fit@hcmus

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

ĐỀ CƯƠNG KHOÁ LUẬN TỐT NGHIỆP

# XÂY DỰNG MÔ HÌNH NHẬN DẠNG ÂM THANH TIẾNG VIỆT

***(Building Vietnamese speech recognition model)***

## 1 THÔNG TIN CHUNG

**Người hướng dẫn:**

– TS. Ngô Huy Biên (Khoa Công nghệ Thông tin)

**Nhóm Sinh viên thực hiện:**

1. Trần Ngọc Quang (MSSV:1712706 )
2. Nguyễn Hoàng Quyên (MSSV:1712712 )

**Loại đề tài:** Nghiên cứu

**Thời gian thực hiện:** Từ *9/2020* đến *3/2021*

## 2 NỘI DUNG THỰC HIỆN

### 2.1 Giới thiệu về đề tài

- Xây dựng mô hình nhận dạng âm thanh Tiếng Việt.
- Vai trò sinh viên: Data Scientist, Data Collector, Developer, Tester, Project Manager/Scrum Master.

- Kỹ năng yêu cầu : Python programming, Machine learning algorithms.
- Ngữ cảnh: Tiếng Việt được coi là một ngôn ngữ khó học với người nước ngoài bởi ngữ pháp, thanh điệu và đặc trưng vùng miền. Máy tính cũng giống như người nước ngoài - để nó nghe hiểu và diễn giải được giọng nói tiếng Việt thành dạng văn bản không phải là việc dễ dàng. Nhận dạng tiếng nói đóng vai trò quan trọng trong giao tiếp giữa người và máy. Nó giúp máy móc hiểu và thực hiện các hiệu lệnh của con người. Hiện nay trên thế giới, lĩnh vực nhận dạng tiếng nói đã đạt được nhiều tiến bộ vượt bậc. Đối với ngôn ngữ tiếng Anh, việc nhận dạng có thể đạt độ chính xác tới 99%. Khóa luận này nhằm mục đích nghiên cứu, xây dựng và đào tạo một mô hình nhận dạng giọng nói Tiếng Việt với mục tiêu độ chính xác tối thiểu 75%.

## 2.2 Mục tiêu đề tài

- Bản luận văn trình bày lí thuyết nền tảng và giải pháp để xử lý việc nhận một tập tin âm thanh tiếng Việt và xuất ra nội dung văn bản ở dạng Tiếng Việt.
- Xây dựng, thu thập dữ liệu, và đào tạo mô hình để nhận một tập tin âm thanh tiếng Việt và xuất ra nội dung ở dạng văn bản.
- Xây dựng ứng dụng web chuyển đổi từ giọng nói sang văn bản Tiếng Việt.
- Cải tiến độ chính xác của mô hình với mục tiêu là 75%.

## 2.3 Phạm vi của đề tài

- Mô hình nhận dạng giọng nói chuyển tập tin âm thanh Tiếng Việt sang văn bản Tiếng Việt.
- Sản phẩm demo được xây dựng trên nền tảng web.

## 2.4 Cách tiếp cận dự kiến

### 2.4.1 Tiếp cận âm học

Phương pháp này dựa trên lý thuyết về Âm học-Ngữ âm học. Lý thuyết đó cho biết có sự tồn tại của các đơn vị ngữ âm trong ngôn ngữ tiếng nói, các đơn vị

ngữ âm này được biểu diễn đặc trưng bởi một tập hợp những thuộc tính thể hiện trong tín hiệu âm thanh hay biểu diễn phổ theo thời gian. Đặc điểm của phương pháp nhận dạng tiếng nói theo hướng tiếp cận Âm học-Ngữ âm học:

- Người thiết kế phải có kiến thức khá sâu rộng về Âm học-Ngữ âm học.
- Phân tích các khối ngữ âm mang tính trực giác, thiếu chính xác.
- Phân loại tiếng nói theo các khối ngữ âm thường không tối ưu do khó sử dụng các công cụ toán học để phân tích.

