

INT3411

XỬ LÝ TIẾNG NÓI

BÀI TẬP LỚN SỐ 2

Sử dụng mô hình Hidden Markov trong nhận diện tiếng nói theo từ

Người thực hiện:

Phạm Quang Anh

MSSV: 17020045

love62uet@gmail.com

Giảng viên hướng dẫn:

Dr. Trần Quốc Long

Người thực hiện:

Dương Thị Thúy Hằng

MSSV: 17020173

Ngày 3 tháng 6 năm 2020



Tóm tắt nội dung

Nhận diện tiếng nói là quá trình chuyển đổi các từ hoặc câu được nói về dạng văn bản mà con người có thể hiểu. Báo cáo này sẽ mô tả ứng dụng của mô hình Hiden Markov (HMM) trong nhận diện từ đơn hoặc đôi từ một file âm thanh. Ở đây, file âm thanh được giới hạn chứa phát âm của một trong các từ "dịch", "nhà", "học", "cách ly" và "y tế". Nội dung chính trong báo cáo được chia làm 2 phần gồm: (1): thiết kế của mô hình HMM trên tập từ vựng; (2): Trình bày về quá trình thực hiện huấn luyện và thử nghiệm. Đồng thời trong phần (2), nhóm chúng tôi sẽ trình bày một lỗi trong gói *hmmlearn* và cách khắc phục.

1 Thiết kế của mô hình HMM

Đầu tiên chúng ta cần phân tích phát âm của các từ để xác định số state trong mô hình HMM của từng từ. Hình 1 mô tả chi tiết âm vị cũng như số states được sử dụng trong mô hình HMM.

	DỊCH	HỌC	NHÀ	Y TẾ	CÁCH LY
Âm vị	z - l - c	h - ɔ - k	n - ə	j - silence - t - e	k - a - c - silence - l - j
Số states	$3 \times 3 = 9$	$3 \times 3 = 9$	$2 \times 3 = 6$	$3 \times 3 + 1 = 10$	$5 \times 3 + 1 = 16$

Hình 1: chi tiết thiết kế

Ở đây, chúng tôi sử dụng kỹ thuật subdividing phone bằng cách sử dụng 3 states (begin, middle và end) cho mỗi âm vị không phải silence thay vì 1 state như bình thường. Mỗi frame được biểu diễn bởi 36 đặc trưng MFCC ứng với 12 Cepstral features, đạo hàm bậc 1 và bậc 2 của 12 Cepstral features.

2 Kết quả thử nghiệm

2.1 Dữ liệu

Tập dữ liệu phát âm của các từ được lấy từ nhiều giọng khác nhau bằng cách cắt ra từ các câu văn có ngữ cảnh. Chi tiết về tập dữ liệu được mô tả trong bảng 1.

Từ	Tổng số lượng	Số lượng trên tập train	Số lượng trên tập test
y tế	87	67	20
dịch	81	60	21
học	99	76	23
cách ly	98	75	23
nhà	92	69	23

Bảng 1: Tỷ lệ dữ liệu

2.2 Huấn luyện mô hình

Trong bài toán này, nhóm chúng tôi chọn sử dụng mô hình HMM rời rạc vì tính đơn giản và phù hợp cho khi tập từ vựng nhỏ.

Để thực hiện và huấn luyện mô hình, nhóm sử dụng API *MultinomialHMM* của thư viện *hmmlearn*. Vì sử dụng mô hình HMM rời rạc nên chúng ta cần rời rạc hóa đặc trưng âm thanh. Ở đây, thuật toán kmeans được lựa chọn với số cụm là 20.

Tuy nhiên trong quá trình thực hiện, nhóm chúng tôi phát hiện gói *hmmlearn* có lỗi dẫn đến một số trường hợp model bị lỗi và không thể huấn luyện. Cụ thể, Đôi khi không gian rời rạc của data nào đó khi huấn luyện (vd của từ Cách ly) : có index lớn nhất nhỏ hơn với số clustering của kmeans thì sẽ có lỗi Index out of bound như hình bên dưới(Vì MultinomialHMM tính kích thước của không gian rời rạc dựa trên index lớn nhất của data):

```

~/anaconda3/envs/speech_processing/lib/python3.6/site-packages/hmmlearn/hmm.py in _compute_log_likelihood(self,
X)
    432
    433     def _compute_log_likelihood(self, X):
--> 434         return log_mask_zero(self.emissionprob_)[:, np.concatenate(X)].T
    435
    436     def _generate_sample_from_state(self, state, random_state=None):
IndexError: index 19 is out of bounds for axis 1 with size 19

```

Hình 2: Lỗi của MultinomialHMM

Để xử lý lỗi như Hình 2, chúng ta cần khởi tạo trước số symbol trong 1 state và sửa core của thư viện *hmmlearn* để chấp nhận lệnh khởi tạo:

+) Step 1 :

Trong thư viện *hmmlearn*, file *hmm.py*, lệnh if ở dòng 469 -> 477 sẽ sửa lại như sau :(thêm 2 dòng : else, return)

```

460     def _check_and_set_n_features(self, X):
461         """
462         Check if ``X`` is a sample from a Multinomial distribution, i.e. an
463         array of non-negative integers.
464         """
465         if not np.issubdtype(X.dtype, np.integer):
466             raise ValueError("Symbols should be integers")
467         if X.min() < 0:
468             raise ValueError("Symbols should be nonnegative")
469         if hasattr(self, "n_features"):
470             if self.n_features - 1 < X.max():
471                 raise ValueError(
472                     "Largest symbol is {} but the model only emits "
473                     "symbols up to {}".
474                     .format(X.max(), self.n_features - 1))
475             else:
476                 return #Fixed when index is out of bound
477             self.n_features = X.max() + 1

