

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
**KHOA CÔNG NGHỆ THÔNG TIN 1**

---



**BÀI TẬP LỚN**

**MÔN HỌC: HỆ CƠ SỞ DỮ LIỆU ĐA PHƯƠNG TIỆN**

**ĐỀ TÀI: XÂY DỰNG HỆ CSDL NHẬN DẠNG NGÔN NGỮ  
TIẾNG ANH BẰNG GIỌNG NÓI**

**Giảng viên: NGUYỄN ĐÌNH HÓA**

**Nhóm môn học: D18-026**

**Danh sách thành viên nhóm 21:**

- |                   |              |
|-------------------|--------------|
| 1. Cù Minh Tiến   | - B18DCCN528 |
| 2. Đỗ Lan Anh     | - B18DCCN011 |
| 3. Bùi Xuân Thuận | - B18DCCN649 |

*Hà Nội, 06/2022*

## Mục Lục

<b>I. Tổng quan về bộ dữ liệu .....</b>	<b>2</b>
<b>II. Kiến thức chuyên môn để trích rút đặc trưng và nhận dạng từ phát âm.....</b>	<b>3</b>
1) Trích rút đặc trưng dựa trên kĩ thuật MFCC( Mel Frequency Cepstral Coefficients) .....	3
2) Tìm hiểu các kĩ thuật nhận dạng từ phát âm .....	7
a) Nhận dạng từ bằng mô hình Markov ẩn ( Hiden Markov Model).....	7
b) Nhận dạng dựa trên Khoảng cách Euclide.....	8
<b>III. Lưu trữ các thuộc tính âm thanh.....</b>	<b>9</b>
<b>IV. Xây dựng hệ thống .....</b>	<b>11</b>
<b>V. Demo hệ thống và đánh giá kết quả đạt được.....</b>	<b>13</b>
<b>VI. Kết luận .....</b>	<b>16</b>
<b>Tài liệu tham khảo.....</b>	<b>17</b>

## I. Tổng quan về bộ dữ liệu

Dữ liệu lấy từ tập dữ liệu trong cuộc thi nhận dạng giọng nói do TensorFlow tổ chức [https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/data?fbclid=IwAR1n50rSoaP3ULkfDI4w18CVa8ytTs-rdlCsXtjDF\\_1Jul4gsVABYdJrYnA](https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/data?fbclid=IwAR1n50rSoaP3ULkfDI4w18CVa8ytTs-rdlCsXtjDF_1Jul4gsVABYdJrYnA)











Các file âm thanh có định dạng .wav , có độ dài 1 s và dung lượng 32 kb. Các từ nhận dạng bao gồm one,two, five, six, nine, yes, no, on, go, tree, mỗi từ bao gồm 12 file âm thanh, tổng cộng có 120 file âm thanh. Lấy 10 file âm thanh mỗi loại từ để làm dữ liệu huấn luyện, 2 file mỗi loại để tiến hành kiểm tra..Các file âm thanh để làm dữ liệu huấn luyện lưu trong tệp data\_train, còn những file dùng để kiểm tra, chạy thử lưu trong data\_set.

data_set	6/18/2022 7:04 PM	File folder
data_train	6/18/2022 7:04 PM	File folder

Cả trong data\_set và data\_train đều có 10 tệp có tên là nhãn của 10 từ để nhận dạng

Name	Date modified	Type	Size
five	6/18/2022 7:04 PM	File folder	
go	6/18/2022 7:04 PM	File folder	
nine	6/18/2022 7:04 PM	File folder	
no	6/18/2022 7:04 PM	File folder	
on	6/18/2022 7:04 PM	File folder	
one	6/18/2022 7:04 PM	File folder	
six	6/18/2022 7:04 PM	File folder	
tree	6/18/2022 7:04 PM	File folder	
two	6/18/2022 7:04 PM	File folder	
yes	6/18/2022 7:04 PM	File folder	

10 Tệp trên trong data\_train, mỗi tệp đều có 10 file .wav . Còn 10 tệp trong data\_set thì mỗi tệp có 2 file .wav. Dưới là 10 file âm thanh nằm trong thư mục five của data\_train:

Name	#	Title
 08ab8082_nohash_0		
 08ab8082_nohash_1		
 8a1c449e_nohash_0		
 8a5acefd_nohash_0		
 8a5acefd_nohash_1		
 8a56f36e_nohash_0		
 8a56f36e_nohash_1		
 8a28231e_nohash_0		
 8eb4a1bf_nohash_3		
 8eb4a1bf_nohash_4		

Mở file âm thanh đầu tiên :



## II. Kiến thức chuyên môn để trích rút đặc trưng và nhận dạng từ phát âm

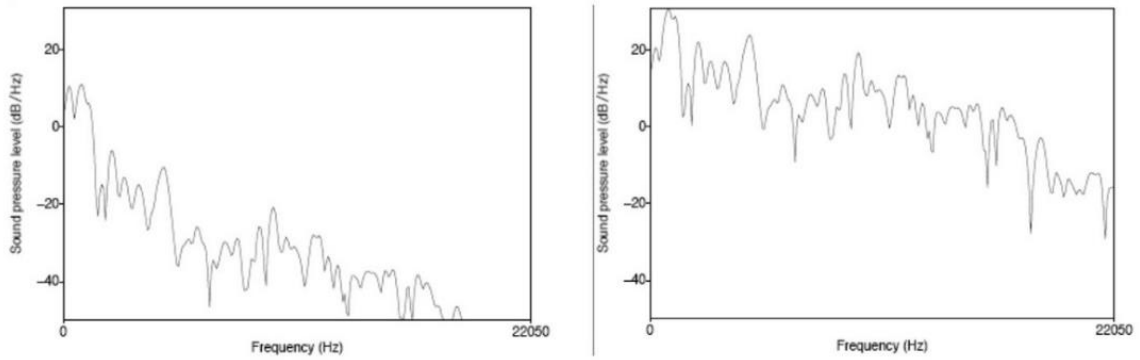
### 1) Trích rút đặc trưng dựa trên kĩ thuật MFCC( Mel Frequency Cepstral Coefficients)

