BÁO CÁO CHUYÊN ĐỀ

Xây dựng mô đun phần mềm huấn luyện mô hình định danh người nói:

MỤC LỤC

[LỜI MỞ ĐẦU 3](#_Toc41869842)

[I. TỔNG QUAN 4](#_Toc41869843)

[II. Tình hình nghiên cứu trong và ngoài nước 4](#_Toc41869844)

[2.1 Phân loại trong speaker recognition: 5](#_Toc41869845)

[III. ĐỀ XUẤT THUẬT TOÁN 5](#_Toc41869846)

[3.1 Các phương pháp đã áp dụng 5](#_Toc41869847)

[3.2 Bộ dữ liệu VoxCeleb2: 6](#_Toc41869848)

[3.3 Xây dựng mô đun đọc đặc trưng âm thanh 6](#_Toc41869849)

[3.3.1 Evaluation 6](#_Toc41869850)

[3.3.2 Kiến trúc trunk và ResNet 7](#_Toc41869851)

[3.4 Nghiên cứu và xây dựng mô đun tuỳ chỉnh tham số huấn luyện cho hệ thống định danh người nói 8](#_Toc41869852)

[3.4.1 Training Loss strategies 8](#_Toc41869853)

[3.4.2 Test time augmentation 9](#_Toc41869854)

[3.4.3 Chi tiết thực hiện 9](#_Toc41869855)

[3.5 Nghiên cứu và xây dựng mô đun định danh người nói. 10](#_Toc41869856)

[3.5.1 Chuẩn bị bộ dataset cho việc đánh giá nhận diện người nói 10](#_Toc41869857)

[IV. KẾT QUẢ 12](#_Toc41869858)

[TÀI LIỆU THAM KHẢO 12](#_Toc41869859)

# LỜI MỞ ĐẦU

Mặc dù có những tiến bộ gần đây trong lĩnh vực nhận dạng người nói, việc tìm các mô hình đơn giản đặc trưng cho các segment của người nói có thể được sử dụng hiệu quả trong điều kiện ồn ào và không bị ràng buộc vẫn là một thách thức đáng kể. Trong báo cáo này, chúng tôi trình bày một hệ thống embedding neural speaker dựa trên CNN sâu, có tên VGGVox, được đào tạo để ánh xạ spectrogram âm thanh đến một không gian Cosine-Euclid, tính khoảng khoảng cách tương ứng trực tiếp giữa các vector đặc trưng của người nói để tìm ra độ tương đồng. Khi một không gian như vậy đã được tạo ra, các tác vụ khác như speaker verification (xác minh người nói), clustering (phân cụm) và diarisation (phân tích) có thể được thực hiện đơn giản bằng cách sử dụng các kỹ thuật cơ bản, với các tính năng embedding.

Phương pháp ánh xạ tới không gian Cosine-Euclid như vậy đã được huấn luyện một cách hiệu quả cho nhận diện hình ảnh khuôn mặt, thông qua việc sử dụng các kiến ​​trúc Deep CNN được đào tạo trên các bộ dữ liệu khuôn mặt quy mô lớn. Thật không may, speaker recognition lại phải đối mặt với sự khan hiếm về các bộ dữ liệu quy mô lớn có sẵn miễn phí. VoxCeleb1 [1] và SITW [2] là những đóng góp có giá trị, tuy nhiên chúng vẫn thua kém rất nhiều so với các bộ dữ liệu khuôn mặt phổ biến, chứa hàng triệu hình ảnh. Để giải quyết vấn đề này, VoxCeleb2, một bộ dữ liệu nhận dạng người nói quy mô lớn được lấy tự động từ các phương tiện media nguồn mở. VoxCeleb2 bao gồm hơn một triệu cách nói từ hơn 6000 người nói. Vì bộ dữ liệu được thu thập tự nhiên không qua xử lý, sẽ có những phân đoạn lời nói bị hỏng bởi tiếng ồn trong thế giới thực bao gồm tiếng cười, nói chuyện chéo, hiệu ứng kênh, âm nhạc và các âm thanh khác. Bộ dữ liệu cũng là đa ngôn ngữ, với những bài phát biểu của các diễn giả thuộc 145 quốc tịch khác nhau, bao gồm nhiều giọng nói, độ tuổi, dân tộc và ngôn ngữ.