```

+) Step 2:

Khi thiết kế mô hình hmm, thêm dòng **hmmhmm.n_features = x** (với x là số clustering trong kmeans , ví dụ x = 22 như hình bên dưới).

```

#Clustering
train_datas[cname] = list([kmeans.predict(v).reshape(-1,1) for v in dataset[cname]])

#Initialization
hmm = hmmlearn.hmm.MultinomialHMM(
    n_components=num_state[cname], random_state=0, n_iter=1000, verbose=True,
    params='te',
    init_params='e'
)

hmm.startprob_ = np.array([0.7, 0.2, 0.1, 0., 0., 0., 0., 0., 0.])
hmm.n_features = 22
hmm.transmat_ = np.array([[0.7, 0.2, 0.1, 0., 0., 0., 0., 0., 0.],
[0., 0.7, 0.2, 0.1, 0., 0., 0., 0., 0.],
[0., 0., 0.7, 0.2, 0.1, 0., 0., 0., 0.],
[0., 0., 0., 0.7, 0.2, 0.1, 0., 0., 0.],
[0., 0., 0., 0., 0.7, 0.2, 0.1, 0., 0.],
[0., 0., 0., 0., 0., 0.7, 0.2, 0.1, 0.],
[0., 0., 0., 0., 0., 0., 0.7, 0.2, 0.1],
[0., 0., 0., 0., 0., 0., 0., 0.7, 0.3],
[0., 0., 0., 0., 0., 0., 0., 0., 1. ]])

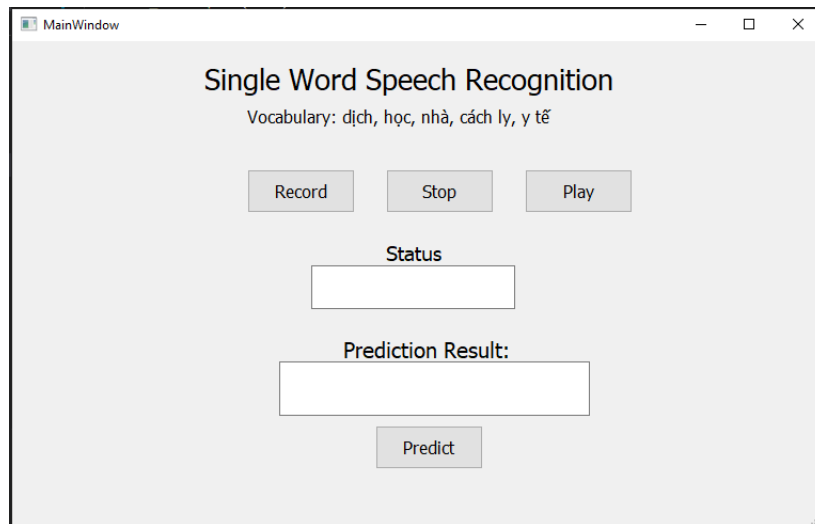
#Fitting data and train
X = np.concatenate(train_datas[cname])
lengths = list([len(x) for x in train_datas[cname]])
print("training class", cname)
print(X.shape, lengths, len(lengths))

```

Chi tiết về code và bộ tham số huấn luyện có thể được tìm thấy ở:
https://github.com/ConstantSun/Speech_Processing/blob/master/Project_%232.ipynb

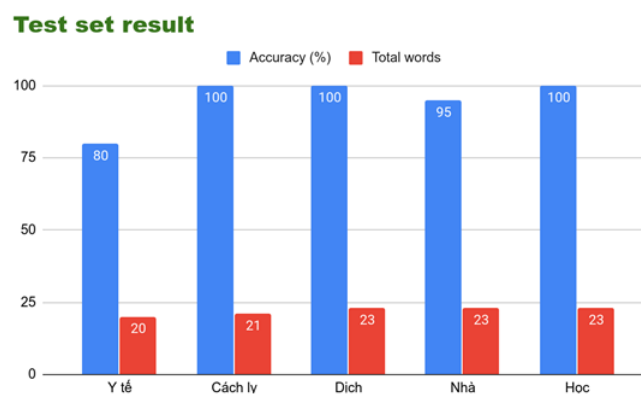
2.3 Kết quả

Để thuận tiện cho việc test trực tiếp qua mic nhóm chúng tôi có làm một phần mềm sử dụng PyQt5 như Hình 3.



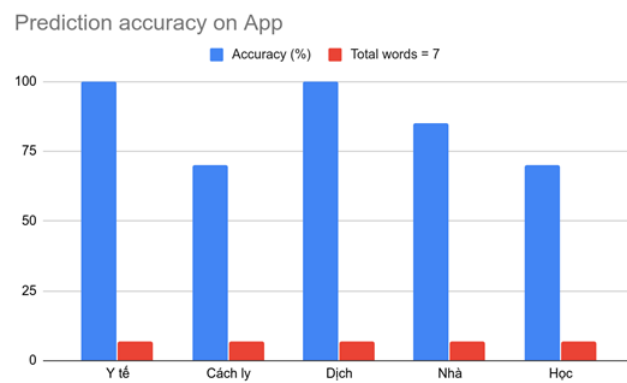
Hình 3: Giao diện app

Mô hình khi được thử nghiệm trên tập test có độ chính xác là 95%. Hình 4 mô tả chi tiết kết quả trên từng từ.



Hình 4: Kết quả trên tập test

Kết quả khi thu trực tiếp qua mic được thể hiện trong hình 5.



Hình 5: Kết quả trên khi thu qua mic

Reference:

[1]: <https://hmmlearn.readthedocs.io/en/latest/api.html>