Kỹ thuật trích xuất thuộc tính của MFCC về cơ bản bao gồm cắt chuỗi tín hiệu âm thanh thành các đoạn có độ dài ngắn bằng nhau =25 ms và overlap lên nhau 10 s, áp dụng DFT, lấy nhật ký độ lớn, sau đó làm cong các tần số trên thang Mel , tiếp theo là áp dụng DCT nghịch đảo. Mô tả chi tiết về các bước khác nhau liên quan đến các bước trong MFCC được giải thích như sau:

**Bước 1:** Pre-emphasis( Tiền nhấn mạnh) : Mục đích của nó là để cân bằng phổ của âm thanh có độ dốc lớn trong vùng tần số cao, cải thiện độ chính xác của việc phát hiện ngữ âm và tránh các vấn đề về số trong hoạt động biến đổi Fourier .

Tiền nhấn mạnh sử dụng bộ lọc để tăng tần số cao hơn. Bộ lọc thường được sử dụng là:

$$x'[t_d] = x[t_d] - \alpha x[t_d - 1]$$



Giá trị của  $\alpha$  thường nằm trong khoảng từ 0.95 đến 0.99. Ở hệ thống của chúng em sử dụng  $\alpha = 0.95$

**Bước 2:** : Flame-blocking and windowing (Phân đã tín hiệu thành các khung và mở cửa sổ) : Tín hiệu lời nói là tín hiệu thay đổi chậm theo thời gian hoặc gần như đứng yên. Để có các đặc tính âm thanh ổn định, giọng nói cần được kiểm tra trong một khoảng thời gian đủ ngắn. Các phép đo quang phổ ngắn được thực hiện trên cửa sổ 25 ms và chồng lẫn nhau 15 ms.

Việc chồng lẫn nhau 15ms cho phép theo dõi các đặc điểm thời gian của từng âm thanh giọng nói và cửa sổ phân tích 25ms thường đủ để cung cấp độ phân giải tốt của những âm thanh này, đồng thời đủ ngắn để giải quyết các đặc điểm thời gian quan trọng.

Tuy nhiên, việc cắt frame sẽ làm các giá trị ở 2 biên của frame bị giảm đột ngột (về giá trị 0), sẽ dẫn tới hiện tượng: khi DFT sang miền tần số sẽ có rất nhiều nhiễu ở tần số cao. Để khắc phục điều này, ta cần làm mượt bằng cách nhân chập frame với 1 vài loại cửa sổ. Trên mỗi khung, một cửa sổ được áp dụng để giảm tín hiệu về phía ranh giới của khung. Nói chung, cửa sổ Hanning hoặc Hamming được sử dụng.

$$w[n] = (1 - \alpha) - \alpha \cos\left(\frac{2\pi n}{L-1}\right)$$

Hamming ( $\alpha = 0.46164$ ) , Hanning  $\alpha = 0.5$ , L là độ rộng của cửa sổ.

Ta được phổ mới :  $X[n] = x'[n] \times w[n]$

**Bước 3:** Biến đổi Fourier rời rạc DFT spectrum (phổ DFT) : mỗi khung cửa sổ được chuyển đổi thành phổ cường độ bằng cách áp dụng DFT

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

Trong đó N là số điểm được sử dụng để tính DFT

**Bước 4:** Mel spectrum (phổ Mel) : Phổ Mel được tính bằng cách truyền tín hiệu đã biến đổi Fourier qua một tập hợp các bộ lọc thông dải được gọi là ngân hàng bộ lọc Mel. Mel là một đơn vị đo lường dựa trên tần số cảm nhận của tai người. Nó không tương ứng tuyến tính với tần số vật lý của âm thanh, vì hệ thống thính giác của con người dường như không cảm nhận được cao độ một cách tuyến tính. Thang đo Mel là khoảng cách tần số tuyến tính dưới 1 kHz và khoảng cách logarit trên 1kHz  
Tính gần đúng của Mel từ tần số vật lý được biểu thị bằng :

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

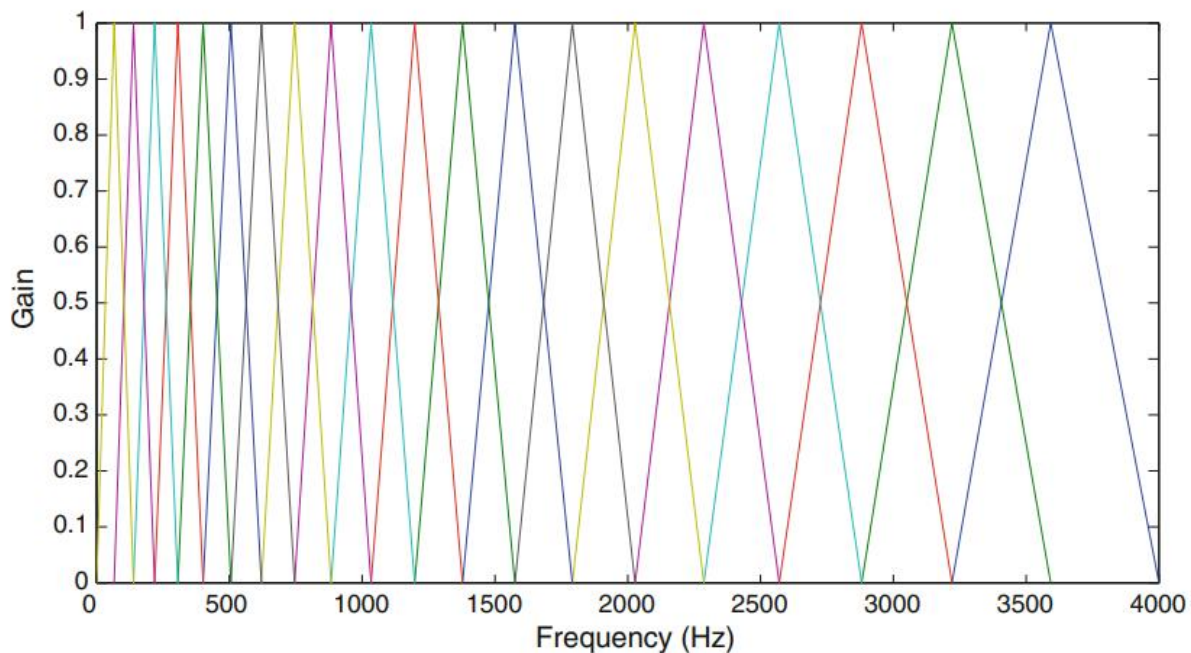
Trong đó f biểu thị tần số vật lý tính bằng Hz và  $f_{Mel}$  biểu thị tần số cảm nhận.