Chúng tôi thực hiện training VGGVox trên bộ dữ liệu này để tìm phần embedding phân biệt người nói. Hệ thống bao gồm ba phần biến chính: một kiến trúc underlying deep CNN trunk, được sử dụng để trích xuất các features, một pooling method được sử dụng để tổng hợp các features, cung cấp một embedding cho một cách nói nhất định và một pairwise loss được đào tạo về các features để trực tiếp tối ưu hóa ánh xạ. Thử nghiệm kiến ​​trúc trunk CNN dựa trên VGG-M [3] và ResNet [4].

Bộ dữ liệu VoxCeleb2 dataset có thê download từ http: //www.robots.ox.ac.uk/~vgg/data/voxceleb2.

# TỔNG QUAN

Tiếng nói là phương tiện trao đổi thông tin phổ biến nhất của con người. Nhận dạng người từ giọng nói hay nhận dạng người nói (speaker recognition) cùng với nhận dạng tiếng nói (speech recognition) là những lĩnh vực nhận dạng liên quan đến xử lý tiếng nói đang được quan tâm nghiên cứu hiện nay. Tiếng nói, ngoài thông tin ngữ nghĩa mà người nói muốn truyền đạt cho người nghe (những thông tin có thể ghi lại dưới dạng chữ viết), còn chứa những thông tin khác như phương ngữ, trạng thái tình cảm khi nói cũng như những thông tin riêng của giọng nói. Trong khi nhận dạng tiếng nói dựa trên thông tin ngữ nghĩa thì nhận dạng người nói lại dựa vào các thông tin riêng của giọng nói.

Lĩnh vực nhận dạng/xác minh người nói đã mở rộng ra rất nhiều nghiên cứu và được ứng dụng rộng rãi trong đời sống xung quanh chúng ta, một số lĩnh vực quan trọng có thể thấy như: Truy cập và điều khiển các máy tính, thiết bị điện tử, hệ thống khóa bảo mật thông minh, nhận dạng tội phạm, nhận thông tin phản hồi,…



Hình 1:Thomas Edision, nhà phát minh vĩ đại người Mỹ bên cạnh chiếc máy ghi âm đầu tiên do ông chế tạo

# Tình hình nghiên cứu trong và ngoài nước

Các nghiên cứu về xác minh và định danh người nói là nghiên cứu quan trọng trong các vấn đề nghiên cứu về xử lý tiếng nói. Trong xác minh định danh người nói, trở ngại dễ thấy nhất chính là việc lời nói của một người dễ dàng bị ảnh hưởng bởi trạng thái tình cảm hay thông tin riêng của giọng nói, các thông tin này thay đổi từ lúc chúng ta còn bé, đến khi trưởng thành và về già. Tuy không bất biến nhưng những thông tin này lại có tính ổn định khá trong những đoạn dài trong cuộc đời. Thêm nữa, khi con người trưởng thành, sự phát triển của thanh quản khiến người đấy có những thói tật khi nói, những đặc trưng khu biệt trong cấu âm sẽ hình thành và mang tính ổn định cao.

## 2.1 Phân loại trong speaker recognition:

Có rất nhiều công trình nghiên cứu về định dạng người nói dựa trên phân loại nhiều loại tác vụ khác nhau, tuy nhiên có thể chia ra hai loại tác vụ chính thường được sử dụng bao gồm:

* Speaker identificaion (xác định xem ai là người đang nói)
* Speaker verification (

Nếu người nói cần được xác minh đã có danh tính nhất định và giọng nói được sử dụng để xác minh ở đây, đây được gọi là verification hoặc authentication. Mặt khác, identification là nhiệm vụ xác định danh tính của người nói không xác định được danh tính. Có nghĩa, identification là khi chúng ta có 2 hay nhiều đoạn ghi âm: nhưng không xác minh được danh tính của người nói trong những đoạn như vậy, identification được sử dụng để tìm ra các đặc trưng của mỗi đoạn ghi âm, so sánh với nhau để xác nhận đó là cùng một người nói hay là nhiều người khác nhau.

Từ góc độ bảo mật, identification khác với verification. Speaker verification thường được sử dụng như một "người gác cổng" để cung cấp quyền truy cập vào một hệ thống an toàn. Các hệ thống này hoạt động với sự nhận biết người dùng và thường yêu cầu sự hợp tác của họ. Speaker identification có thể được triển khai một cách tình cờ mà không có sự nhận biết về người nói để xác định người nói trong cuộc thảo luận, cảnh báo hệ thống tự động thay đổi người nói, kiểm tra xem người dùng đã đăng ký vào hệ thống chưa, v.v.

