**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 4: ĐỌC CHỮ SỐ**

**GIÁO VIÊN**

**PGS. TS. Vũ Hải Quân**

**TP.HCM - 6/2018**

MỤC LỤC

1 Mô tả bài toán 2

2 Phương pháp làm 2

2.1 Tổng quát cách làm 2

2.2 Các bước huấn luyện (mở cmd trỏ đến thư mục htk, lưu ý các file trong project này đều phải lưu dưới dạng ANSI): 4

2.3 Các bước test: 5

3 Kịch bản thử nghiệm 8

4 Kết quả, vẽ biểu đồ 9

4.1 Kết qủa thực nghiệm 9

4.2 Đánh giá kết quả 9

5 Đánh giá cá nhân 10

6 Tài liệu tham khảo 10

# Mô tả bài toán

* Mục tiêu của bài tập là phát âm các kí số được nhập vào trong một tập tin.
* Đọc nội dung tập tin gồm các kí số, tìm kiếm trong bộ dữ liệu âm thanh cho trước, cắt ghép và tạo ra tập tin âm thanh tương ứng với nội dung đó.

# Phương pháp làm

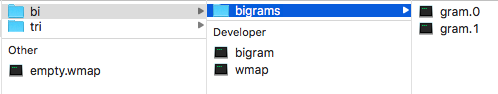
## Tổng quát cách làm

Sử dụng tiếp tục kết quả bài 2 (nhận dạng rời rạc)s

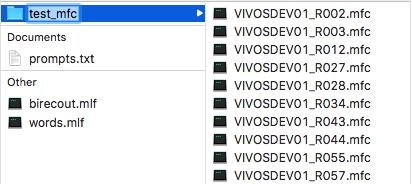
* Sử dụng HTK Toolkit của Đại học Cambridge và mô hình HMM.
* Tạo thư mục bài tập **htk** (Báo cáo giả định là thư mục chạy ngay trên ổ đĩa C)
* Cài đặt thêm môi trường **perl** và **python** cho một số bước phát sinh tập tên file...

Tổ chức dữ liệu:

* Tập dữ liệu huấn luyện gồm:
  + *Train*: gồm tất cả file âm thanh **.wav** có trong thư mục *vivos/train*.
  + *Test*: gồm tất cả file âm thanh .wav trong thư mục *vivos/test*.
* Thêm thư mục:
  + *gram*: gồm 2 thư mục con **bi** và **tri** chứa *bigram* và *trigram* tương ứng. Trong mỗi thư mục có thư mục bigrams chứa các file gram.\*. Cấu trúc file như hình:

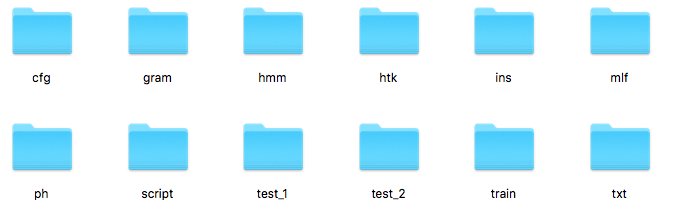


* + *test*: chứa các tập tin *prompts.txt* – dữ liệu chuỗi chữ thường, *words.mlf*, tập tin kết quả *birecout.mlf*,thư mục *test\_mfc* - danh sách file mfc phát sinh.



Cấu trúc thư mục **htk** có các mục sau:

* **cfg**: chứa các file config.
* **gram**:chứa language model bigram, trigram.
* **hmm**: chứa các thư mục hmm.
* **htk**: chứa các chương trình con của HTK.
* **ins**: chứa các file *.hed*, *.led*, *fulllist*, *tiedlist*.
* **mlf**: chứa các file *.mlf*.
* **ph**: chứa các file *phones*.
* **scripts**: chứa các file Perl (cần cài đặt môi trường cho Perl).
* **test\_1**: chứa tập tin kết qủa *recout.mlf* và *result.txt* của bigram trên toàn bộ dữ liệu *vivos/test*, chứa thư mục test\_mfc – gồm tất cả các tập tin test .*mfc*.
* **test\_2**: chứa tập tin kết qủa *recout.mlf* và *result.txt* của **bigram** và **trigram** trên 556 file dữ liệu *vivos/test*.
* **train**: tất cả tập tin train *.mfc*.
* **txt**: tập tin từ điển, và một số tập tin cấu hình khác.



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

|  |  |
| --- | --- |
| **Bước 1** | Xây dựng hệ thống LVCSR – huấn luyện mô hình cross-word triphones.   * Copy đè ccác file cấu hình trong thư mục đề bài tập cung cấp: *Hdecode.conf*, *fullist*, *mktri.hed*, *mktri.led*, *tree.hed* vào trong thư mục bài làm tương ứng. * Thực hiện chạy lại từ bước tạo triphones ở bài tập 2 với thông số thay đổi để tạo lại *tiedlist* và huấn luyện đến *hmm15*. |
| **Bước 2** | Xây dựng Language model   * Tiền xử lý dữ liệu: Tạo tập tin *train.txt* gồm: dữ liệu prompts của tập tin âm thanh huấn luyện và browse bài báo web, lấy nội dung bài báo (5Mb) * Cấu trúc file theo dạng <s> ... </s>, được kết quả: |
| **Bước 3** | Chạy các lệnh sau để tạo *bigram* và *trigram*:   * Tạo bảng map   htk/LNewMap.exe -f WFC LMName gram/empty.wmap   * Tạo wmap và gram.0   htk/LGPrep.exe -T 1 -a 100000 -b 200000 -n 3 gram/empty.wmap txt/train.txt   * Tạo bigram, trigram   htk/LBuild.exe -T 1 -c 2 0 -c 3 0 -n 3 gram/tri/wmap trigram gram.0   * Lượng giá huấn luyện. Sử dụng 1 file dữ liệu nhỏ để đánh giá bigram và trigram   htk/LPlex.exe -n 3 -t gram/tri/trigram gram/test.txt   * Tương tự, để thực hiện tạo *bigram* thì đổi **–n 3** thành **–n 2**   Kết quả lượng giá của *trigram*    Kết quả lượng giá của *bigram* |

