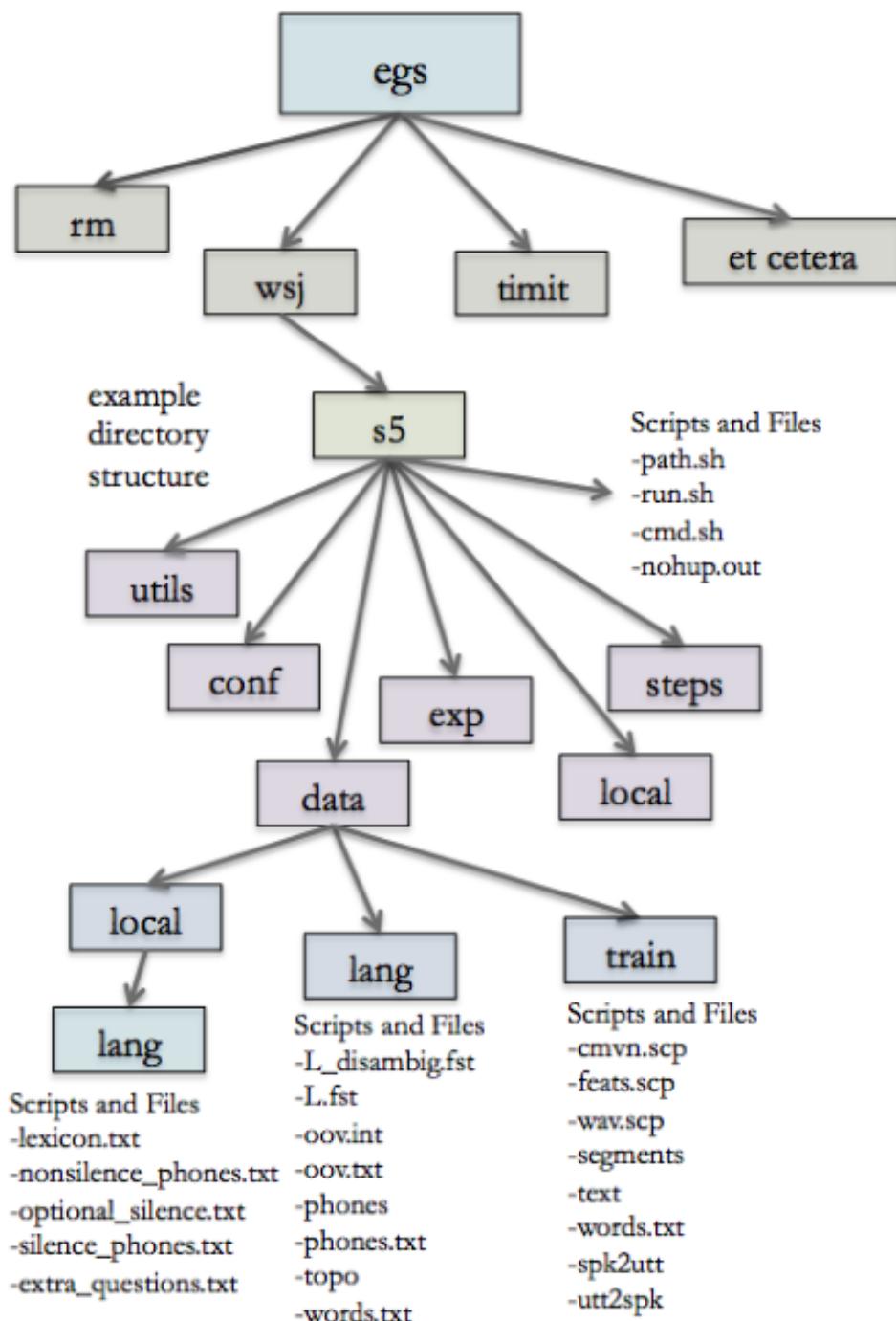
4.3 Cấu trúc thư mục trong Kaldi

Các thư mục cấp cao nhất là egs, src, tools, misc và windows. Các thư mục sẽ được dùng là egs và src.

Trong đó egs là viết tắt của examples chứa các công thức đào tạo ví dụ cho hầu hết các kho dữ liệu bài phát biểu chính. Các công thức đào tạo sẵn cho Wall Street Journal Corpus (wsj), TIMIT (timit), Resource Management (rm) và nhiều công ty khác. Dưới mỗi các thư mục là các phiên bản khác nhau (s3, s4, s5, ...) Số lượng cao nhất thường là s5, là phiên bản mới nhất và nên đường sử dụng cho bất kỳ phát triển hoặc đào tạo mới nào. Các phiên bản cũ hơn chỉ giữ cho mục đích lưu trữ.

src là viết tắt của source/source code, chứa hầu hết các mã nguồn cho các chương trình mà công thức đào tạo gọi đến.

Đối với mỗi thư mục công thức đào tạo, có một cấu trúc thư mục con tiêu chuẩn. Điều này được minh họat trong thư mục Quản lý tài nguyên (egs/rm/s5), cũng như thư mục con là conf (cấu hình), data, exp (thí nghiệm), local, steps và utils (tiện ích). Thư mục data cuối cùng sẽ chứa thông tin liên quan đến dữ liệu của riêng người dùng như bảng điểm, từ điển, ... Thư mục exp cuối cùng sẽ chứa đầu ra của các tập lệnh đào tạo và căn chỉnh, hoặc các mô hình âm thanh.



Hình 4.2: Cấu trúc thư mục egs - Kaldi

Các thư mục này chứa:

- Tập lệnh sẵn sàng khi khởi chạy, chẳng hạn như tập lệnh khởi chạy run.sh toàn bộ ví dụ và path.sh đảm bảo cấu hình phù hợp
- cmd.sh là một tập lệnh chỉ định loại tính toán đang chọn.
- conf là một thư mục chứa cài đặt cấu hình để trích xuất tính năng MFCC

và phát hiện hoạt động giọng nói dựa trên năng lượng Voice Activity Detection (VAD). Ví dụ, ta có thể cài đặt ngưỡng mà ta không phát hiện ra giọng nói.

- local chứa mã để thiết lập tập dữ liệu ở định dạng chính xác và định hình các tính năng cho x-vector pipeline
- sid khá quan trọng và chứa mã tính toán VAD, trích xuất i-vector, x-vector, đào tạo UBM, ...
- steps chứa các tập lệnh cấp thấp hơn như trích xuất tính năng, các chức năng định dạng lại dữ liệu đào tạo và các tiện ích khác.

Cách đơn giản nhất để hoạt động là khởi chạy run.sh. Đây là tập lệnh cấp cao chạy mọi thứ được đề cập. Thay vì chạy nó, nếu người dùng cần sự chuyên biệt hóa, nên chia nó ra thành nhiều bước nhỏ.