Trong các ứng dụng pháp y, đầu tiên người ta thường thực hiện quy trình speaker identification để tạo danh sách "kết quả phù hợp nhất" và sau đó thực hiện một loạt các quy trình verification để người nói trong đó.

# ĐỀ XUẤT THUẬT TOÁN

## 3.1 Các phương pháp đã áp dụng

**Phương pháp truyền thống**: Trước đây, lĩnh vực nhận dạng người nói đã bị kiểm soát bởi các I-Vectors [5], được phân loại bằng các kỹ thuật như heavy-tailed PLDA [6] và GaussPLDA [7]. Trong lúc tìm hiểu các công nghệ tiên tiến trong một thời gian dài, các phương pháp như vậy bị bất lợi bởi sự phụ thuộc của chúng vào kỹ thuật thủ công. Một đánh giá chuyên sâu về các phương pháp truyền thống này được đưa ra trong [8].

**Phương pháp Deep Learning**: Thành công của Deep Learning trong lĩnh vực computer vision và nhận dạng giọng nói đã thúc đẩy việc sử dụng mạng nơ ron sâu (deep neural networks-DNN) như các trình trích xuất features kết hợp với phân loại tuy không được huấn luyện end-to-end. Mặc dù các phương pháp kết hợp như vậy đem lại hiệu quả cao, chúng vẫn đòi hỏi nhiều kỹ thuật thủ công. Ngược lại, kiến trúc CNN có thể được áp dụng trực tiếp vào các raw spectrogram và được huấn luyện theo phương pháp end-to-end. Ví dụ: sử dụng Siamese feedforward DNN để so sánh phân biệt hai giọng nói, tuy nhiên điều này phụ thuộc vào các feature MFCC được tính toán trước, trong khi vẫn học các feature thay vì sử dụng MFCC. Liên quan nhất đến công việc của chúng ta là [9], họ đào tạo một hệ thống neural embedding bằng cách sử dụng triplet loss. Tuy nhiên, họ lại sử dụng các bộ dữ liệu nội bộ riêng cho cả training và testing, do đó không thể so sánh trực tiếp với bài toán của họ.

**Dữ liệu**. Các bộ dữ liệu nhận dạng người nói hiện tại thường gặp phải một hoặc nhiều hạn chế sau: (i) chúng được lấy trong các điều kiện được kiểm soát (ví dụ: từ các cuộc gọi điện thoại hoặc trong phòng thí nghiệm âm thanh), (ii ) chúng được chú thích thủ công do đó bị giới hạn về kích thước hoặc (iii) không có sẵn miễn phí cộng đồng người nói (xem [1] để xem xét đầy đủ các bộ dữ liệu hiện có). Tuy nhiên, bộ dữ liệu VoxCeleb2 không gặp phải bất kỳ hạn chế nào trong số này.

## 3.2 Bộ dữ liệu VoxCeleb2:

VoxCeleb2 chứa hơn 1 triệu cách nói cho hơn 6.000 người nổi tiếng, được trích xuất từ ​​các video được tải lên YouTube. Bộ dữ liệu khá cân bằng giới tính, với 61% người nói là nam. Các diễn giả trải rộng trên một loạt các sắc tộc, giọng nói, ngành nghề và lứa tuổi khác nhau. Số lượng lớn video trong bộ dữ liệu được quay trong điều kiện có nhiều tiếng ồn. Chúng bao gồm các cuộc phỏng vấn từ thảm đỏ, sân vận động ngoài trời và các studio, các bài phát biểu dành cho khán giả lớn, trích đoạn từ đa phương tiện được quay chuyên nghiệp và thậm chí cả video thô được quay trên các thiết bị cầm tay. Các phân đoạn âm thanh có trong bộ dữ liệu bị suy giảm với tiếng nhiễu như trò chuyện xung quanh, tiếng cười, lời nói chồng chéo và âm thanh phòng khác nhau. Bộ dữ liệu cũng cung cấp tính năng nhận diện khuôn mặt và theo dõi khuôn mặt cho các diễn giả trong bộ dữ liệu và hình ảnh khuôn mặt tương tự một cách tự nhiên, với các biến thể trong (bao gồm cả cấu hình) như ánh sáng, chất lượng hình ảnh và độ mờ chuyển động. Bộ dữ liệu bao gồm cả development (training/validation) và test set. Tuy nhiên, vì sử dụng bộ dữ liệu VoxCeleb1 cho testing, nên chỉ bộ development set được sử dụng cho tác vụ nhận dạng người nói. Bộ test set VoxCeleb2 sẽ chứng minh được sự hữu ích cho các ứng dụng khác trong audio-visual learning khi nó có thể được sử dụng. Bộ development set của VoxCeleb2 không có sự trùng lặp với các danh tính trong bộ dữ liệu VoxCeleb1 hoặc bộ SITW.