#### 2.4.2 Tiếp cận nhận dạng mẫu thống kê

Đây là một phương pháp sử dụng trực tiếp các mẫu tiếng nói (chính là đoạn tiếng nói cần nhận dạng) mà không cần xác định thật rõ các đặc trưng và cũng không cần phân đoạn tín hiệu. Phương pháp này cũng có 2 bước:

- Bước 1: thu thập các mẫu tiếng nói: sử dụng tập mẫu tiếng nói (cơ sở dữ liệu mẫu tiếng nói) để đào tạo các mẫu tiếng nói đặc trưng (mẫu tham chiếu) hoặc các tham số hệ thống.
- Bước 2: nhận dạng mẫu: đối sánh mẫu tiếng nói từ ngoài với các mẫu đặc trưng để ra quyết định.

Cơ sở dữ liệu tiếng nói cho đào tạo có đủ các mẫu cần nhận dạng thì quá trình đào tạo có thể xác định chính xác các đặc tính âm học của mẫu, từ đó tăng độ chính xác cho mô hình.

Tiếp cận nhận dạng mẫu thường được lựa chọn cho các ứng dụng nhận dạng tiếng nói bởi các lý do sau:

- Tính dễ sử dụng và dễ hiểu trong thuật toán.
- Tính bất biến và khả năng thích nghi đối với những từ vững, người sử dụng, các tập hợp đặc trưng, các thuật toán so sánh mẫu và các quy tắc quyết định khác nhau.
- Khẳng định tính năng cao trong thực tế.

Hiện nay, một số kỹ thuật nhận dạng mẫu được áp dụng thành công trong nhận dạng tiếng nói là lượng tử hóa vector, so sánh thời gian động (DTW), mô hình Markov ẩn (HMM), mạng nơron nhân tạo (ANN), sử dụng cơ sở tri thức,...

### 2.4.3 Cách tiếp cận học máy

Tiếp cận học máy là phương pháp cố gắng "máy móc hóa" chức năng nhận dạng theo cách mà con người áp dụng trí thông minh của mình trong việc quan sát, phân tích và thực hiện những quyết định trên các đặc trưng âm học của tín hiệu. Cách tiếp cận này kết hợp các cách tiếp cận trên nhằm tận dụng tối đa các ưu điểm của chúng.

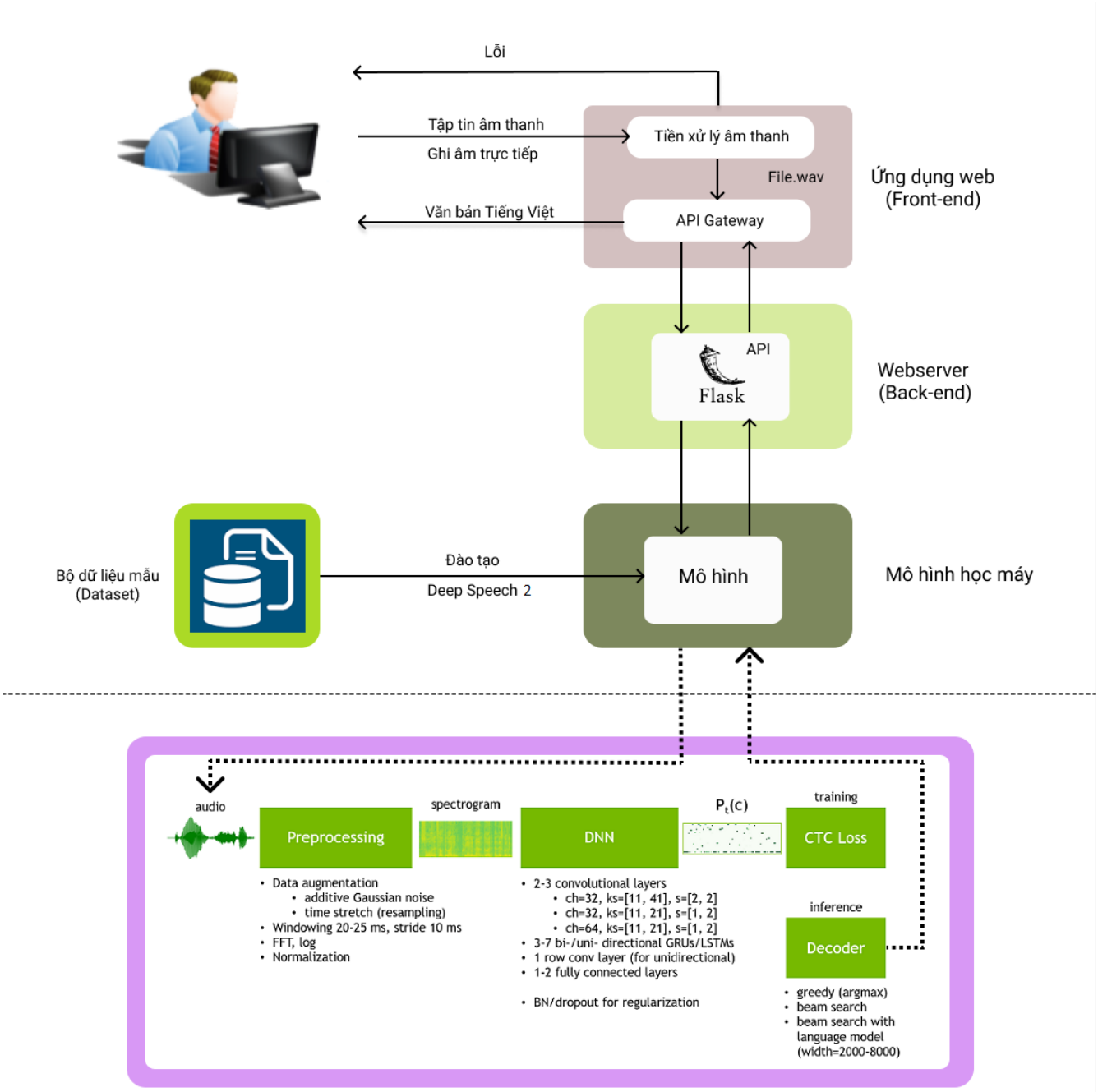
Đặc điểm của các hệ thống nhận dạng theo cách tiếp cận học máy:

- Sử dụng hệ chuyên gia để phân đoạn, gán nhãn ngữ âm. Điều này làm đơn giản hóa hệ thống so với phương pháp nhận dạng ngữ âm.
- Sử dụng mạng nơron nhân tạo để học mối quan hệ giữa các ngữ âm, sau đó dùng nó để nhận dạng tiếng nói.

Đây sẽ là hướng tiếp cận tương lai của nhận dạng tiếng nói.

### 2.4.4 Mô hình dự kiến thực hiện trong đề tài

Kiến trúc hệ thống dự kiến thực hiện với sản phẩm là trang web giới thiệu cách sử dụng mô hình được mô tả như hình:

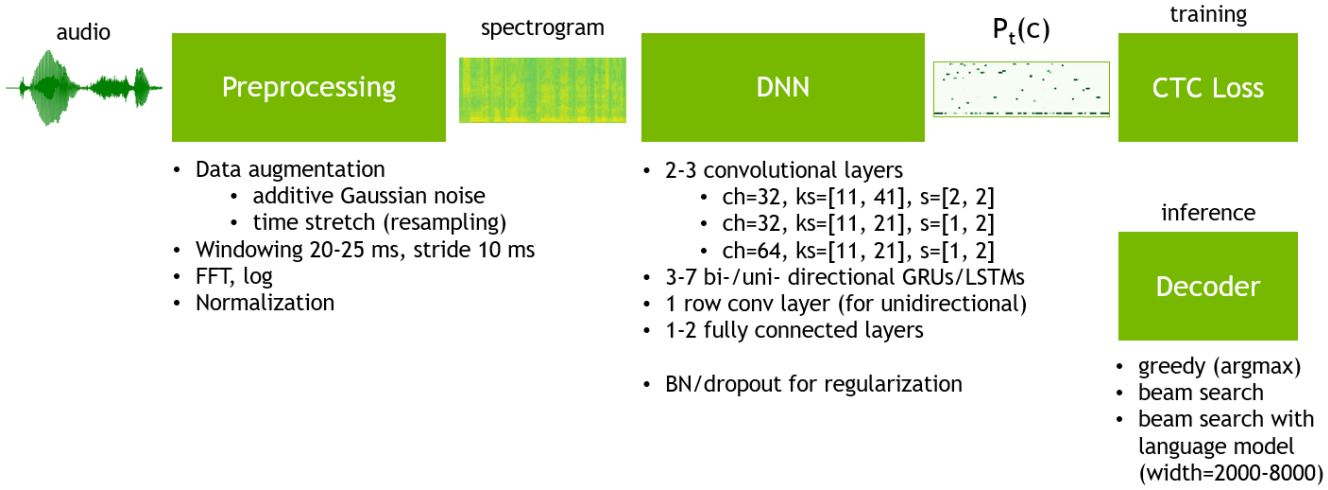



#### 2.4.4.1 Giới thiệu mô hình

Đề tài dự kiến xây dựng hệ thống nhận dạng giọng nói sử dụng kiến trúc mô hình nhận dạng đầu cuối (End-to-End). Mô hình được xây dựng kết hợp cùng ý tưởng DeepSpeech 2, một nghiên cứu của Baidu được công bố vào ngày 8/12/2015 tại Silicon Valley AI Lab. Nội dung bài báo trình bày về nhận dạng giọng nói được

thực hiện trên ngôn ngữ Tiếng Anh(English) và Tiếng Quan Thoại(Mandarin).

DeepSpeech 2 sử dụng mô hình Mạng nơ-ron hồi quy (RNN – Recurrent Neural Network) và sử dụng Connectionist Temporal Classification (CTC) để dự đoán đầu ra:




- Đầu vào của hệ thống là đoạn âm thanh thô chưa được xử lý.
- Phần tiền xử lý (Preprocessing) lấy một tín hiệu dạng sóng âm thanh thô và chuyển nó thành một biểu đồ phổ có kích thước (N_timesteps, N_frequency_features). N_timesteps phụ thuộc vào thời lượng của tệp âm thanh gốc, N_frequency_features có thể được chỉ định trong tệp cấu hình của mô hình dưới dạng thông số "num_audio_features".
- Phần Deep Neural Network (DNN) tạo ra phân phối xác suất $P_t$(c) trên các ký tự từ vựng c cho mỗi bước thời gian t.