Cụ thể về quá trình đào tạo trong Kaldi, có thể tham khảo đường dẫn tài liệu sau: <https://www.eleanorchodroff.com/tutorial/kaldi/training-acoustic-models.html>

Chương 5

Hệ thống nhận diện người nói trên thiết bị nhúng Raspberry Pi

Để xây dựng hệ thống nhận diện người nói trên thiết bị nhúng Raspberry Pi. Sau khi đã nắm rõ các phần kiến thức đã được đề cập ở các phần trước. Cần chuẩn bị phần cứng và phần mềm phục vụ cho việc vận hành hệ thống. Sau đó là các bước cài đặt Kaldi trên Raspberry Pi và cấu hình hệ thống để thực hiện đúng mục tiêu đề tài. Các đánh giá, kết luận cũng sẽ được trình bày cụ thể trong chương này.

5.1 Chuẩn bị

Hệ thống bao gồm hai phần chính: phần cứng và phần mềm.

5.1.1 Phần cứng

Phần cứng của hệ thống bao gồm các thiết bị:

- Raspberry Pi 3
- USB Mic: Microphone USB đi kèm với một card âm thanh có sẵn và được

sử dụng làm thiết bị đầu vào để xác nhận người nói. Microphone có công suất thấp và có thể chạy thoải mái với nguồn được cung cấp bởi cổng USB của Raspberry Pi.

- Adapter: Adapter tiêu chuẩn 5V 2A được sử dụng để cấp nguồn cho mâm Raspberry Pi
- Thẻ nhớ: Phần cứng chứa hệ điều hành cho Raspberry Pi. Thẻ nhớ nên có dung lượng lớn hơn 8GB vì hệ điều hành thường chiếm khoảng 4GB. Ở đây nhóm sử dụng thẻ nhớ Micro SD Card 16GB
- Màn hình LCD 3.5 inch: Màn hình LCD kích thước 85 x 55 x 17mm có cảm ứng, phương thức giao tiếp chuẩn SPI được cài đặt để cho người dùng có thể thao tác với hệ thống.

5.1.2 Phần mềm

Phần mềm gồm có:

- Hệ điều hành Raspbian: Đây là hệ điều hành cơ bản, phổ biến nhất và do chính Raspberry Pi Foundation cung cấp. Theo đánh giá của nhóm, Raspbian hoạt động rất ổn định, tốc độ nhanh trên Raspberry Pi 3.
- ALSA, PulseAudio: ALSA (Advanced Linux Sound Architecture) là một hệ thống con của nhân Linux gồm các driver cho sound card và dụng cụ nhạc điện tử (MIDI), các thư viện để lập trình giao tiếp với các thiết bị nói trên và một số công cụ cần thiết (mixer,...). Hiểu đơn giản ALSA tức là driver cho sound card. PulseAudio là một dạng máy chủ âm thanh mục đích để chạy như một lớp trung gian giữa các ứng dụng và các thiết bị phần cứng, sử dụng ALSA hoặc OSS (OSS từng là driver mặc định cho các hệ điều hành trước khi bị thay thế bởi ALSA). ALSA được cài đặt sẵn trong hệ điều hành Raspbian, còn PulseAudio thì có thể cài đặt bằng lệnh:

```
sudo apt-get install pulseaudio
```

cùng với PulseAudio Volume Control (pavucontrol) là giao diện để cấu hình PulseAudio:

```
sudo apt-get install pavucontrol
```

5.2 Cài đặt

5.2.1 Cài đặt Kaldi trên Raspberry Pi

Việc cài đặt Kaldi có thể tốn nhiều RAM hơn số RAM gốc là 1GB của Raspberry Pi 3. Vì vậy một lựa chọn để cài đặt Kaldi trên thiết bị là cấp phát thêm bộ nhớ swap. Các bước cơ bản được thực hiện như sau:

1. Tạo một file sẽ được sử dụng cho swap:

```
sudo fallocate -l 1G /swapfile
```

2. Chỉ có user root có quyền đọc và ghi file. Để cấp quyền đúng, dùng lệnh:

```
sudo chmod 600 /swapfile
```

3. Dùng lệnh mkswap thiết lập file thành bộ nhớ swap:

```
sudo mkswap /swapfile
```

4. Kích hoạt swap:

```
sudo swapon /swapfile
```

5. Để lưu thay đổi, mở file /etc/fstab và viết tiếp những dòng sau:

```
/swapfile swap swap defaults 0 0
```

Để chắc chắn rằng swap đã được kích hoạt, sử dụng lệnh swapon hoặc free như sau:

```
sudo swapon --show
```

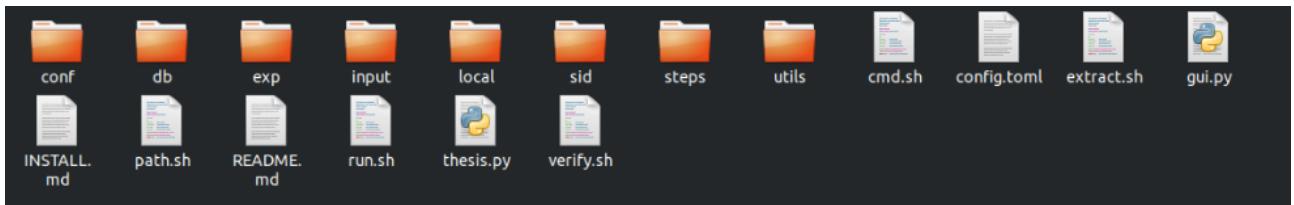
```
1 NAME      TYPE      SIZE      USED      PRIO
2 /swapfile  file     1024M    507.4M    -1
```

```
sudo free -h
```

```
1          total    used     free     shared   buff/cache   available
2 Mem:      488M    158M     83M      2.3M     246M       217M
3 Swap:    1.0G    506M    517M
```

5.2.2 Cấu hình hệ thống xác nhận người nói

Hệ thống xác nhận người nói được chứa đựng trong thư mục Speaker-Recognition-Using-Raspberry-PI. Thư mục có thể được clone về từ git. Cấu trúc thư mục:

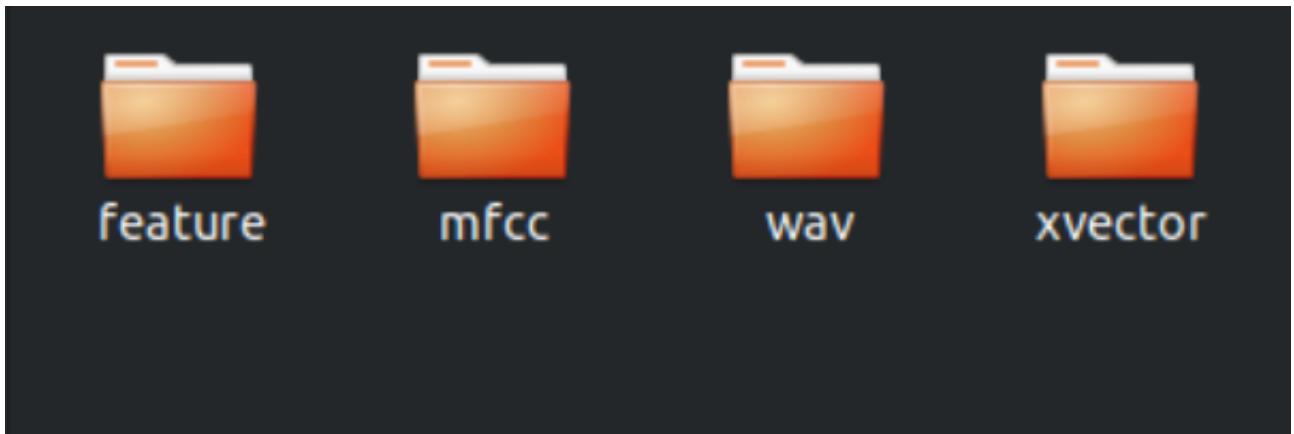


Hình 5.1: Cấu trúc thư mục

- **conf**: Cài đặt cấu hình cho MFCC, VAD bao gồm:

- MFCC:
 - * Tần số lấy mẫu: 16000Hz
 - * Kích thước: 30 chiều
- VAD

- * Nguồn năng lượng: 5.5
- **db:** Lưu dữ liệu đăng ký của người dùng. Mỗi người dùng được lưu bằng một thư mục có tên ứng với ID của người đó. Ở đây nhóm đặt ID theo dạng 001, 002, ..., 100. Cấu trúc của mỗi thư mục như sau:



Hình 5.2: Cấu trúc thư mục db

- feature: gồm có các file chứa thông tin của giọng nói như: utt2spk, spk2utt, utt2num_frames, ...
- wav: File giọng nói gốc được đăng ký
- mfcc: MFCC sau khi được rút trích
- xvector: x-vector sau khi được rút trích
- **exp:** chứa pretrained model. Mô hình có thể được tìm thấy tại: Mô hình đã được huấn luyện trên tập dữ liệu VoxCeleb. Nhóm đã tiến hành kiểm thử mô hình trên máy và đạt được độ lỗi tốt (3.128%).
- **input**
- **local, sid, steps, utils:** Bao gồm các file có sẵn của Kaldi để thao tác với dữ liệu
- **extract.sh:** lệnh dùng để rút trích và lưu đặc trưng từ giọng nói đăng ký. Cách sử dụng: extract.sh <id>, vd: extract.sh 001
- **verify.sh:** lệnh dùng để rút trích đặc trưng từ giọng nói cần xác nhận sau đó xuất ra kết quả. Cách sử dụng: verify.sh <id>, id là ID của người

dùng cần xác nhận, vd: verify.sh 001 Kết quả sau khi xác nhận sẽ được xuất ra file results, chứa chuỗi True hoặc False.

- **thesis.py**: phiên bản command line của hệ thống xác nhận.

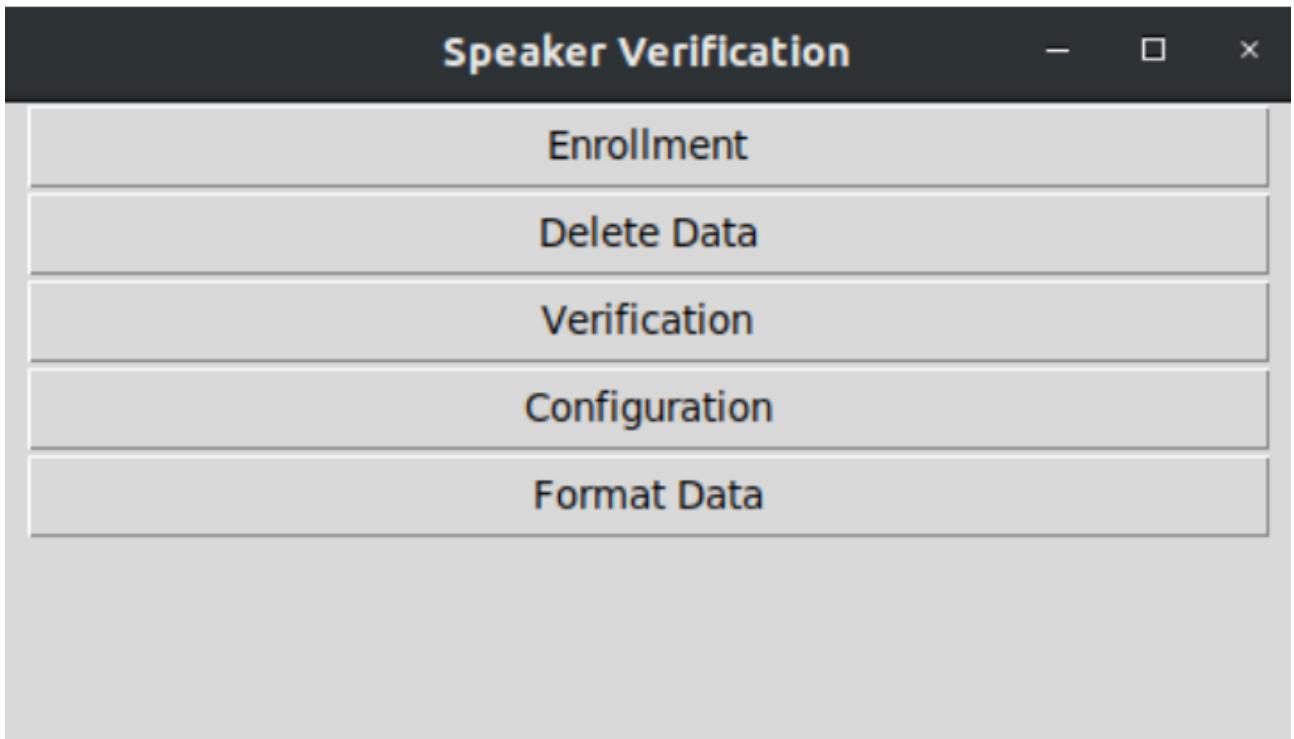
```
Welcome, 

Please choose the menu you want to start:
1. Enrollment
2. Delete Data
3. Verification
4. Configuration
5. Format Data

0. Quit
>> █
```

