**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**TRẦN TẤN BẢO – 17C 11002**

**TRẦN THÚY HIỀN – 17C 11026**

**NGUYỄN HÀ DUY PHƯƠNG – 17C 11 032**

ĐỒ ÁN MÔN HỌC

XỬ LÝ NGÔN NGỮ NÓI

**BÀI TẬP 2 : NHẬN DẠNG RỜI RẠC**

**GIÁO VIÊN**

**PGS. TS. Vũ Hải Quân**

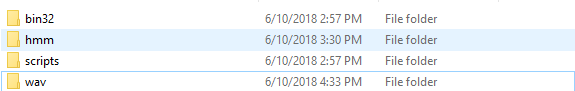
**TP.HCM - 6/2018**

# Mô tả bài toán

* Mục tiêu của bài tập là nhận dạng các từ rời rạc, chẳng hạn nhận dạng chữ số từ không đến chín.
* Kiểm tra sự thay đổi chất lượng nhận dạng thông qua thay đổi kịch bản thực nghiệm.

# Phương pháp làm

* Sử dụng HTK Toolkit của Đại học Cambridge và mô hình HMM.
* Xây dựng n mô hình tương ứng với n âm vị.
* Tạo thư mục bài tập **htklab** (trong bài tập này mặc định thư mục nằm ở ổ C của máy)
* Trong thư mục **htklab** có các mục sau: **bin32** chứa các chương trình con của HTK, **cfg** chứa các file config, **scripts** chứa các file Perl (cần cài đặt môi trường cho Perl), **wav** chứa hai thư mục con **train** và **test** để chứa các file \*.wav dùng để train và test tương ứng:



* Các bước huấn luyện (mở cmd trỏ đến thư mục **htklab**, lưu ý các file trong project này đều phải lưu dưới dạng ANSI):

|  |  |
| --- | --- |
| **Bước 1** | Tạo file **gram.txt** chứa các từ cần nhận dạng (ở bài tập này là từ 0 đến 9)  (Lưu ý có dấu xuống dòng ở cuối file, nếu không sẽ bị lỗi) |
| **Bước 2** | Tạo file **wdnet** bằng lệnh:  bin32\HParse gram.txt wdnet |
| **Bước 3** | Tạo file dict và sắp xếp theo thứ tự alphabet:  (Lưu ý có dấu xuống dòng ở cuối file, nếu không sẽ bị lỗi) |
| **Bước 4** | Tạo file thu âm với số lượng là 30:  bin32\HSGen -l -n 30 wdnet dict >train  Ta đã tạo được file **train** chứa tên và nội dung của các file cần được ghi âm và đưa vào thư mục **wav\train** |
| **Bước 5** | Chạy lệnh sau (lưu ý khi chạy các file Perl cần có từ “Perl” đứng đầu câu lệnh trong cmd):  scripts\listwavmfc.pl wav\train listwavmfccTrain.scp  Lệnh này nhằm tạo ra file **listwavmfccTrain.scp**, file này nhằm tổng hợp các file cần rút trích đặc trưng mfcc trong **wav\train** |
| **Bước 6** | Rút trích đặc trưng mfcc:  bin32\HCopy -T 1 -C cfg\HCopy.cfg -S listwavmfccTrain.scp  Ta sẽ có các file mfcc tương ứng với từng file train trong **wav\train** |
| **Bước 7** | Tạo file **train.scp**, lưu danh sách các file mfcc trong **wav\train**:  scripts\mkTrainFile.pl wav\train train.scp |
| **Bước 8** | Tạo thư mục **hmm\hmm0**, **hmm\hmm1**, …, **hmm\hmm22**  Tạo file proto mẫu theo htkbook, copy vào **hmm\hmm0**, hoặc tạo file proto với nội dụng sau và copy vào **hmm\hmm0:** |
| **Bước 9** | Khởi tạo giá trị train:  bin32\HCompV -C cfg\HCompV.cfg -f 0.01 -m -S train.scp -M hmm\hmm0 hmm\hmm0\proto |
| **Bước 10** | Tạo file **lable.mlf**:  scripts\prompts2mlf.pl lable.mlf train  Tạo file **wlist** (word list):  scripts\prompts2wlist.pl train wlist |
| **Bước 11** | Tạo file **mkphone0.led**:    (Lưu ý có dấu xuống dòng ở cuối file) |
| **Bước 12** | Tạo file **phones0.mlf**:  bin32\HLed -l '\*' -d dict -i phones0.mlf mkphone0.led lable.mlf  Tới bước này ta phải sửa đường dẫn trong file phones0.mlf như sau:  "'\*'/1..lab" -> Đường dẫn đầy đủ: "C: \htklab\wav\train\1.lab", … |
| **Bước 13** | Tạo file **lexicon**: |
| **Bước 14** | Tạo file **monophones0**:  bin32\HDMan -A -D -T 1 -m -w wlist -n monophones0 -i -l dlog dict lexicon  Sau đó sắp xếp monophones0 lại theo thứ tự alphabet được file như sau: |
| **Bước 15** | Tạo file **hmmdefs**:  scripts\mkHmmdefsFile.pl hmm\hmm0\proto monophones0 hmm\hmm0\hmmdefs |
| **Bước 16** | Tạo file **macros**:  scripts\mkMacrosFile.pl hmm\hmm0\vFloors hmm\hmm0\macros |
| **Bước 17** | Sửa **phones0.mlf**:   * Bỏ sil * Chuyển \ thành / |
| **Bước 18** | Xây dựng mô hình:  bin32\HERest -C cfg\HERest.cfg -I phones0.mlf -t 250.0 150.0 1000.0 -S train.scp -H hmm\hmm0\macros -H hmm\hmm0\hmmdefs -M hmm\hmm1 monophones0  bin32\HERest -C cfg\HERest.cfg -I phones0.mlf -t 250.0 150.0 1000.0 -S train.scp -H hmm\hmm1\macros -H hmm\hmm1\hmmdefs -M hmm\hmm2 monophones0  …  Tiếp tục cho đến hmm22 |