- DeepSpeech 2 được đào tạo với nhiều thử nghiệm với mạng nơ-ron được huấn luyện với chức năng suy giảm phân loại theo thời gian kết nối (CTC) để dự đoán phiên âm giọng nói từ âm thanh.
- Tỷ lệ lỗi từ (WER) là số liệu đánh giá độ chính xác của mô hình. Để đưa các từ ra khỏi một mô hình được đào tạo, cần sử dụng một bộ giải mã (Decoder). Bộ giải mã chuyển đổi phân phối xác suất trên các ký tự thành văn bản. Có hai loại bộ giải mã thường được sử dụng với các mô hình dựa trên CTC:

bộ giải mã tham lam (Greedy decoder) và bộ giải mã tìm kiếm chùm (Beam search decoder) với mô hình ngôn ngữ.

Một bộ giải mã tham lam xuất ra ký tự có thể xảy ra nhất ở mỗi bước thời gian. Nó rất nhanh và có thể tạo ra các câu rất chính xác, nhưng có thể mắc nhiều lỗi chính tả nhỏ. Tuy nhiên, do bản chất của chỉ số WER, một lỗi ký tự cũng làm cho một từ không chính xác. Một bộ giải mã tìm kiếm chùm có chức năng ghi lại mô hình ngôn ngữ cho phép kiểm tra nhiều giải mã bằng cách chỉ định điểm cao hơn cho nhiều N-grams tùy vào mô hình ngôn ngữ nhất định.

Mô hình ngôn ngữ cũng giúp sửa lỗi chính tả. Nhược điểm là nó chậm hơn đáng kể so với một bộ giải mã tham lam.

- Đầu ra của hệ thống là một đoạn văn bản Tiếng Việt hoàn chỉnh.

Nhóm sinh viên dự kiến sẽ chỉnh mô hình thông qua thay đổi các tham số phù hợp với đặc trưng của giọng nói Tiếng Việt với nền tảng hỗ trợ việc tính toán chính bên dưới là thư viện Tensorflow. Bên cạnh đó, để đạt được độ chính xác cao nhất nhóm sinh viên dự kiến thu thập một lượng lớn bộ dữ liệu phục vụ cho huấn luyện mô hình, các nguồn của bộ dữ liệu trình bày cụ thể trong mục 2.6.