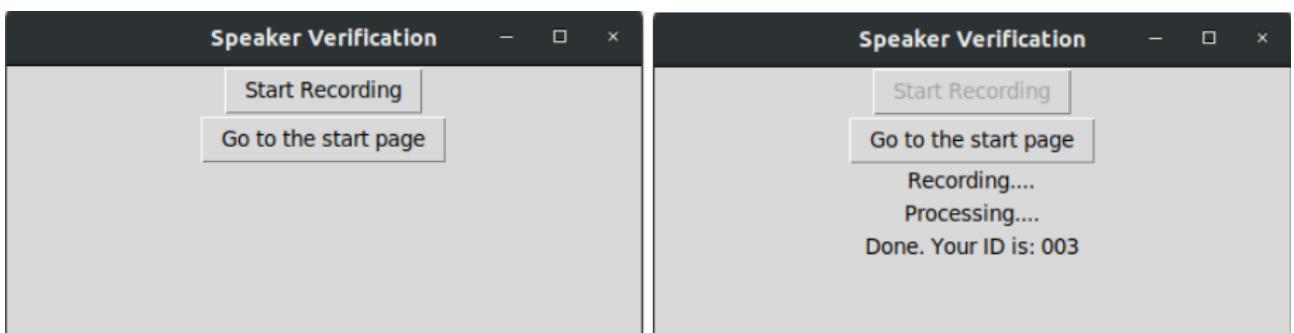
Hình 5.3: Giao diện phiên bản command line

- **gui.py**: phiên bản GUI của hệ thống xác nhận người nói. Đây là phiên bản chính được nhóm áp dụng. Chương trình gồm có các chức năng cơ bản:
 - Enrollment: Đăng ký dữ liệu người nói
 - Delete Data: Xóa dữ liệu được đăng ký
 - Verification: Xác nhận người nói
 - Configuration: Thay đổi dữ liệu được đăng ký của người nói
 - Format Data: Xóa hết dữ liệu đăng ký



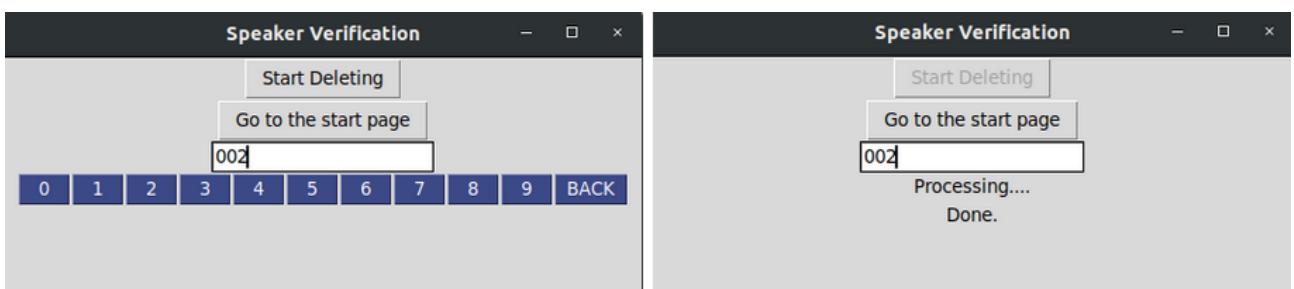
Hình 5.4: Giao diện phiên bản GUI

- Enrollment: Thu âm trong vòng 5s, sau khi thu âm sẽ chuyển sang chế độ processing: rút trích MFCC và x-vector



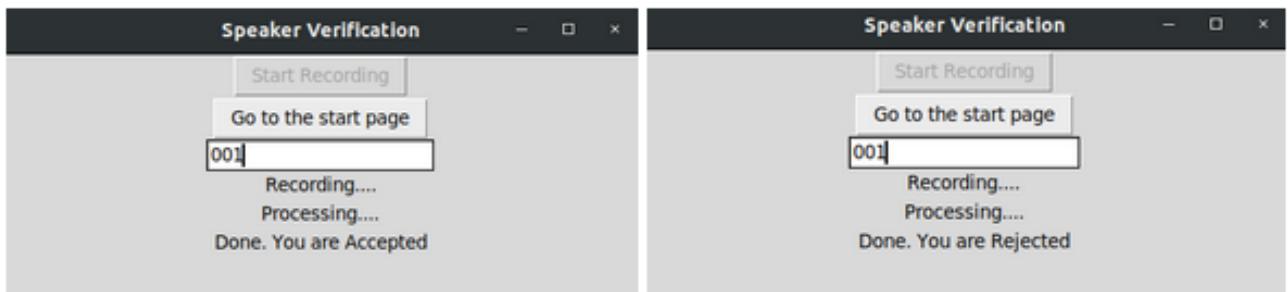
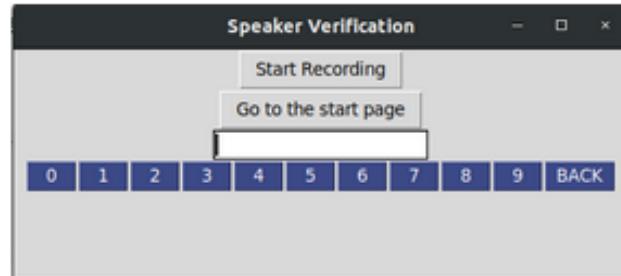
Hình 5.5: Chức năng Enrollment

- Delete Data: Nhập vào ID của người dùng để xóa



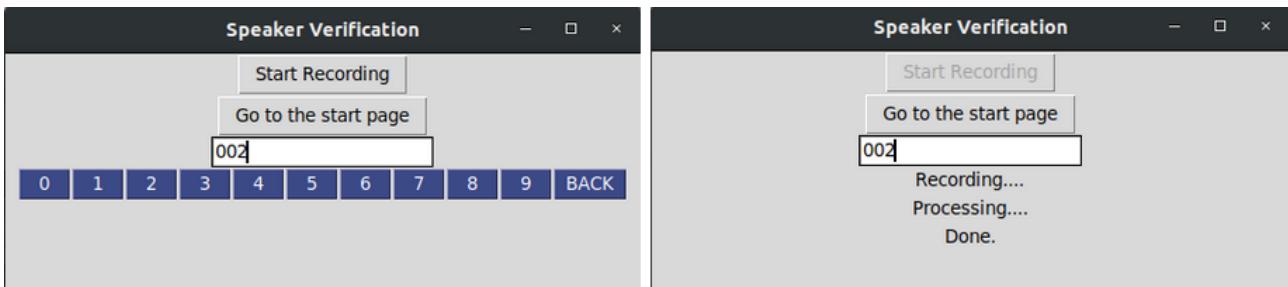
Hình 5.6: Chức năng Delete Data

- Verification: Nhập vào ID của người dùng sau đó thu âm, xử lý và cho ra kết quả. Kết quả sẽ là Accepted (Chấp nhận) hoặc Rejected (Từ chối)