## Các bước test:

|  |  |
| --- | --- |
| **Bước 1** | * Chuẩn bị dữ liệu test: lấy tập dữ liệu test trong thư mục vivos. * Trong thư mục **test\_1** và **test\_2** dùngđể lưu kết quả test. * Với mỗi bộ kết quả gồm 2 file: *recout.mlf* và *result.txt*   *Cấu trúc test\_1*    *Cấu trúc test\_2* |
| **Bước 2** | * Tạo file **test.scp** và **test\_mfc.scp:**   python script/listwavmfc.py test/ test\_mfc/ txt/test.scp txt/test\_mfc.scp   * Rút trích đặc trưng mfcc của các file test:   htk/HCopy.exe -T 1 -C cfg/HCopy.cfg -S txt/test.scp |
| **Bước 3** | * Tạo file **words.mlf** lưu trong thư mục **test\_result**:   perl script/prompts2mlf.pl test\_result/words.mlf test\_result/prompts.txt |
| **Bước 4** | Thực hiện test dữ liệu:  htk/HDecode.exe -H hmm/hmm15/macros -H hmm/hmm15/hmmdefs -S txt/test\_mfc.scp -t 220.0 220.0 -C cfg/HDecode.cfg -i test\_1/birecout.mlf -w gram/tri/trigram -p 0.0 -s 5.0 txt/dictHDecode.dct ins/tiedlist  Lưu ý:   * File từ điển *dictHDecode.dct* ở đây đã được thêm vào cuối file, 2 kí tự bắt đầu **<s>** và kết thúc **</s>** của trigram      * File HDecode.cfg thay đổi **STARTWORD = <s>** và **ENDWORD = </s>**     Sau khi thực hiện, tập tin recout.mlf có cấu trúc như sau: |
| **Bước 5** | Thống kê kết quả, xuất kết quả ra tập tin txt.  htk/HResults.exe -f -t -I test\_1/words.mlf ins/tiedlist test\_1/birecout.mlf > test\_1/biresult.txt  Đây là kết quả của lần test dùng hmm15.    Thực hiện kiểm thử trên bigram và trigram để có các kết quả khác. |

# Kịch bản thử nghiệm

* Tiến hành kiểm thử 1: qua 2 language model chạy trên toàn tập dữ liệu test trong thư mục *vivos/test* cho bigram.
* Tiến hành kiểm thử 2: qua 2 language model chạy trên 556 tập dữ liệu test trong thư mục *vivos/test*:
  + *Bigram*: chạy trên tập dữ liệu
  + *Trigram*: chạy trên tập dữ liệu

# Kết quả, vẽ biểu đồ

## Kết qủa thực nghiệm

**Tiến hành kiểm thử 1:** với **bigram** và toàn bộ tập dữ liệu test.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Language Model** | **Độ chính xác %** | **Tổng số từ** | **Số từ đúng** | **Lỗi chèn** |
| Bigram | 40.08 | 7722 | 3203 | 108 |

**Tiến hành kiểm thử 2:** qua 2 language model: **bigram** và **trigram** và 556 dữ liệu test.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Language Model** | **Độ chính xác %** | **Tổng số từ** | **Số từ đúng** | **Lỗi chèn** |
| Bigram | 39.26 | 5138 | 2084 | 67 |
| Trigram | 41.94 | 5138 | 2225 | 70 |

## Đánh giá kết quả

Qua kết quả thực nghiệm ở 2 lần thử nghiệm. Nhận thấy rằng:

* Xét trên tập dữ liệu đầy đủ, độ chính xác đạt được gần 40% với tổng số từ là 7722. Trong khi đó với tập dữ liệu 556 file thì độ chính xác là 39.26% với tổng số từ là 5138. Độ chính xác của việc thử nghiệm trên 1 phần bộ dữ liệu gần bằng tổng độ chính xác toàn tập dữ liệu.
* Xét trên hai mô hình bigram và trigram trên cùng tập dữ liệu thử nghiệm là 5138 từ, thì độ chính xác của trigram (41.94%) cao hơn bigram (39.26%). Tuy nhiện, thời gian để test dữ liệu khi dùng trigram, tốn nhiều thời gian hơn bigram.

Khi thực hiện bước build file gram từ cùng tập dữ liệu *train.txt* thì quá trình LBuild bigram chỉ tạo ra 2 file gram (gram.0, gram.1), còn với trigram thì tạo được 5 file gram (tu72 gram.0 đến gram.4).

# Đánh giá cá nhân

* Quá trình huấn luyện các hmm, tạo ra tiedlist với toàn bộ dữ liệu vivos tốn nhiều thời gian.
* Hdecode cho toàn bộ dữ liệu test, với mỗi file âm thanh tốn khoảng 30s để nhận dạng.
* Sử dụng trigram cho kết quả test cao hơn bigram nhưng tốn nhiều thời gian hơn.

# 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 tham khảo ở htk\_training\_vn.pdf và htkbook.pdf