Mô hình DeepSpeech 2 trong nghiên cứu lần này này có nhiều cải tiến so với bài báo nghiên cứu trước đó của DeepSpeech:

- Để học hỏi từ tập dữ liệu lớn này, DeepSpeech 2 được tăng dung lượng mô hình thông qua chiều sâu. Kiến trúc có tới 11 lớp bao gồm nhiều lớp hồi quy hai chiều và lớp tích tụ. Các mô hình này có số lượng tính toán gần gấp 8 lần trên mỗi dữ liệu mẫu như các mô hình trong Deep Speech, làm cho việc tối ưu hóa và tính toán nhanh trở nên quan trọng. Để tối ưu hóa thành công các mô hình này, DeepSpeech 2 sử dụng Chuẩn hóa hàng loạt (Batch Normalization) cho RNN và một từ điển tối ưu hóa mới mà gọi là SortaGrad.

- Mặc dù nhiều kết quả nghiên cứu của DeepSpeech 2 sử dụng các lớp hồi quy hai chiều, DeepSpeech 2 nhận thấy rằng các mô hình tuyệt vời chỉ tồn tại bằng cách sử dụng các lớp lặp lại một chiều — một tính năng giúp triển khai các mô hình như vậy dễ dàng hơn nhiều. Kết hợp các tính năng này lại với

nhau cho phép DeepSpeech 2 tối ưu hóa một cách dễ dàng các mạng thần kinh hồi quy sâu và cải thiện hiệu suất hơn 40% ở cả tỷ lệ lỗi Tiếng Anh và Tiếng Mandarin so với các mô hình cơ sở nhỏ hơn.

- DeepSpeech 2 xem xét các mạng bao gồm nhiều lớp kết nối hồi quy (recurrent connections), bộ lọc tích chập và phi tuyến tính, cũng như tác động của một phiên bản cụ thể của Chuẩn hóa hàng loạt (BatchNorm) được áp dụng cho RNN. DeepSpeech 2 không chỉ tìm thấy các mạng tạo ra các dự đoán tốt hơn nhiều so với các mạng trong nghiên cứu trước đó, mà còn tìm thấy các trường hợp của mô hình hồi quy có thể được triển khai trong cài đặt sản xuất mà độ chính xác không bị giảm đáng kể.

- Đào tạo về số lượng lớn dữ liệu thường yêu cầu sử dụng các mô hình lớn hơn. DeepSpeech 2 có nhiều tham số hơn các tham số trong mô hình được dùng trước đó.

  Độ chính xác của mô hình dự kiến sẽ được đánh giá dựa trên tỉ lệ lỗi từ (WER).

  Tỷ lệ lỗi từ là một số liệu phổ biến về hiệu suất của hệ thống nhận dạng giọng n. Khó khăn chung của việc đo lường hiệu suất nằm ở thực tế là chuỗi từ được nhận dạng có thể có độ dài khác với chuỗi từ tham chiếu. WER có nguồn gốc từ khoảng cách Levenshtein, hoạt động ở cấp độ từ thay vì cấp độ âm vị.

  Tỷ lệ lỗi từ có thể được tính như sau:

$$WER = \frac{S + D + I}{N} = \frac{S + D + I}{S + D + C}$$

  Trong đó

  – S là số từ thay thế x.

  – D là số từ bị xóa.

  – I là số từ được thêm vào

  – C là số từ đúng.

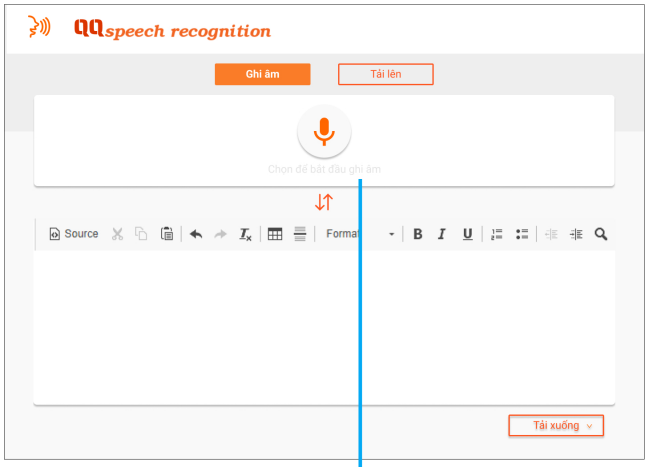
– N là tổng số từ của một câu trong câu chính xác

Độ chính xác từ được tính bằng công thức :