Ngân hàng bộ lọc có thể được thực hiện trong cả miền thời gian và tần số. Đối với tính toán MFCC, các ngân hàng bộ lọc thường được triển khai trong miền tần số. Các tần số trung tâm của các bộ lọc thường cách đều nhau trên trục tần số. Tuy nhiên, để bắt chước nhận thức của tai người, trục bị cong vênh, theo hàm phi tuyến được đưa ra trong phương trình  $f_{Mel}$  ở trên . Bộ định hình lọc được sử dụng phổ biến nhất là hình tam giác và trong một số trường hợp, bộ lọc Hanning có thể được sử dụng.

Phổ Mel của quang phổ cường độ  $X(k)$  được tính bằng cách nhân phổ cường độ với từng bộ lọc trọng số Mel tam giác:

$$s(m) = \sum_{k=0}^{N-1} [|X(k)|^2 H_m(k)]; \quad 0 \leq m \leq M - 1$$

Trong đó M là tổng số bộ lọc trọng số Mel. (k) là trọng số được cấp cho thùng phổ năng lượng thứ k đóng góp vào dải đầu ra thứ m và được biểu thị bằng :



Ngân hàng bộ lọc Mel :

$$H_m(k) = \begin{cases} 0, & k < f(m-1) \\ \frac{2(k-f(m-1))}{f(m)-f(m-1)}, & f(m-1) \leq k \leq f(m) \\ \frac{2(f(m+1)-k)}{f(m+1)-f(m)}, & f(m) \leq k \leq f(m+1) \\ 0, & k > f(m+1) \end{cases}$$

Với m nằm trong khoảng từ 0 đến M-1

**Bước 5:** Discrete cosine transform-DCT (Biến đổi cosin rời rạc) : Vì đường âm mượt mà, mức năng lượng trong các dải lân cận có xu hướng tương quan với nhau. DCT được áp dụng cho các hệ số tần số Mel được biến đổi, tạo ra một tập hợp các hệ số cepstral. Trước khi tính toán DCT, phổ Mel thường được biểu diễn trên thang log. Điều này dẫn đến một tín hiệu trong miền cepstral với đỉnh tần số tương ứng đến cao độ của tín hiệu và một số công thức biểu thị các đỉnh tần số thấp. Vì hầu hết thông tin tín hiệu được biểu diễn bằng một vài hệ số MFCC đầu tiên, nên hệ thống có thể trở nên mạnh mẽ bằng cách chỉ trích xuất các hệ số đó bỏ qua hoặc cắt bớt các thành phần DCT bậc cao hơn. Cuối cùng MFCC được tính là :

$$c(n) = \sum_{m=0}^{M-1} \log_{10}(s(m)) \cos\left(\frac{\pi n(m-0.5)}{M}\right); \quad n = 0, 1, 2, \dots, C-1$$

Trong đó c(n) là hệ số cepstral, C là số MFCC. Các hệ thống MFCC chỉ sử dụng 8 – 13 hệ số cepstral. Hệ số thứ 0 được thay bằng tổng năng lượng tại khung đó.

**Bước 6 :** Tính các đặc tính MFCC động : Các hệ số cepstral thường được gọi là các đối tượng tĩnh, vì chúng chỉ chứa thông tin từ một khung nhất định. Thông tin động theo thời gian của tín hiệu thu được bằng cách tính các đạo hàm bậc nhất (delta) và bậc hai (delta-delta) của hệ số cepstral hệ số delta cho biết về tốc độ giọng nói và hệ số delta – delta cung cấp thông tin tương tự như tốc độ của giọng nói. Công thức delta được định nghĩa :

$$\Delta c_m(n) = \frac{\sum_{i=-T}^T i c_m^{(n+i)}}{\sum_{i=-T}^T |i|}$$

Với  $c_m(n)$  biểu thị cho đặc điểm thứ m trong khung thời gian thứ n. T là số khung liên tiếp được sử dụng để tính toán. Thường T được sử dụng là 2. Hệ số delta-delta được tính bằng cách lấy đạo hàm của delta bậc nhất.

Như vậy cuối cùng mỗi khung ta được 39 thuộc tính, bao gồm 13 thuộc tính lấy từ mfcc bao gồm tổng năng lượng và 12 giá trị trích rút bằng phương pháp mfcc, 13 thuộc tính sau là đạo hàm bậc 1, 13 thuộc tính còn lại là đạo hàm bậc 2

## 2) Tìm hiểu các kĩ thuật nhận dạng từ phát âm

### a) Nhận dạng từ bằng mô hình Markov ẩn ( Hidden Markov Model)

Đây là một mô hình thống kê, thành phần của mô hình bao gồm tập N trạng thái  $\{S_i\}$ , Các trạng thái chuyển qua lại với nhau với một xác suất nhất định, tập xác suất di chuyển này được gọi là ma trận dịch chuyển trạng thái  $A = [a_{ij}]$ .

Mô hình hoạt động khi cho chuỗi dữ liệu đầu vào  $O = [o_1, o_2, \dots, o_T]$  gọi là chuỗi quan sát. Đây là dữ liệu trích rút từ tiếng nói cần nhận dạng trong ứng dụng nhận dạng tiếng nói.