* Các bước test:

|  |  |
| --- | --- |
| **Bước 1** | * Tạo thư mục **kq** để lưu kết quả test * Tạo HmmList như sau (để ý thấy nội dung file này là file **monophones0** và thêm dấu “”): |
| **Bước 2** | * Thu âm các file test và bỏ vào thư mục **wav/test** * Tạo file **listwavmfccTest.scp:**   scripts\listwavmfc.pl wav\test listwavmfccTest.scp   * Rút trích đặc trưng mfcc của các file test:   bin32\HCopy -T 1 -C cfg\HCopy.cfg -S listwavmfccTest.scp   * Tạo file **test.scp**:   scripts\mkTrainFile.pl wav\test test.scp   * Chạy test:   bin32\HVite -H hmm\hmm3\macros -H hmm\hmm3\hmmdefs -S test.scp -i kq\recout.mlf -w wdnet -p 0.0 -s 5.0 dict HmmList |
| **Bước 3** | * Tạo file **phones0test.mlf**: |
| **Bước 4** | Thống kê kết quả:  bin32\HResults -I phones0test.mlf Hmmlist kq\recout.mlf, xem độ chính xác trên cmd |

# Kịch bản thử nghiệm

* Thử nghiệm với đặc trưng MFCC\_0\_D\_A

Trong đó: MFCC là đặc trưng bố cục khoang miệng khi phát âm, \_0 là độ lớn của vector, \_D là đạo hàm bậc 1 (đặc trưng tương quan theo thứ tự), \_A là đạo hàm bậc 2 (đặc trưng tương quan theo thời gian). \_X nghĩa là kết hợp MFCC với đặc trưng X (X là 0, D, A).

# Kết quả, vẽ biểu đồ

|  |  |
| --- | --- |
| **Đặc trưng** | **Độ chính xác** |
| MFCC\_0\_D\_A | 90.91% |

# Đánh giá cá nhân

* Việc kết hợp MFCC và các đặc trưng khác như đạo hàm bậc 1, đạo hàm bậc 2 làm tăng khả năng nhận dạng của mô hình.
* Chưa thực hiện kiểm thử được cho các kịch bản MFCC\_0\_D, MFCC\_0\_A, …; thay đổi số vòng lặp huấn luyện, …
* Việc sử dụng HTK đòi hỏi nhiều thời gian nghiên cứu nên các thông số cũng như các hàm trong các bước trên chưa được mô tả rõ ràng và có thể phát sinh lỗi.

# Tài liệu tham khảo

* Các bước thực hiện huấn luyện và kiểm thử phía trên được tham khảo chủ yếu (có chỉnh sửa, thêm hình ảnh và giải thích) từ file “**DauTai\_HuongDanHTK\_2014\_HCMUS.docx**” có trong thư mục **ref**
* Ngoài ra còn tham khảo ở htk\_training\_vn.pdf và htkbook.pdf