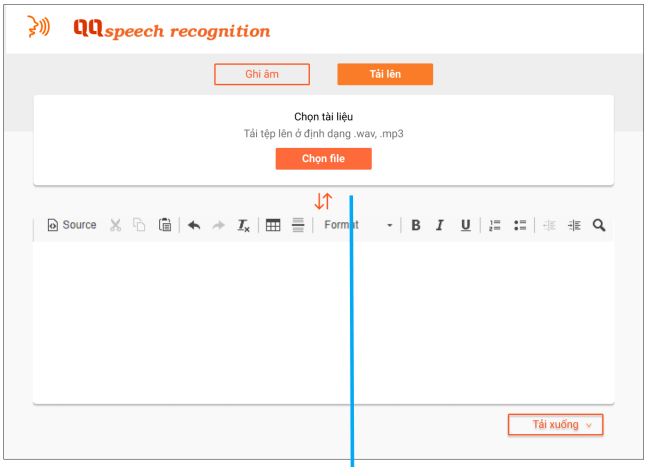
$$WAcc = 1 - WER = \frac{N - S - D - I}{N} = \frac{C - I}{N}$$

N là số từ của câu đầu vào, do đó tỉ lệ lỗi từ có thể lớn hơn 1 và độ chính xác từ có thể nhỏ hơn 0.

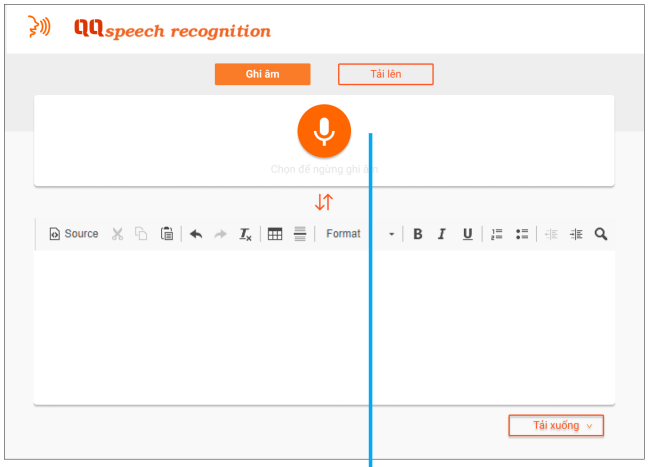
#### 2.4.4.2 Bản mẫu giới thiệu ứng dụng