## 3.3 Xây dựng mô đun đọc đặc trưng âm thanh

Trong phần này, chúng tôi mô tả hệ thống neural embedding, được gọi là VGGVox. Hệ thống được huấn luyện bởi các spectrogram cường độ ngắn hạn được trích xuất trực tiếp từ các phân đoạn âm thanh raw, không có quá trình tiền xử lý nào khác. Một kiến trúc trunk deep neural network được sử dụng để trích xuất các đặc trưng ở mức frame, sau đó được gộp lại để thu được các speaker embeddings ở mức độ phát âm. Toàn bộ mô hình sau đó được huấn luyện bằng cách sử dụng contrastive loss. Pre-training sử dụng lớp softmax và cross-entropy trên một danh sách người nói cố định giúp cải thiện hiệu suất mô hình; do đó chúng tôi pre-train mô hình kiến trúc trunk cho nhiệm vụ identification đầu tiên.

### 3.3.1 Evaluation

Mô hình được huấn luyện trên bộ dữ liệu VoxCeleb2. Tại thời điểm training, các cặp được lấy mẫu trực tuyến bằng phương pháp được mô tả trong phần 4.3. Testing được thực hiện trên bộ dữ liệu VoxCeleb1, với các mẫu testing được cung cấp trong bộ dữ liệu đó. Chương trình báo cáo hai số liệu hiệu suất: (i) Equal Error Rate (EER) là tỷ lệ mà cả hai lỗi accept và reject đều bằng nhau; và (ii) cost function:

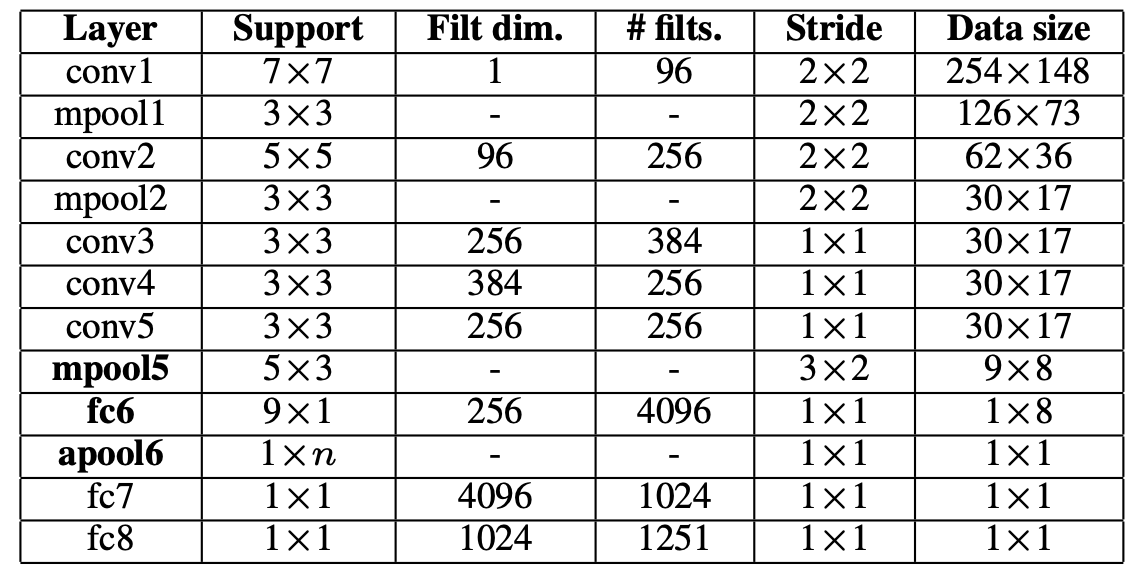
Trong đó chúng ta giả sử prior target probability Ptar là 0,01 và trọng số equal weights bằng 1.0 giữa “misses” Cmiss và “false alarm” Cfa. Cả hai số liệu thường được sử dụng để đánh giá hệ thống xác minh danh tính.