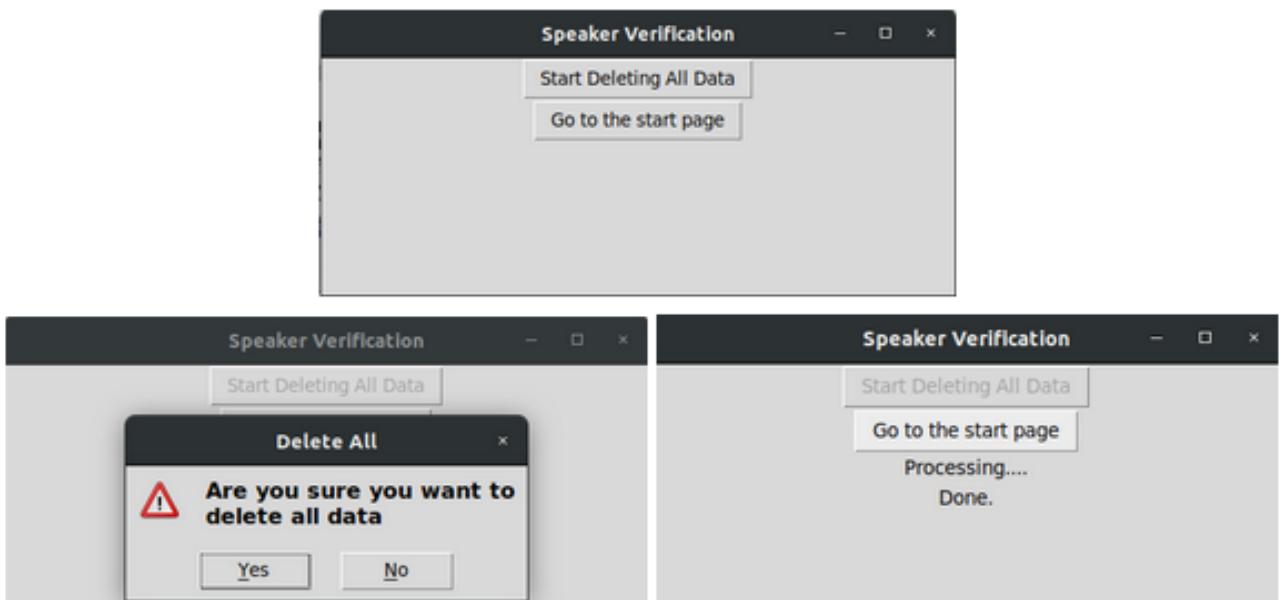
Hình 5.7: Chức năng Verification

- Configuration: Nhập vào ID của người dùng sau đó thu âm, xử lý và lưu lại đặc trưng



Hình 5.8: Chức năng Configuration

- Format Data: Việc xóa hết dữ liệu là khá nguy hiểm, người dùng cần suy nghĩ kỹ trước khi thao tác, nên hệ thống sẽ hiện ra popup khi cần xóa



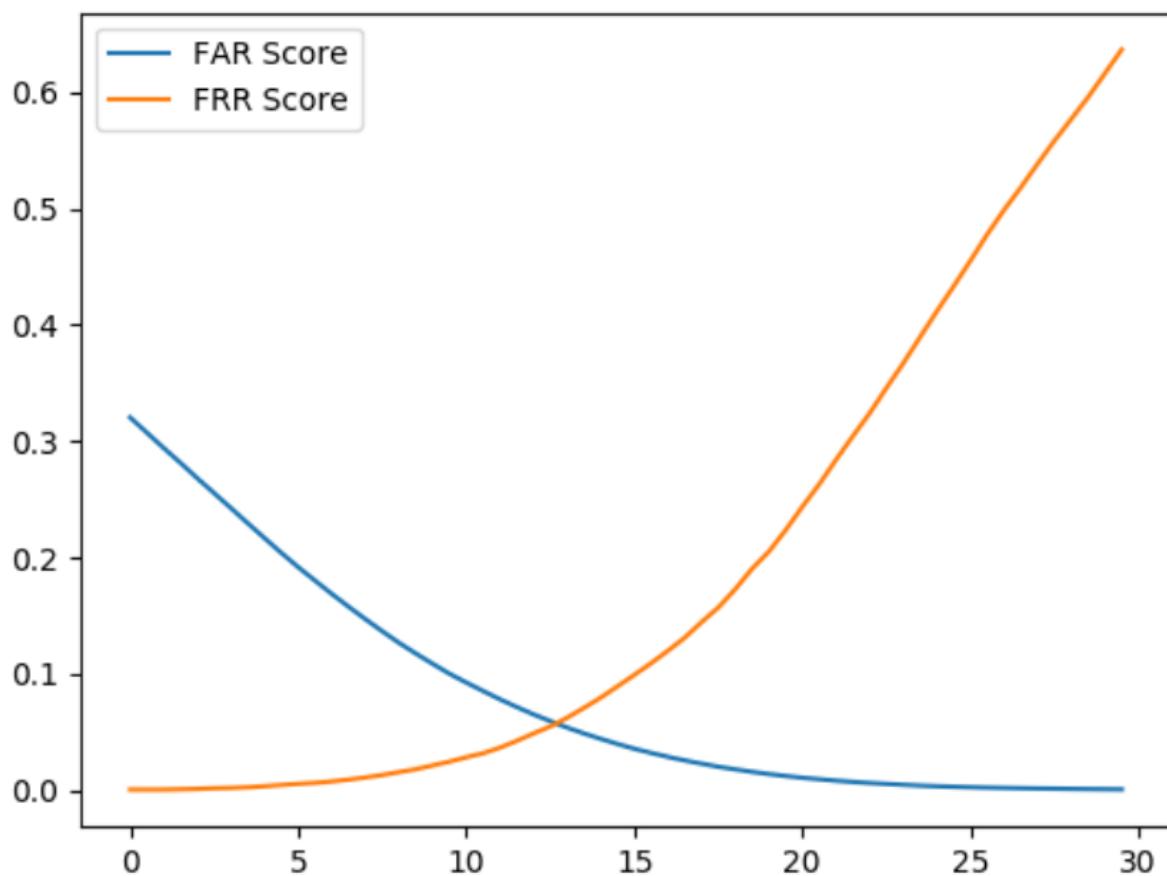
Hình 5.9: Chức năng Format Data



Hình 5.10: Hình ảnh hệ thống trong thực tế. (1): USB Microphone; (2): Raspberry Pi 3; (3): Màn hình LCD; (4): Adapter 5V 2A

5.2.3 Chọn ngưỡng PLDA

Ta cần đặt ngưỡng cho PLDA để hệ thống có thể ra được kết quả chính xác nhất. Ngưỡng PLDA được tính dựa vào tỷ lệ lỗi cân bằng. Tỷ lệ lỗi của hệ thống tại một điểm hoạt động mà FAR (False Accepted Rate - Tỷ lệ chấp nhận sai) bằng với FRR (False Rejected Rate - Tỷ lệ từ chối sai) được gọi là tỷ lệ lỗi cân bằng – Equal Error Rate (EER), giá trị này được sử dụng để biểu diễn độ chính xác của hệ thống. Giá trị EER càng thấp thì có thể nói hệ thống càng chính xác và ngược lại. Từ EER ta có thể tìm được ngưỡng ứng với EER, sau đó đặt ngưỡng đó làm ngưỡng cho việc xác nhận sau này. Việc đặt ngưỡng được thực hiện trên bộ dữ liệu tiếng Việt VIVOS Corpus từ phòng thí nghiệm Trí tuệ nhân tạo (AIIlab), thuộc Trường DH Khoa học tự nhiên TP.HCM.