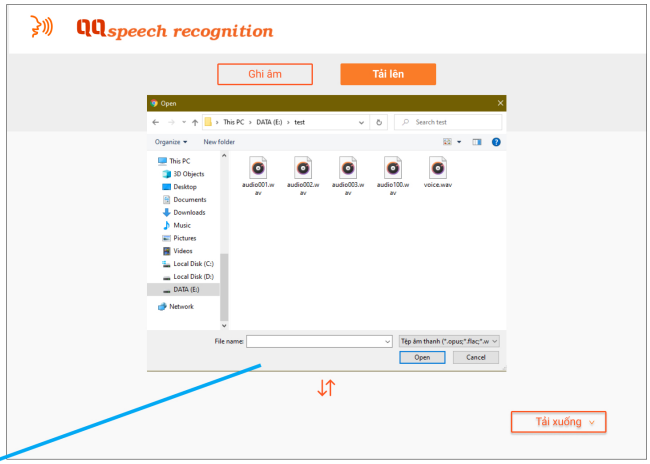

1. Màn hình chính



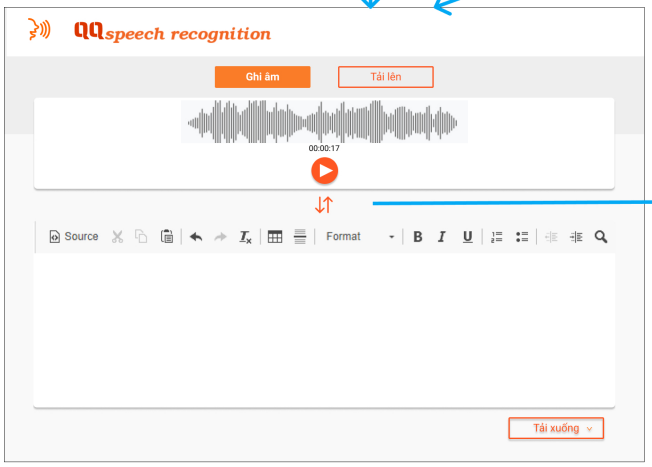

2. Chọn tải lên tập tin



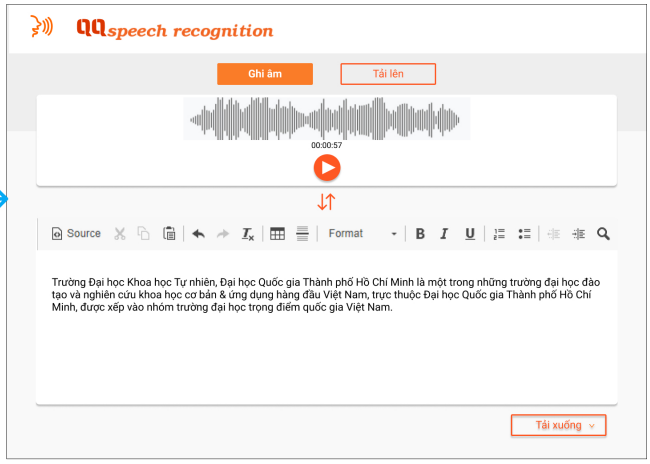

3. Ghi âm/ dừng ghi âm




4. Chọn tập tin tải lên




5. Đoạn âm thanh đã ghi/ được chọn




6. Kết quả đoạn văn bản nhận dạng

## 2.5 Kết quả dự kiến của đề tài

- Mô hình nhận dạng giọng nói Tiếng Việt.
- Dịch vụ web (API) sử dụng mô hình nhận dạng giọng nói tiếng Việt cho phép nhận vào một tập tin âm thanh Tiếng Việt sau đó chuyển sang dạng văn bản Tiếng Việt.
- Website mẫu việc sử dụng API của mô hình dịch máy từ tiếng Anh sang tiếng Việt đã xây dựng.

## 2.6 Một số nguồn dữ liệu có sẵn có thể được sử dụng

Để huấn luyện một hệ thống nhận dạng âm thanh tiếng Việt đại độ chính xác cao thì lượng dữ liệu âm thanh dùng để huấn luyện cũng phải đủ nhiều và đủ tốt Bên dưới là một số nguồn dữ liệu mà nhóm sinh viên dự kiến sử dụng:

- Bộ dữ liệu của FPT sẽ với 30 giờ dữ liệu tiếng nói đã được xử lí (gồm tiếng nói và văn bản tương ứng) – 30.000 câu hội thoại trên mạng xã hội kèm theo mô tả chi tiết về dán nhãn từ loại và tách từ.
- Bộ dữ liệu từ VIVOS cung cấp kho ngữ liệu Tiếng Việt miễn phí bao gồm 15 giờ ghi âm giọng nói. Bộ tài liệu do AILAB, phòng máy tính ĐHQG TP.HCM - Đại Học Khoa Học Tự Nhiên biên soạn, với GS.Vũ Hải Quân là chủ nhiệm.
- Bộ dữ liệu từ thu thập từ các trang đọc truyện audio, các trang báo nói,.. với gần 13000 dữ liệu là các tập tin âm thanh Tiếng Việt và các mô tả dạng văn bản tương ứng.
- Bộ dữ liệu từ do nhóm sinh viên tự thu thập thêm.
- Bộ dữ liệu từ do nhóm sinh viên tự ghi âm.

## 2.7 Kế hoạch thực hiện

| Thời gian thực hiện | Công việc thực hiện | Người thực hiện |
|---|---|---|
| 7/9/2020-13/9/2020 | • Nhận đề tài.<br>• Xây dựng bản kế hoạch sơ bộ cho các công việc cần thực hiện. | Quang, Quyên |
| 14/9/2020-20/9/2020 | • Tìm hiểu và phân tích các yêu cầu về kiến thức nền cho đề tài.<br>• Khảo sát và dùng thử các hệ thống cung cấp dịch vụ mẫu có sẵn trên thị trường: fpt.ai, https://vais.vn/,..<br>• Tạo Trello. | Quang, Quyên |
| 21/9/2020-27/9/2020 | • Thống nhất nội dung chính của ứng dụng demo việc sử dụng API.<br>• Biên soạn đề cương cho luận văn (dạng slide). | Quang, Quyên |
| 28/9/2020-<br>11/10/2020 | • Tìm hiểu lý thuyết nền tảng trong máy học.<br>• Biên soạn đề cương cho luận văn (dạng word).<br>• Tìm hiểu lý thuyết nền tảng trong việc nhận dạng giọng nói.<br>• Viết chương 1 luận văn. | Quang, Quyên |