Mỗi quan sát  $o_t$  có một xác suất xuất hiện trên mỗi trạng thái  $S_i$ . Tập hợp các trạng thái này gọi là phân phối xác suất của quan sát  $B = \{b_i(o_t)\}$ ,  $1 \leq i \leq n$ .

Ngoài ra còn có tập  $\pi$  là xác suất quan sát đầu tiên  $o_1$  tại trạng thái i

Tập  $\lambda = \{S, A, B, \pi\}$  là các tham số của HMM.

HMM có hai vấn đề chính cần phải giải quyết để nó có thể ứng dụng trong hệ thống nhận dạng:

Vấn đề 1: Nhận dạng. Cho chuỗi quan sát  $O = \{o_1, o_2, \dots, o_T\}$  và một mô hình HMM  $\lambda$ . Tính xác suất  $P(O|\lambda)$  của chuỗi O trên mô hình đó

Vấn đề 2: Huấn luyện. Làm thế nào điều chỉnh các tham số của mô hình  $\lambda$  để  $P(O|\lambda)$  cực đại, nghĩa là tối ưu hóa  $\lambda$ .

Giải quyết vấn đề 1 :Để xác định xác suất chuỗi quan sát O trên một mô hình có sẵn  $\lambda$ , chúng ta dùng thuật toán hướng tới (forward algorithm):

Bước 1 : Khởi tạo:

$$\alpha_t(i) = \pi b_i(o_1), \quad 1 \leq i \leq N$$

Bước 2 : Quy nạp:

$$\alpha_{t+1}(j) = [\sum_{i=1}^N \alpha_t(i) a_{ij}] b_j(o_{t+1}) \quad 1 \leq t \leq T-1, \quad 1 \leq j \leq N$$



Bước 3 : Kết thúc:

$$P(O \setminus \lambda) = \sum_i^N \alpha_T(i)$$

$$\beta_i(T) = 1,$$

$$\beta_i(t) = \sum_{j=1}^N \beta_j(t+1) a_{ij} b_j(y_{t+1}).$$

Với phân phối xác suất của mỗi quan sát  $b_i(o_t)$  được tính theo phân phối chuẩn gauss nhiều chiều :

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\} \quad \mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$$

$$\Sigma = \begin{pmatrix} E[(x_1 - \mu_1)(x_1 - \mu_1)] & E[(x_1 - \mu_1)(x_2 - \mu_2)] & \dots & E[(x_1 - \mu_1)(x_n - \mu_n)] \\ E[(x_2 - \mu_2)(x_1 - \mu_1)] & E[(x_2 - \mu_2)(x_2 - \mu_2)] & \dots & E[(x_2 - \mu_2)(x_n - \mu_n)] \\ \vdots & \vdots & \ddots & \vdots \\ E[(x_n - \mu_n)(x_1 - \mu_1)] & E[(x_n - \mu_n)(x_2 - \mu_2)] & \dots & E[(x_n - \mu_n)(x_n - \mu_n)] \end{pmatrix}$$

E.g. positive  
semi-definite  
 $x \sim \mathcal{N} \left( \begin{pmatrix} 190 \\ 70 \end{pmatrix}, \begin{pmatrix} 100 & 25 \\ 25 & 50 \end{pmatrix} \right)$   
 $\mu$                        $\Sigma$

Với n là số đoạn của file âm thanh,

$\mu$  là *vector* đặc tính đã được trích rút từ đoạn tương ứng

Giải quyết vấn đề 2 :Nội dung của vấn đề 2 là thực hiện quá trình huấn luyện hệ thống để điều chỉnh mô hình  $\lambda$  sao cho đạt được các thông số tối ưu.

$$\overline{a_{ij}} = \frac{\sum_{r=1}^R \sum_{t=1}^{T-1} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{r=1}^R \sum_{t=1}^{T-1} \alpha_t(i) \beta_t(i)} \quad \overline{b_j}(k) = \frac{\sum_{r=1}^R \sum_{t=1}^T \alpha_t(j) \beta_t(j)}{\sum_{r=1}^R \sum_{t=1}^T \alpha_t(j) \beta_t(j)} \quad \begin{matrix} O_t = v_k \end{matrix}$$

Với R là số mẫu tiếng nói để huấn luyện của mỗi từ.

### b) Nhận dạng dựa trên Khoảng cách Euclide

Ta thấy rằng việc triển khai mô hình HMM ở trên là tương đối phức tạp, cho nên có một cách đơn giản hơn để nhận dạng từ tiếng anh từ các vector đặc tính được trích rút ra nhờ phương pháp MFCC được trình bày ở trên đó là tính độ lệch khoảng cách trung bình giữa file âm thanh cần nhận dạng so với các file đã được nhận dạng trước đó.

Mỗi file âm thanh sau quá trình trích rút ta được một ma trận có 99 hàng (tương ứng với 99 khung) mỗi khung có 39 giá trị (39 giá trị mfcc).

Gọi a và b lần lượt là 2 ma trận chứa thuộc tính của 2 file âm thanh tương ứng ta tính khoảng cách euclide giữa 2 file âm thanh như sau :

$$Euclide(a, b) = \sqrt{\sum_{i=1}^{99} \sum_{j=1}^{39} (a_{ij} - b_{ij})^2}$$

Ta lấy tổng khoảng cách Euclide giữa file âm thanh đầu vào, với tất cả các file có nhãn tương ứng. Ví dụ nhãn “Yes” có 10 file âm thanh, ta tính tổng khoảng cách euclide giữa file đầu vào với tất cả 10 file âm thanh có nhãn Yes. Tương ứng tính tổng khoảng cách giữa file đầu vào với các file âm thanh có nhãn “No”, rồi đến nhãn “On”,...

Sau khi tính được tổng khoảng cách giữa file đầu vào với các file trong 10 nhãn tương ứng. ta tìm được nhãn có khoảng cách thấp nhất chính là nhãn của file đầu vào mà ta cần dự đoán.