Hình 5.11: Kết quả chọn ngưỡng

Đây là kết quả mà nhóm thu được. Dễ dàng thấy được $EER \approx 0.05$ và ngưỡng tại ≈ 12.5 . Nên ta sẽ chọn 12.5 làm ngưỡng xác nhận.

5.3 Kết quả

5.3.1 Chuẩn bị dữ liệu

Nhóm tiến hành tự thu thập dữ liệu bằng giọng của các thành viên trong nhóm, cùng với đó là lấy từ những video trên Youtube và thu âm bằng microphone mà hệ thống sử dụng. Nhóm đã điều chỉnh sao cho giọng nói thu được thuộc về ba điều kiện chính:

- Giọng nói lớn trong môi trường im lặng (Loud voices)
- Giọng nói nhỏ trong môi trường im lặng (Low voices)
- Giọng nói trong môi trường có nhiều tạp âm (Noisy background voices)

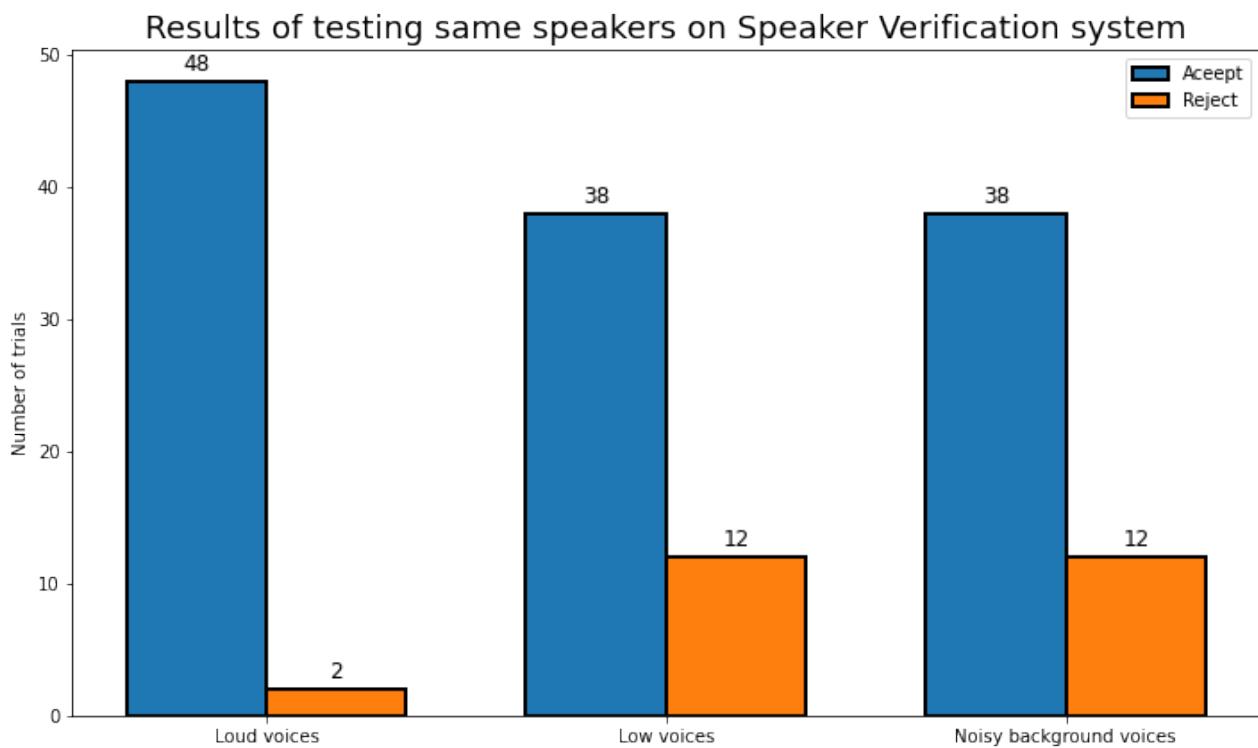
Mỗi điều kiện có dữ liệu của năm người, mỗi người có mười một giọng trong đó một giọng làm đăng ký, mười giọng còn lại dùng để xác nhận. Tổng cộng là kiểm thử trên 150 giọng nói.

5.3.2 Tiến hành và kết quả

Dầu tiên, nhóm tiến hành kiểm thử hệ thống xác nhận trên cùng một người nói và trên cùng một điều kiện. Kết quả được hiển thị như ở bảng và hình dưới:

	Accept	Reject
Loud voices	48	2
Low voices	38	12
Noisy background voices	38	12
Total	124	26

Bảng 5.1: Bảng kết quả kiểm thử trên cùng người nói



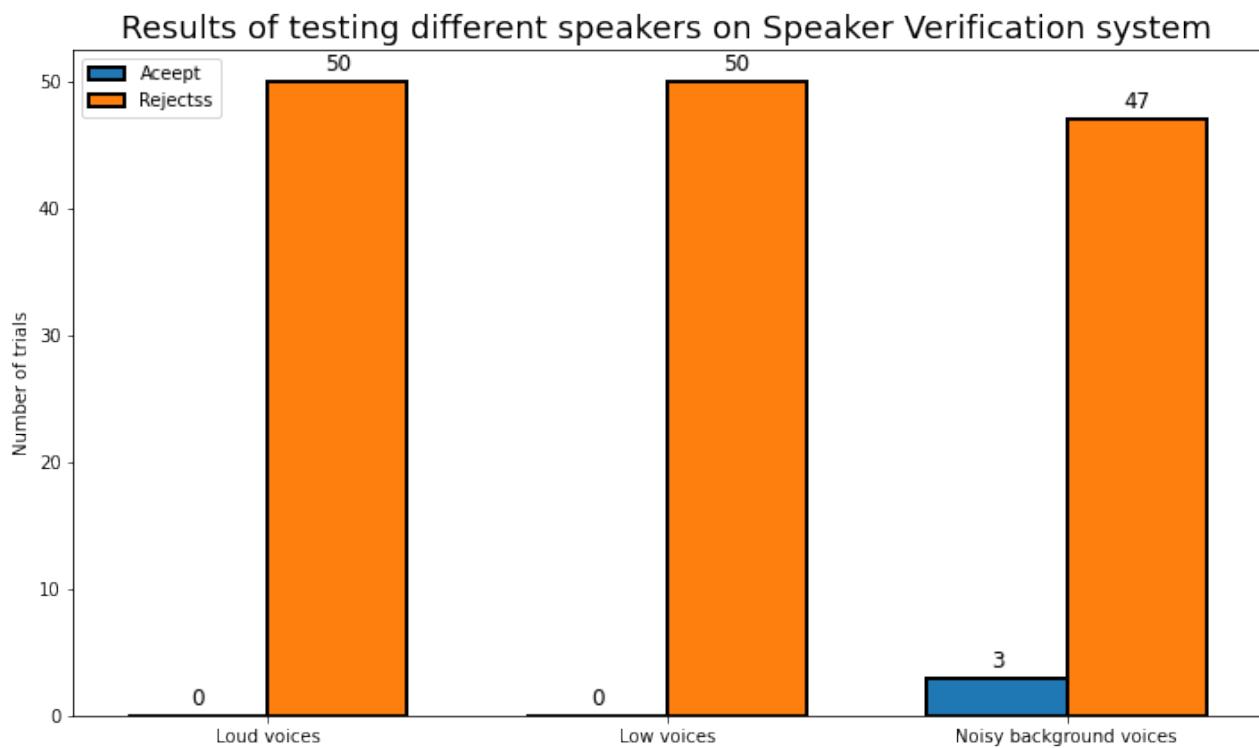
Hình 5.12: Biểu đồ thể hiện kết quả kiểm thử trên cùng người nói