| | | |
|---|---|---|
| 19/10/2020-<br>1/11/2020 | • Tìm hiểu về các thư viện Scikit-Learn, Tensorflow, Keras.<br>• Tìm hiểu các model và kiến trúc, chạy thử các ví dụ để đánh giá.<br>• Chỉnh sửa chương 1 luận văn. | Quang, Quyên |
| 2/11/2020-8/11/2020 | • Chạy thử mô hình nhận dạng âm thanh Tiếng Việt sang văn bản. | Quang, Quyên |
| 9/11/2020-<br>22/11/2020 | • Thu thập dữ liệu âm thanh.<br>• Chỉnh sửa dữ liệu âm thanh. | Quang, Quyên |
| 23/11/2020-<br>29/11/2020 | • Tìm hiểu và xây dựng mô hình nhận dạng âm thanh Tiếng Việt. | Quang, Quyên |
| 30/11/2020-<br>13/12/2020 | • Tiếp tục xây dựng mô hình nhận dạng âm thanh Tiếng Việt. | Quang, Quyên |
| 14/12/2020-<br>20/12/2020 | • Huấn luyện mô hình.<br>• Viết chương 2 luận văn. | Quang, Quyên |
| 21/12/2020-3/1/2020 | • Cải tiến mô hình.<br>• Chỉnh sửa chương 2 luận văn. | Quang, Quyên |
| 4/1/2020-17/1/2020 | • Viết chương 3 luận văn.<br>• Chỉnh sửa chương 3 luận văn. | Quang, Quyên |

| | | |
|---|---|---|
| 18/1/2020-31/1/2020 | • Xây dựng và triển khai hệ thống cung cấp dịch vụ web (API).<br>• Viết chương 4 luận văn. | Quang, Quyên |
| 1/2/2020-7/2/2020 | • Xây dựng ứng dụng demo việc sử dụng API trên nền tảng web.<br>• Chỉnh sửa chương 4 luận văn. | Quang, Quyên |
| 8/2/2020-14/2/2020 | • Viết chương 5 luận văn.<br>• Chỉnh sửa chương 5 luận văn. | Quang, Quyên |
| 15/2/2020-21/2/2020 | • Hoàn thành luận văn.<br>• Chỉnh sửa và cải thiện hiệu năng ứng dụng demo.<br>• Nâng cấp mô hình hoàn thiện hơn.<br>• Cải thiện hiệu năng hệ thống cung cấp dịch vụ web(API). | Quang, Quyên |
| 22/2/2020-28/2/2020 | • Hoàn chỉnh cuốn luận văn. | Quang, Quyên |
| 1/3/2020-7/3/2020 | • Hoàn chỉnh slide trình bày.<br>• Hoàn chỉnh sản phẩm khoá luận. | Quang, Quyên |

# Tài liệu

[1] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Y. Ng, "Deep speech: Scaling up end-to-end speech recognition," 2014.

[2] V. H. Nguyen, "An end-to-end model for vietnamese speech recognition," 2019.

[3] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks. proceedings of the 31st international conference on international conference on machine learning," vol. 32, 2014.

[4] I. L. Tom Hope, Yehezkel S. Resheff, *Learning TensorFlow: A Guide to Building Deep Learning Systems 1st Edition.* 2017.

[5] "Chương 3 : Lý thuyết nhận dạng giọng nói." http://read.pudn.com/downloads443/doc/comm/1868404/Nhan%20dang%20tieng%20noi%20-%20Mo%20phong%20bang%20Matlab/Chuong3%20-%20Ly%20thuyet%20nhan%20dang%20tieng%20noi_completed.pdf.

**XÁC NHẬN**
**CỦA NGƯỜI HƯỚNG DẪN**
***(Ký và ghi rõ họ tên)***

***TP. Hồ Chí Minh, ngày..../tháng..../năm....***
**NHÓM SINH VIÊN THỰC HIỆN**
***(Ký và ghi rõ họ tên)***