\*misses là thông số khi có tín hiệu nhưng hệ thống lại không phát hiện ra

\*false alarm là khi không có tín hiệu nhưng hệ thống lại báo là có

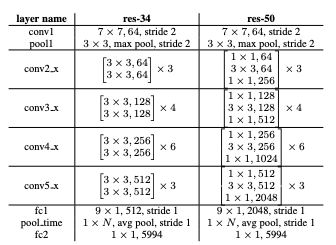
### 3.3.2 Kiến trúc trunk và ResNet

VGG-M: Kiến trúc cơ sở trunk là CNN được giới thiệu trong [1]. Kiến trúc này được sửa đổi từ VGGM CNN, được biết đến với hiệu quả cao và hiệu suất phân loại tốt trên dữ liệu hình ảnh. Cụ thể, lớp fully connected layer fc6 từ VGG-M ban đầu được thay thế bằng hai lớp - lớp fully connected layer 9 × 1 (hỗ trợ trong miền tần số) và lớp average pool layer có hỗ trợ 1 × n, trong đó n phụ thuộc vào độ dài của đoạn nói đầu vào (ví dụ với đoạn 3 giây, n = 8). Lợi ích của việc sửa đổi này là mạng trở nên bất biến với temporal position nhưng không phải là dạng frequency, điều này cần cho speech, nhưng lại không cho images. Điều này cũng làm giảm số lượng tham số từ 319M trong VGG-M xuống 67M trong mạng này, giúp tránh tình trạng overfitting. Kiến trúc CNN hoàn chỉnh được chỉ định trong Bảng 4.



Bảng 4: Kiến trúc CNN. Kích thước dữ liệu lên đến fc6 dành cho đoạn đầu vào 3 giây, nhưng mạng có thể chấp nhận đầu vào có độ dài thay đổi.

ResNets: Kiến trúc mạng Residual-network (ResNet) [10] tương tự như standard multi-layer CNN, nhưng thêm vào các kết nối tắt để các lớp thêm phần residual vào ánh xạ định danh trên đầu ra kênh. Thử nghiệm với cả kiến trúc ResNet34 và ResNet-50 và sửa đổi các lớp để thích ứng với spectrogram đầu vào. Áp dụng batch normalization trước khi tính toán Rectified linear unit (ReLU). Các kiến trúc được chỉ định trong Bảng 3.



Bảng 3: Kiến trúc Res-34 và Res-50 đã sửa đổi với lớp average pool ở cuối. Các lớp ReLU và batchnorm không được hiển thị. Mỗi hàng chỉ định số lượng bộ lọc convolutional và kích thước của chúng là kích size × size, # filters.

## 3.4 Nghiên cứu và xây dựng mô đun tuỳ chỉnh tham số huấn luyện cho hệ thống định danh người nói

### 3.4.1 Training Loss strategies

Mô hình sử dụng contrastive loss trên các paired embedding, tìm cách giảm thiểu khoảng cách giữa các cặp embeddings positive và penalise khoảng cách cặp negative là nhỏ hơn margin parameter α. Pair-wise loss tương tự như constrative loss, rất khó train, và do đó để tránh hàm tiệm cận tới cực tiểu sớm trong đào tạo, chương trình tiến hành theo hai giai đoạn: đầu tiên, pre-training cho identification sử dụng softmax loss, thứ hai, tinh chỉnh với contrastive loss

Pre-training cho identification: Chiến lược đầu tiên là sử dụng softmax pre-training để khởi tạo các trọng số của mạng. cross entropy loss tạo ra độ hội tụ ổn định hơn so với contrastive loss, có thể do đào tạo softmax không bị ảnh hưởng bởi độ khó của các cặp khi sử dụng mất tương phản. Để đánh giá hiệu suất nhận dạng, chúng tôi tạo một bài kiểm tra xác thực được tổ chức bao gồm tất cả các phân đoạn lời nói từ một video cho mỗi danh tính.