Đúng như dự đoán, trong điều kiện Loud voices thì hệ thống hoạt động tốt nhất, sau đó là Low voices và Noisy background voices thì có kết quả như nhau. Lý do có thể là trong Low voices thì âm lượng người nói nhỏ quá nên bị VAD nhận nhầm là không phải giọng nói, và trong Noisy background voices thì do có quá nhiều tạp âm nên hệ thống khó có thể học được đặc trưng của người nói.

Tiếp theo, nhóm tiến hành kiểm thử hệ thống với người nói khác với người nói đăng ký và trên cùng một điều kiện. Kết quả được hiển thị như ở bảng và hình dưới:

	Accept	Reject
Loud voices	0	50
Low voices	0	50
Noisy background voices	3	47
Total	3	147

Bảng 5.2: Bảng kết quả kiểm thử khác người nói



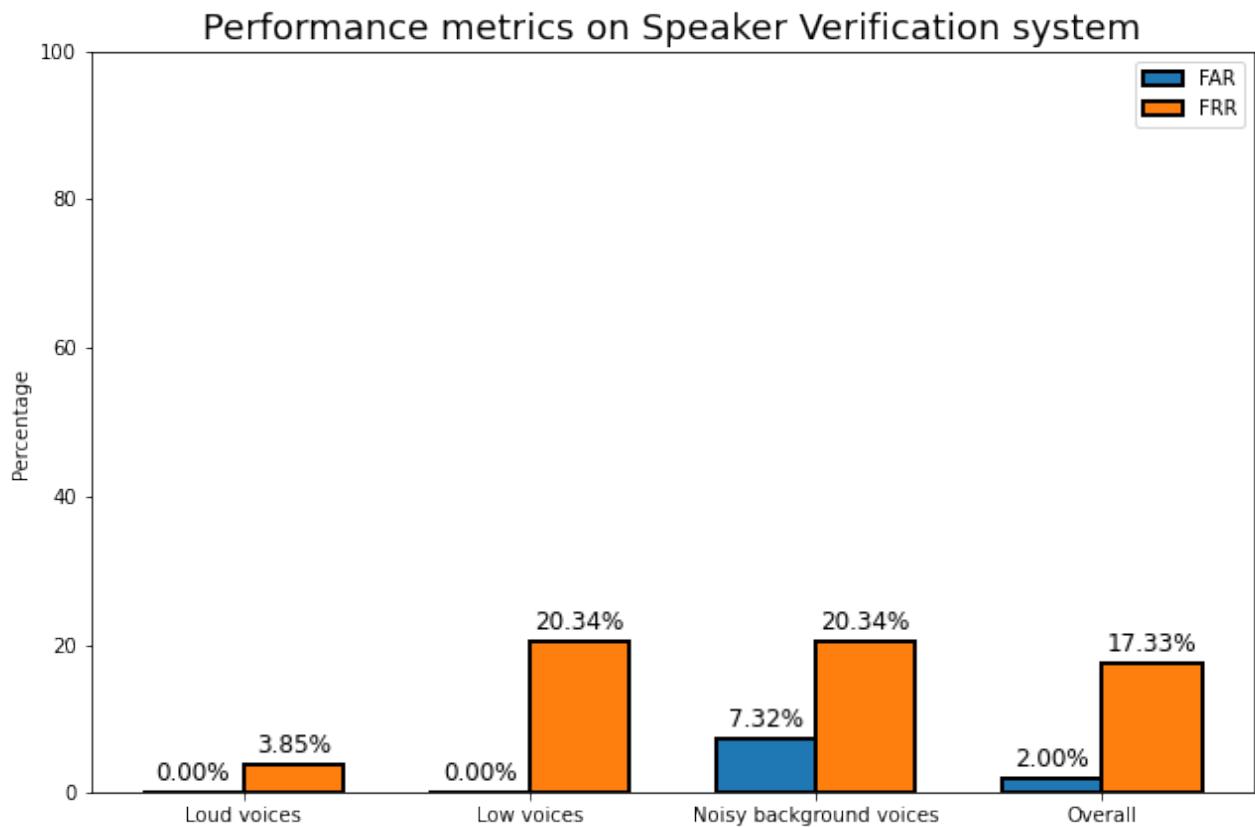
Hình 5.13: Biểu đồ thể hiện kết quả kiểm thử khác người nói

Số lần chấp nhận sai của hệ thống là rất thấp, cụ thể là không nhận nhầm với Loud voices, Low voices và chỉ nhận nhầm 3 trên 50 Noisy background Voices, tổng cộng là chỉ có 3 trên 150 giọng nói bị nhận nhầm. Tỉ lệ này là chấp nhận được, có thể cải thiện trong thực tế bằng cách sử dụng microphone chất lượng tốt hơn, hoặc thu âm trong điều kiện môi trường im lặng, tránh tiếng ồn.

Để đánh giá được hệ thống, ta cần tính tỷ lệ FAR và FRR. Hai tỷ lệ này tỷ lệ nghịch với nhau. Khi tỷ lệ FRR cao thì người dùng sẽ gặp khó khăn với hệ thống, còn khi tỷ lệ FRR quá cao thì hệ thống sẽ không có bảo mật tốt. Kết quả kiểm thử được thể hiện ở bảng và hình dưới:

	FAR	FRR
Loud voices	0%	3.85%
Low voices	0%	20.34%
Noisy background voices	7.32%	20.34%
Overall	2%	17.33%

Bảng 5.3: Bảng kết quả đánh giá hệ thống



Hình 5.14: Biểu đồ thể hiện kết quả đánh giá hệ thống

Như vậy, xét tổng thể thì tỷ lệ FAR là 2% và FRR là 17.33%. Ta thấy được tỷ lệ FRR ở mức vừa phải và FAR ở mức tốt.