### III. Lưu trữ các thuộc tính âm thanh

Việc trích rút các đặc trưng từ file .wav mất nhiều thời gian cho nên không thể cứ mỗi lần dự đoán ta lại trích rút trực tiếp từ file .wav được. Chính vì vậy ta cần lưu các đặc trưng ấy thành dạng text vào các file .txt để cho việc sử dụng lại không mất nhiều thời gian, tài nguyên của máy tính.











Tạo tệp data lưu các file .wav trong 2 tệp data\_set và data\_train được trình bày ở trong phần I. Tạo tệp features để chứa các file .txt lưu trữ giá trị đặc trưng của các file âm thanh

Name	Date modified	Type
 data	6/18/2022 7:04 PM	File folder
 features	6/18/2022 7:04 PM	File folder











Trong tệp features chỉ chứa tệp data\_train, bởi vì ta cần lưu các đặc trưng của các file âm thanh huấn luyện. còn các tệp âm thanh .wav trong data\_set dùng để chạy thử.

Name	Date modified	Type
 data_train	6/18/2022 7:04 PM	File folder

Trong tệp data\_train ở features cũng chứa 10 tệp tương ứng với 10 từ cần nhận dạng như trong tệp data\_train của data.

Name	Date modified	Type
 five	6/18/2022 7:04 PM	File folder
 go	6/18/2022 7:04 PM	File folder
 nine	6/18/2022 7:04 PM	File folder
 no	6/18/2022 7:04 PM	File folder
 on	6/18/2022 7:04 PM	File folder
 one	6/18/2022 7:04 PM	File folder
 six	6/18/2022 7:04 PM	File folder
 tree	6/18/2022 7:04 PM	File folder
 two	6/18/2022 7:04 PM	File folder
 yes	6/18/2022 7:04 PM	File folder

Thuộc tính được lưu trong các file có đuôi .txt có tên tương ứng như các file âm thanh .wav đầu vào. Dưới đây là các tệp trong thư mục five

Name	Date modified	Type	Size
 08ab8082_nohash_0.wav	6/18/2022 11:00 PM	Text Document	100 KB
 08ab8082_nohash_1.wav	6/18/2022 11:00 PM	Text Document	100 KB
 8a1c449e_nohash_0.wav	6/18/2022 11:00 PM	Text Document	100 KB
 8a5acefd_nohash_0.wav	6/18/2022 11:00 PM	Text Document	100 KB
 8a5acefd_nohash_1.wav	6/18/2022 11:00 PM	Text Document	100 KB
 8a56f36e_nohash_0.wav	6/18/2022 11:00 PM	Text Document	100 KB
 8a56f36e_nohash_1.wav	6/18/2022 11:00 PM	Text Document	100 KB
 8a28231e_nohash_0.wav	6/18/2022 11:00 PM	Text Document	100 KB
 8eb4a1bf_nohash_3.wav	6/18/2022 11:00 PM	Text Document	100 KB
 8eb4a1bf_nohash_4.wav	6/18/2022 11:00 PM	Text Document	100 KB

Bằng kỹ thuật MFCC, tại mỗi file âm thanh trong tập dữ liệu huấn luyện, ta rút ra được ma trận 99x39 chứa các giá trị đặc trưng gồm 39 thuộc tính tương ứng với mỗi khung trong 99 khung. Dưới đây là tệp chứa ma trận các đặc trưng của file .wav được mở ra trong phần I:

```

File Edit Format View Help
2.751749667395698040e+00, 2.716182118581098148e+00, -1.294822469645046681e+01, 1.625492269533770795e+01, 1.4
69e-01
5.082917285879471336e-01, 2.622301024632978539e+00, -1.009713640370898702e+01, 8.780832403061964442e+00, -9.
018890e-01
-1.669549674527662120e+00, -1.865561777557667167e-01, -6.683050839509102836e+00, 7.894915528801789151e+00, .
352e-02
-2.584369138957889334e+00, -5.007899857889267503e-01, -6.477038286146046708e+00, 5.671779804030758321e+00, .
916895e-01
-4.132118213284591945e+00, -3.087509457400718471e+00, -3.616803154113973129e+00, 3.594790807415650935e+00, .
25489e-01
-4.090045575032287495e+00, -3.663380787958389728e+00, -1.617018166898966047e+00, 2.515456516649080765e+00, .
3751e-01
-3.931935952282813673e+00, -3.395952169454905079e+00, 1.407032272765544434e+00, 2.950164691367149317e+00, -f
02
-3.336881586190795446e+00, 2.343701548192073769e+00, 3.624090931417872419e+00, 3.542811387118149025e+00, -4.
852e-02
-2.558088452850276084e+00, 5.011606701568521061e+00, 3.930652160471663770e+00, 8.776553192940301784e-01, -1.
2e-01
-4.015946038123967554e+00, 6.536474615197841453e-01, 4.252160924317200497e+00, 3.484473299205966601e+00, -2.
1806e-01
-3.690294117201760926e+00, -3.649421642174860425e-01, 2.276611925620388277e+00, 3.783169165004225398e+00, -;
424445e-01

```

#### IV. Xây dựng hệ thống

Các file âm thanh từ vựng nào được lưu trong thư mục có tên tương ứng. Các thư mục được chứa trong 1 tệp là dataset.

**Đầu tiên, ta tiến hành trích rút đặc trưng file âm thanh đầu vào bằng kỹ thuật MFCC:**

1. Đọc các file âm thanh , sau khi đọc tại mỗi file ta được tốc độ lấy mẫu(mẫu/s) phổ theo miền tần số của file âm thanh tương ứng.
2. Tại mỗi file âm thanh, Áp dụng công thức  $x'[t_d] = x[t_d] - \alpha x[t_d - 1]$  để cân bằng phổ của file âm thanh đó.
3. Tiếp tục, ta chia mỗi file âm thanh thành các khung, mỗi khung có độ dài 0.025 s, mỗi khung chồng lên nhau 0.01 s. Áp dụng cửa sổ hamming :

$$w[n] = 0.54 - 0.46 \cos\left(\frac{2\pi n}{L-1}\right) \text{ với } L = 400$$

Ta được phổ mới :

$$X[n] = x'[n] \times w[n]$$

4. Áp dụng DFT:

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

5. Tính:

$$s(m) = \sum_{k=0}^{N-1} [|X(k)|^2 H_m(k)]; \quad 0 \leq m \leq M - 1$$

M = tốc độ lấy mẫu / 2.