Học một embedding với contrastive loss – hard negative mining: Lấy mô hình được pre-trained về nhiệm vụ identification và thay thế lớp 5994-way classification layer bằng một lớp fully connected layer kích thước đầu ra 512. Mạng này được train với contrastive loss

Khó khan chính liên quan đến việc học embedding thông qua contrastive loss là khi tập dữ liệu càng lớn, số lượng các cặp possible tăng lên theo phương trình bậc hai. Trong trường hợp như vậy, mạng nhanh chóng học cách ánh xạ chính xác các ví dụ dễ dàng và hard negative mining thường được yêu cầu để cải thiện hiệu suất để cung cấp cho mạng một learning signal tốt hơn. Sử dụng một chiến lược offline hard negative mining, cho phép chọn các harder negative (ví dụ: top 1 phần trăm các cặp được tạo ngẫu nhiên) so với phương pháp online (in-batch) hard negative mining phương pháp khai thác bị giới hạn bởi batch size. Chương trình không khai thác positives, vì các cặp false positive xuất hiện nhiều hơn các cặp false negative trong một mẫu ngẫu nhiên (do possible label noise có thể xảy ra khi xác minh) và các error label này sẽ dẫn đến động việc học của model có chất lượng kém hơn.

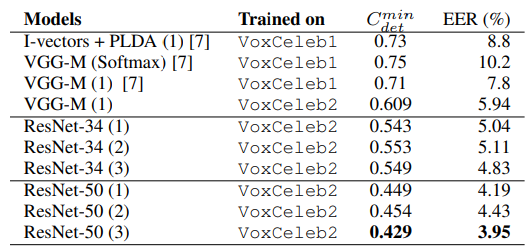
### 3.4.2 Test time augmentation

Sử dụng average pooling tại thời điểm test bằng cách đánh giá toàn bộ cách âm test cùng một lúc bằng cách thay đổi kích thước của lớp apool6. Ở đây, chương trình được thử nghiệm với các augmentation protocols khác nhau để đánh giá hiệu suất tại thời điểm test. Chúng tôi đề xuất ba phương pháp:

(1) Baseline: variable average pooling như được mô tả trong [1];

(2) Mười mẫu 3 giây được cắt từ mỗi segmen test và sử dụng giá trị trung bình của các feature

(3) Mười mẫu 3 giây được cắt từ mỗi segmen test, tính khoảng cách giữa mỗi possible pair các phần cắt (10 × 10 = 100) từ hai speech segments và sử dụng giá trị trung bình của 100 distances. Phương pháp này dẫn đến sự cải thiện marginal trong hiệu suất, như được hiển thị trong Bảng 4



Bảng 4: Kết quả xác minh trên bộ VoxCeleb1 test set ban đầu (thấp hơn là tốt hơn). Số trong ngoặc liên quan đến các phương pháp test time augmentation

### 3.4.3 Chi tiết thực hiện

**Input features.** Spectrograms được tính toán từ âm thanh raw theo kiểu sliding window bằng cách sử dụng một hamming window có chiều rộng 25ms và bước 10ms. Điều này tạo nên các spectrograms có kích thước 512 x 300 trong 3 giây nói. Mean normalisation và variance normalisation được thực hiện trên frequency bin của spectrum.

**Training.** Trong quá trình training, ta lấy mẫu ngẫu nhiên các segment 3 giây từ mỗi cách nói. Việc triển khai dựa trên toolbox deep learning MatConvNet [11]. Mỗi mạng được đào tạo trên ba GPU Titan X trong 30 epoch hoặc cho đến khi validation error ngừng giảm, tùy theo thời gian sớm hơn, sử dụng batch-size 64. Sử dụng SGD với momentum (0,9), giảm weight (5E - 4) và logarithmically decaying learning rate (bắt đầu từ 10−2 và phân rã thành 10−8).

## 3.5 Nghiên cứu và xây dựng mô đun định danh người nói.

Quy trình thực hiện xác minh: Đầu tiên, chương trình sẽ lấy mỗi file trong bộ enroll samples data, đây là file ghi âm đã xác minh giọng nói của từng người, chuyển thành hình ảnh spectrogram, sau đó đưa vào model VGGVox để lấy ra các vector giọng đặc trưng của từng người, tương tự, chương trình cũng sẽ lấy ra các vector giọng đặc trưng trong các file test, tiến hành so sánh với mỗi vector giọng đặc trưng của những người đã ghi danh, dựa theo cosine/euclidean distance, tìm ra minimum, từ đó xác minh được người nói của đoạn file test.