5.4 Kết luận và hướng phát triển

5.4.1 Kết luận

Mục đích chính của đề tài này là sử dụng mô hình nhận diện người nói Kaldi, cài đặt và sử dụng trên thiết bị nhúng Raspberry Pi một cách độc lập với internet (offline). Từ đó kiểm tra độ chính xác của kết quả xác thực trả về chấp nhận hay từ chối danh tính một giọng nói đầu vào so với giọng nói đã đăng ký. Trong khuôn khổ luận văn này, mô hình nhận diện người nói Kaldi được sử dụng, bằng cách sử dụng pretrained model, thay đổi các tham số cấu hình để phù hợp với nhu cầu đề tài. Nhóm không can thiệp quá nhiều đến việc xây dựng mô hình. Cùng với đó, nhóm xây dựng các hình thức sử dụng trên

command line và phiên bản GUI để dễ dàng sử dụng. Việc thu thập nguồn dữ liệu để kiểm thử cũng được thực hiện đầy đủ, với nguồn thu thập cá nhân các thành viên trong nhóm và nguồn âm thanh được rút trích từ các video trên Youtube. Giọng nói sau khi thu được phân vào các nhóm khác nhau về môi trường âm thanh, với tổng cộng 150 giọng nói được kiểm thử và đánh giá. Từ kết quả cho thấy hệ thống đạt tỷ lệ chấp nhận sai sót FAR đạt 2% (ở mức tốt) và tỷ lệ từ chối sai đạt 17.33% (ở mức vừa phải).

Việc thực hiện cài đặt và vận hành trên thiết bị Raspberry Pi 3, cho hiệu suất cũng như tốc độ thực hiện công việc nhận diện còn hạn chế. Bên cạnh đó, việc đánh giá hệ thống nên được thực hiện trên bộ dữ liệu đa dạng hơn nữa. Mặt khác, tốc độ nhận diện còn chậm, do mô hình Kaldi được sử dụng lại, và nhóm không can thiệp vào sâu việc thay đổi tham số mô hình để cải thiện tốc độ nhận diện.

5.4.2 Hướng phát triển

Lĩnh vực xác thực người nói riêng là một lĩnh vực khá đang dạng về phương pháp và cách thực hiện. Vì vậy, đối với đề tài này, nhóm sẽ mở rộng nghiên cứu các tham số mô hình ảnh hưởng lớn đến hiệu suất cũng như tốc độ nhận diện của hệ thống. Bằng việc nâng cấp thiết bị phần cứng là Raspberry Pi, sử dụng thiết bị ở phiên bản mới hơn, phần cứng mạnh hơn so với thiết bị hiện tại. Đối với mô hình nhận diện người nói Kaldi, tìm hiểu và thay đổi tham số mô hình để tối ưu hóa quá trình nhận diện. Bên cạnh đó mở rộng huấn luyện mô hình về kích thước cũng như sự đa dạng về người nói huấn luyện nhằm đạt được hiệu suất tốt nhất có thể.

Bên cạnh đó, đề tài giúp tạo cơ sở cho nhóm hướng đến mục tiêu xa hơn là ứng dụng kết quả hệ thống xác thực là chấp nhận hoặc từ chối danh tính để triển khai trên những thiết bị thực tế về các hệ thống bảo mật. Không những vậy, kết hợp với nhận diện giọng nói (speech recognition) để áp dụng trong các hệ thống điều khiển bằng giọng nói. Các ứng dụng này được nêu rõ ở chương một của luận văn.

Tài liệu tham khảo

Tiếng Anh

- [1] A.Reynolds, Douglas, F.Quatieri, Thomas, and B.Dunn, Robert. “Speaker Verification Using Adapted Gaussian Mixture Models”. In: *Digital Signal Processing* 10(2000), 19–41 (2000).
- [2] Chen, Nanxin, Qian, Yanmin, and Yu, Kai. “Multi-Task Learning for Text-dependent Speaker Verification”. In: *INTERSPEECH 2015* (2015).
- [3] Chodroff, Eleanor. *Kaldi Tutorial*. <https://eleanorchodroff.com/tutorial/kaldi/index.html>.
- [4] Dehak, Najim et al. “Front-End Factor Analysis for Speaker Verification”. In: *IEEE Transactions on Audio, Speech, and Language Processing (Volume: 19, Issue: 4, May 2011)* (2010).
- [5] Ioffet, Sergey. “Probabilistic Linear Discriminant Analysis”. In: *ECCV’06: Proceedings of the 9th European conference on Computer Vision - Volume Part IV* (2006).
- [6] Kaldi. *Kaldi toolkit*. <https://kaldi-asr.org/>.
- [7] Kaldi. *VoxCeleb Models*. <https://kaldi-asr.org/models/m7>.
- [8] Kenny, Patrick et al. “Joint Factor Analysis Versus Eigenchannels in Speaker Recognition”. In: *IEEE Transactions on Audio, Speech, and Language Processing (Volume: 15, Issue: 4, May 2007)* (2007).
- [9] Liu, Xuechen, Sahidullah, Md, and Kinnunen, Tomi. “Learnable MFCCs for Speaker Verification”. In: (Feb. 2021).

- [10] Microsoft. *What is Speaker Recognition (Preview)?* <https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/speaker-recognition-overview>.
- [11] Nagrani, Arsha, Chung, Joon Son, and Zisserman, Andrew. “VoxCeleb: a large-scale speaker identification dataset”. In: *Interspeech* (2017).
- [12] Reynolds, Douglas A and Rose, Richard C. “Robust text-independent speaker identification using Gaussian mixture speaker models”. In: *IEEE transactions on speech and audio processing* 3.1 (1995), pp. 72–83.
- [13] Snyder, David, Garcia-Romero, Daniel, and Povey, Daniel. “Time delay deep neural network-based universal background models for speaker recognition”. In: *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE. 2015, pp. 92–97.
- [14] Snyder, David et al. “X-vectors: Robust DNN embeddings for speaker recognition”. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, 2018).
- [15] Sztahó, Dávid, Szaszák, György, and Beke, András. “Deep learning methods in speaker recognition: a review”. In: (2019).
- [16] Variani, Ehsan et al. “Deep Neural Network for small footprint text-dependent speaker verification”. In: *2014 IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP)* (2014).
- [17] Wikipedia. *Raspberry Pi*. https://en.wikipedia.org/wiki/Raspberry_Pi.