6. Biến đổi DCT từ kết quả trên bằng công thức :

$$c(n) = \sum_{m=0}^{M-1} \log_{10}(s(m)) \cos\left(\frac{\pi n(m - 0.5)}{M}\right); \quad n = 0, 1, 2, \dots, C - 1$$

13 giá trị đầu của kết quả trên trong mỗi khung chính là 13 đặc trưng ta cần trích rút tại khung đó

Thay c(0) bằng năng lượng của khung đó :

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

7. Tính đạo hàm bậc nhất và bậc 2 từ 13 thuộc tính được lấy ở trên để thu được 39 thuộc tính cho mỗi khung :

$$\Delta c_m(n) = \frac{\sum_{i=-T}^T i c_m^{(n+i)}}{\sum_{i=-T}^T |i|} \quad c_m(n) \text{ biểu thị cho đặc điểm thứ } m \text{ trong khung}$$

thời gian thứ n, T=2.

**Nhận dạng file âm thanh :**

8. Lấy các ma trận thuộc tính từ các file .txt trong tệp features lưu trữ thuộc tính đã trích rút của các file âm thanh dùng để huấn luyện
9. Tính khoảng cách euclide giữa file đầu vào với lần lượt từng file huấn luyện tương ứng :

$$Euclide(a, b) = \sqrt{\sum_{i=1}^{99} \sum_{j=1}^{39} (a_{ij} - b_{ij})^2}$$

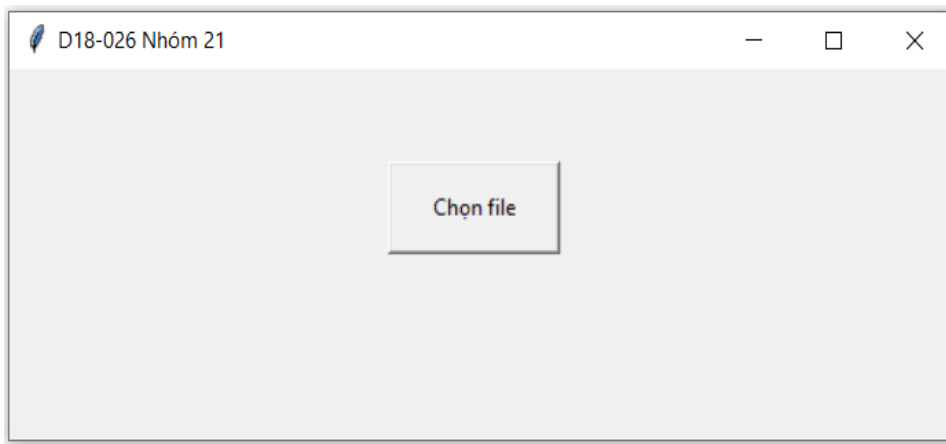
Với a,b lần lượt là ma trận thuộc tính của file đầu vào và mỗi file trong tập huấn luyện,  $a_{ij}$  ,  $b_{ij}$  là giá trị của thuộc tính thứ j tại khung thứ i .

10. Tính giá trị trung bình khoảng cách Euclide giữa file đầu vào với các file có cùng 1 nhãn. Nhãn nào có giá trị trung bình khoảng cách Euclide giữa các file nhỏ nhất thì sẽ là nhãn của file âm thanh đầu vào.
11. Hiển thị kết quả.

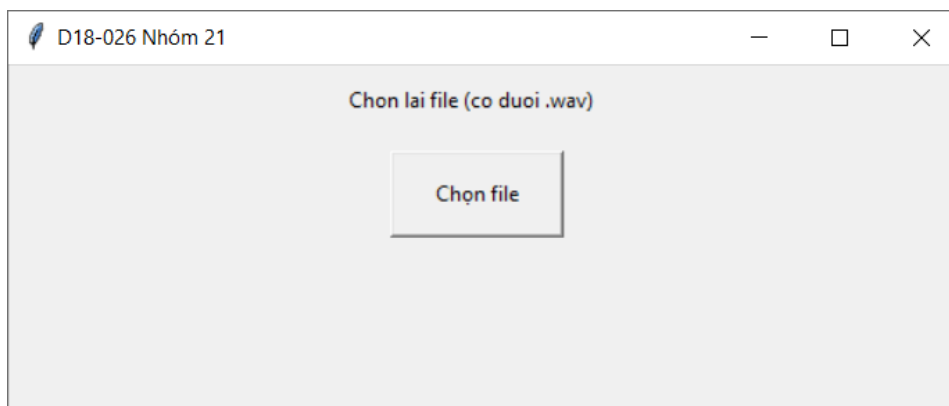
## V. Demo hệ thống và đánh giá kết quả đạt được

Sau khi cài đặt thì tệp demo của chúng em xây dựng có 3 tệp bao gồm :tệp dataset để chứa các file .wav đầu vào và các file .txt lưu các đặc trưng,file ghifile.py để tiến hành trích rút các thuộc tính từ các file .wav đầu vào rồi lưu vào các file .txt và file final.py để chạy giao diện dự đoán .