Tương tự như quá trình xử lý đầu vào của training, thực hiện modun định danh người nói cần tiền xử lý như sau:

Tất cả âm thanh trước tiên được chuyển đổi thành các luồng 16 bit đơn kênh (mono/single-channel) với tốc độ lấy mẫu 16kHz để thống nhất. Các biểu đồ spectrogram sau đó được tạo ra theo kiểu sliding window bằng cách sử dụng hamming window có chiều rộng 25ms và bước 10ms. Điều này mang lại cho các spectrogram đồ có kích thước 512 x 300 trong 3 giây nói. Chuẩn hóa trung bình và phương sai được thực hiện trên mọi ngăn tần số của spectrogram. Việc chuẩn hóa này là rất quan trọng, dẫn đến độ chính xác phân loại tăng gần 10%, như trong Bảng 5. Không có quá trình tiền xử lý cụ thể nào khác (ví dụ: loại bỏ các đoạn im lặng, phát hiện hoạt động giọng nói hoặc loại bỏ câu nói không được nói). Những quang spectrogram cường độ thời gian ngắn này sau đó được sử dụng làm đầu vào cho CNN.

Quy trình triển khai chế độ offline:

### 3.5.1 Chuẩn bị bộ dataset cho việc đánh giá nhận diện người nói

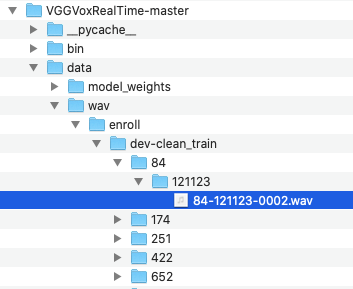
Bộ data để thực hiện đánh giá gồm 2 phần:

-Phần xác minh, sẽ bao gồm file ghi âm giọng nói của một người ghi kèm tên người đó, phần dataset này sẽ là mục đích để cho máy nhận biết giọng này là của ai. Các file đó sẽ lưu trong đường dẫn:

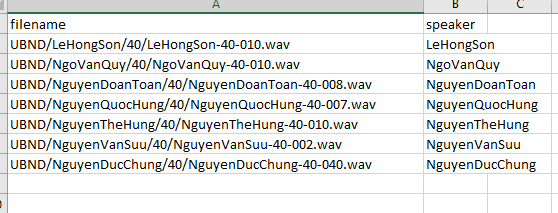
VGGVoxRealTime-master/data/wav/enroll/“tên\_bộ\_dataset”/“tên\_người\_nói”/“file\_ghi\_âm.wav”

Lưu ý phần đường dẫn rất quan trọng và mình cần phải làm một list file excel có 2 cột, 1 cột đầu tiên là “filename” chứa đoạn đường dẫn “tên\_bộ\_dataset”/“tên\_người\_nói”/“file\_ghi\_âm.wav”

và một cột “speaker chứa tên người nói đó. Ví dụ:



Các file ghi âm xác minh theo bộ dataset này thì chỉ có một file cho mỗi người và độ dài khoảng 20-60s. Lần lượt như vậy ta sẽ có một bộ dataset để xác minh giọng của người nói và một file list excel như thế này.



File excel này cần được đặt tên chính xác là “batch\_enroll\_list” và được lưu trong đường dẫn:

VGGVoxRealTime-master/lst

-Phần đánh giá: Đây là bộ dataset ta cần đưa vào để xác minh ai là người nói trong các đoạn ghi âm đấy. Bộ dataset này thực hiện giống y hệt ở trên, tuy nhiên ở phần đường dẫn sẽ cần sửa lại một chút:

VGGVoxRealTime-master/data/wav/test\_clean/“tên\_bộ\_dataset”/“tên\_người\_nói”/“file\_ghi\_âm.wav”

Ta sẽ thay folder “enroll” bằng “test\_clean” so với bộ dataset trên

Ở đây, phần “tên\_người\_nói” sẽ vẫn được đưa vào để kiểm tra kết quả xem có đúng là người này nói hay không, tuy nhiên, chương trình vẫn sẽ đánh giá tất cả những người nói đã ghi danh ở phần data trên, từ đó cho ra kết quả ai là người nói trong file đấy, so sánh với “tên\_người\_nói” ta đã đưa vào và cho ra kết quả trong file excel là 0 hay 1 ở cột “correct”. Em nghĩ vấn đề này sẽ dùng để đánh giá độ chính xác của thuật toán trong thử nghiệm và từ đó quyết định sử dụng cho chế độ online.

Cũng tương tự vậy, ta sẽ tạo một file excel có 2 cột, cột “filename” lưu đoạn đường dẫn

/“tên\_bộ\_dataset”/“tên\_người\_nói”/“file\_ghi\_âm.wav”

Và cột “speaker” chứa “tên\_người\_nói”, file excel này cần được lưu tên chính xác là batch\_test\_list và đưa vào

VGGVoxRealTime-master/data/wav/test\_clean/“tên\_bộ\_dataset”/“tên\_người\_nói”/“file\_ghi\_âm.wav”

Như vậy là đã xong phần dataset để tiến hành xác minh.

# KẾT QUẢ

Phương pháp được thử nghiệm trên tập dữ liệu bàn giao của phía bên UBND Hà Nội, các file được đưa vào thử nghiệm trước tiên bao gồm: file của chủ tịch Nguyễn Đức Chung cùng các phó chủ tịch. Quá trình tiền xử lý bao gồm: Chuyển đổi định dạng mp3 về wav-16bit float, giảm tần số từ 44100Hz và 32000Hz xuống cùng 16000Hz, chuyển đổi dạng audio từ stereo về mono (single-channel). Các file sau đó được cắt nhỏ thành nhiều đoạn nhỏ với các mức thời gian tạo thành các bộ enroll set và test set khác nhau, các mức thời gian bao gồm: 20 giây, 40 giây và 60 giây. Toàn bộ file chia nhỏ được lấy hết tạo thành tập test set, các file enroll do chỉ lấy một file nên được lấy ngẫu nhiên từ nhiều file nhỏ đã cắt với từng mức thời gian tương ứng.

Dưới đây là bảng kết quả thử nghiệm

|  |  |  |  |
| --- | --- | --- | --- |
| ENROLL | TEST | Số file xác minh đúng | Tỷ lệ |
| 20 giây | 20 giây | 261/281 | 92.88% |
| 20 giây | 40 giây | 129/142 | 90.84% |
| 20 giây | 60 giây | 89/95 | 93.68% |
| 40 giây | 20 giây | 259/281 | 92.17% |
| 40 giây | 40 giây | 132/142 | 92.96% |
| 40 giây | 60 giây | 90/95 | 94.73% |
| 60 giây | 20 giây | 260/281 | 92.52% |
| 60 giây | 40 giây | 132/142 | 92.96% |
| 60 giây | 60 giây | 88/95 | 92.63% |

# TÀI LIỆU THAM KHẢO

[1] A. Nagrani, J. S. Chung, and A. Zisserman, “VoxCeleb: a largescale speaker identification dataset,” in INTERSPEECH, 2017.

[2] M. McLaren, L. Ferrer, D. Castan, and A. Lawson, “The speakers in the wild (SITW) speaker recognition database,” in INTERSPEECH, 2016.

[3] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, “Return of the devil in the details: Delving deep into convolutional nets,” in Proc. BMVC., 2014.

[4] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” arXiv preprint arXiv:1512.03385, 2015.

[5] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speaker verification,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 19, no. 4, pp. 788–798, 2011.

[6] P. Matejka, O. Glembek, F. Castaldo, M. J. Alam, O. Plchot, ˇP. Kenny, L. Burget, and J. Cernocky, “Full-covariance ubm and ˇheavy-tailed plda in i-vector speaker verification,” in Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on. IEEE, 2011, pp. 4828–4831.

[7] S. Cumani, O. Plchot, and P. Laface, “Probabilistic linear discriminant analysis of i-vector posterior distributions,” in Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013, pp. 7644–7648.

[8] J. H. Hansen and T. Hasan, “Speaker recognition by machines and humans: A tutorial review,” IEEE Signal processing magazine, vol. 32, no. 6, pp. 74–99, 2015.

[9] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, “Deep speaker: an end-to-end neural speaker embedding system,” arXiv preprint arXiv:1705.02304, 2017.

[10] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, “A novel scheme for speaker recognition using a phonetically-aware deep neural network,” in Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. IEEE, 2014, pp. 1695– 1699.

[11] A. Vedaldi and K. Lenc, “Matconvnet – convolutional neural networks for matlab,” CoRR, vol. abs/1412.4564, 2014.