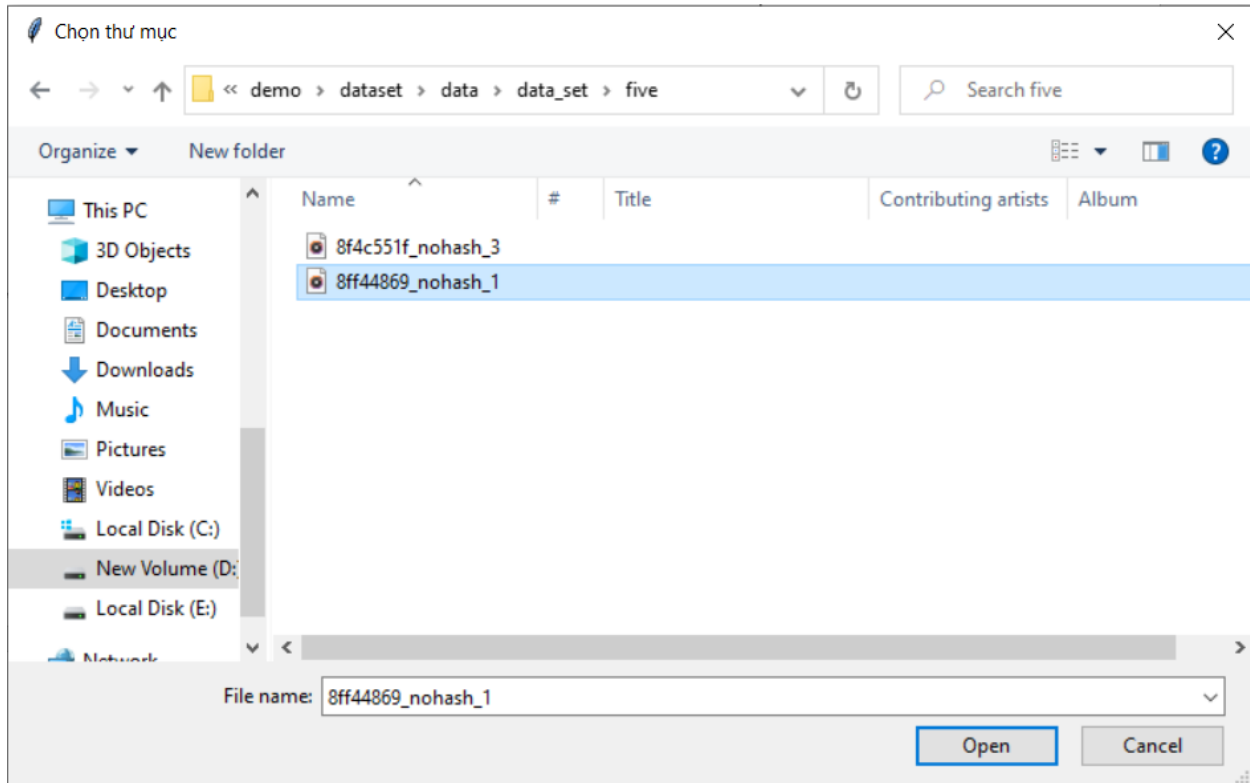
Giao diện hệ thống của chúng em chỉ đơn giản bao gồm 1 button để nhấn chọn file .wav để dự đoán, 1 label để hiển thị đường dẫn file đã chọn hoặc hiện thông báo khi không chọn hoặc không chọn đúng file có đuôi .wav, 1 label còn lại để hiển thị từ tiếng anh dự đoán.



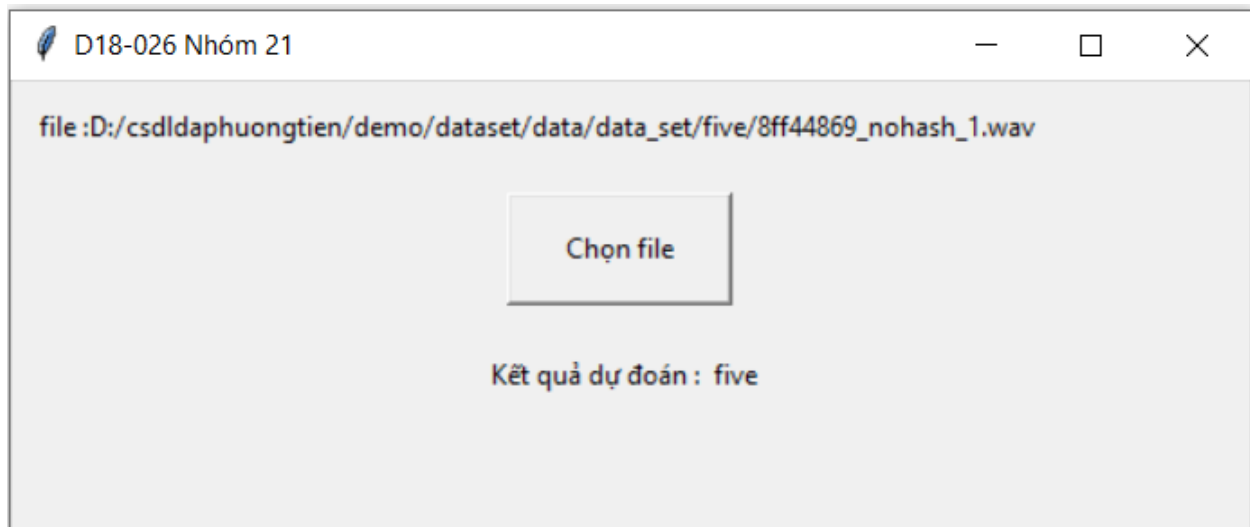
Nếu không chọn file hoặc chọn các file có đuôi khác .wav thì sẽ hiện thông báo:



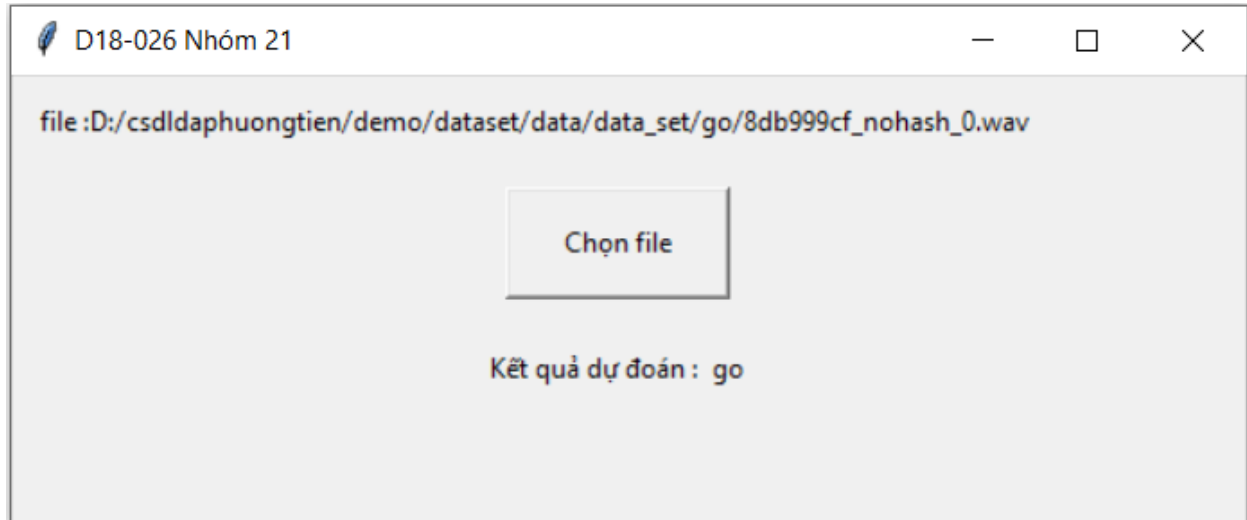
Chọn một tệp âm thanh không nằm trong tập huấn luyện, dưới đây em chọn một tệp âm thanh phát âm từ “five”:



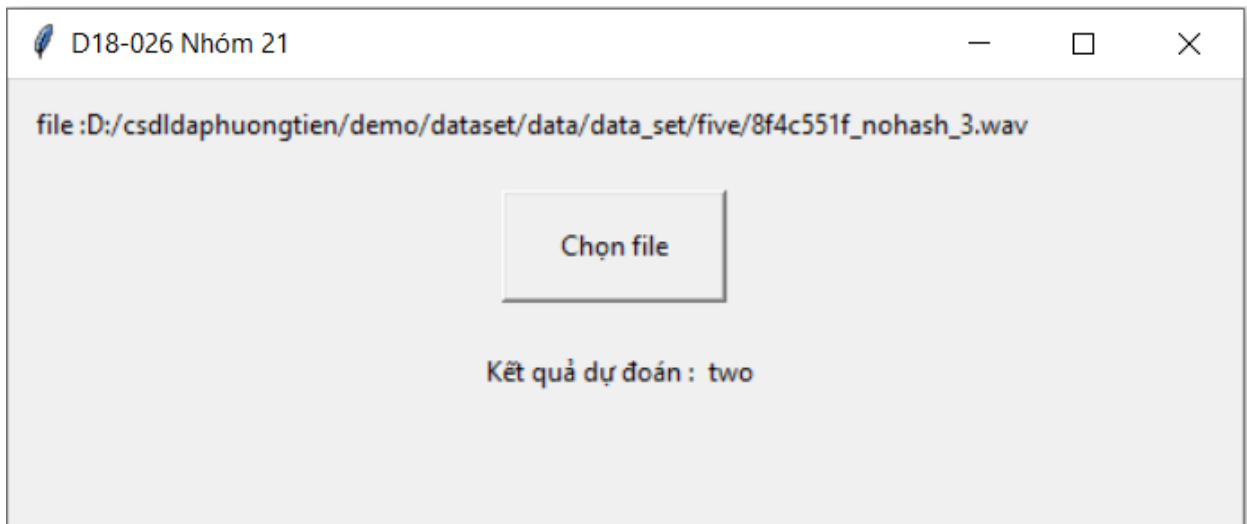
Ta được kết quả dự đoán chính xác :



Thử dự đoán một file phát âm từ “go”, kết quả chính xác:



Tuy nhiên những kết quả dự đoán sai trong hệ thống của chúng em lại khá nhiều. Ví dụ dưới đây là 1 file âm thanh khác so với 2 lần chạy trước phát âm từ “five” nhưng hệ thống lại dự đoán là “two” :





## VI. Kết luận

Dự đoán toàn bộ các file âm thanh trong tập thử nghiệm chúng em được kết quả như sau :

```
five : two
five : five
go : go
go : go
nine : go
nine : two
no : two
no : two
on : on
on : two
one : two
one : go
six : six
six : six
tree : six
tree : two
two : go
two : two
yes : yes
yes : two
du doan dung 8 trong tong so 20 file
Ti le dung 40.0
```

Với bên trái là từ phát âm, bên phải là từ mà hệ thống dự đoán. Hệ thống của chúng em có tỉ lệ dự đoán đúng khá thấp chỉ 40%. Nguyên nhân cho việc dự đoán thấp này do cách nhận dạng từ tiếng anh từ những thuộc tính trích rút còn tương đối đơn giản. Chúng em có thử sử dụng thư viện của sklearn đào tạo model từ thuật toán HMM thì cho ra tỉ lệ tốt hơn là 70% .Như vậy ,chúng em có thể cải thiện hệ thống bằng cách cài đặt những thuật toán nhận dạng cải tiến hơn bằng học máy như HMM, mạng noron nhân tạo thì kết quả sẽ cho ra tốt hơn.

### **Tài liệu tham khảo**

- [1] : <https://www.ijert.org/research/control-system-with-speech-recognition-using-mfcc-and-euclidian-distance-algorithm-IJERTV2IS1384.pdf>
- [2] : 2017\_Bookmatter\_SpeechRecognitionUsingArticula
- [3] : Multimedia\_Database\_Management\_Systems\_(Artech House): Trang 196-222
- [4] : <https://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html>
- [5] : <https://jonathan-hui.medium.com/speech-recognition-acoustic-lexicon-language-model-aacac0462639>