1. **Các mô hình speech to text**
2. Mô hình âm học (Acoustic Model)

* Hoạt động giống như tai người, đầu vào là đặc trưng của tiếng nói, X, đầu ra là xác suất có điều kiện của từ hay âm vị, P(X|W).
* Các tham số của mô hình âm học được xác định thông qua quá trình huấn luyện trên tập mẫu có trước.
* Mô hình xác suất được sử dụng phổ biến cho mô hình âm học là mô hình Markov ẩn (HMM - Hidden Markov Model) - cung cấp 1 mô hình đơn giản và hiệu quả để mô hình hóa chuỗi vector theo thời gian.
* Để mô tả phân bố xác suất của các đặc trưng đầu vào về không gian thường dùng mô hình Gaussian Mixture Model (GMM) hoặc mạng nơ ron (Neural Network)
* Một số vấn đề cần tìm hiểu khi xây dựng mô hình âm học:
* Xây dựng mô hình ổn định trong môi trường nhiễu, nhiều tạp âm.
* Mô hình hóa được giọng vùng miền khác nhau
* Mô hình hóa được các kênh truyền khác nhau
* HMM (Mô hình Markov ẩn): là mô hình thống kê máy trạng thái, cho phép chúng ta xem xét đến hai thành phần là sự kiện quan sát được và các sự kiện ẩn.

VD: Trong nhận dạng giọng nói thì sự kiện quan sát được là các đặc trưng âm học của tiếng nói, còn sự kiện ẩn là các từ.

* HMM bao gồm các thành phần chính:
* Q = q1, q2, …, qN: là tập của N trạng thái
* O = o1, o2, …, oT: là một chuỗi T các quan sát tại các thời điểm t khác nhau.

Tương ứng với mỗi trạng thái tại thời điểm t sẽ có 1 tập V = {o1, o2, …, om} là tập hợp các quan sát có thể được quan sát thấy trong mỗi trạng thái.

* 𝐵 = {𝑏𝑗 (𝑣𝑘 )}: B là phân bố xác suất quan sát được các quan sát o trong trạng thái qj.
* *Π = {π1, π2, …, πN}*: tập các phân bố xác suất cho trạng thái khởi đầu, πi là xác suất để trạng thái i được chọn tại thời điểm khởi đầu t = 1
* HMM có 3 bài toán chính:
* Bài toán 1: Computing likelihood (Tính toán)
* Bài toán 2: Decoding (Nhận dạng)
* Bài toán 3: Learning (Huấn luyện)
* 1 âm vị thường được biểu diễn bằng HMM tuyến tính 3 hoặc 5 trạng thái.

1. Mô hình ngôn ngữ (Language Model)

* Mô tả não người
* Lưu trữ những tri thức về từ ngữ, ngữ pháp, những thông tin liên quan đến knowledge của 1 ngôn ngữ.
* Những kiến thức được thay đổi liên tục => mô hình ngôn ngữ cần được update thường xuyên.
* Mô hình ngôn ngữ và phổ biến hiện nay là mô hình n-gram.
* Một số vấn đề cần tìm hiểu:
* Xây dựng mô hình ngôn ngữ cho các domain khác nhau
* Xây dựng mô hình ngôn ngữ cho các domain có ít dữ liệu
* Mô hình ngôn ngữ n-gram
* Công thức: P(w|h), tính xác suất của từ w khi biết trước các từ trước nó h



( n có thể là 1 unigram, 2 bigram, 3 trigram)

* Nhược điểm:
* Được đào tạo dựa trên 1 kho ngữ liệu cố định -> dữ liệu ở ngoài tập ngữ liệu này thì xác suất bằng 0
* Thiếu tính tổng quát vì tùy từng thể loại, từng chủ đề sẽ có các cách kết hợp câu, từ khác nhau.
* Giải pháp khắc phục nhược điểm: sử dụng các mô hình ngôn ngữ học sâu.

1. **Các mô hình Speech Recognition**
2. **Mô hình Wav2Vec Facebook - học cách nhận dạng giọng nói từ dữ liệu không được gắn nhãn**

* Thực hiện phương pháp tự giám sát (còn được gọi là không giám sát)
* Được cung cấp dữ liệu “không xác định” - không có nhãn nào được xác định trước đó tồn tại. => Hệ thống phải tự dạy cách phân loại dữ liệu, xử lý dữ liệu để học hỏi từ cấu trúc của nó.
* Wav2vec học hoàn toàn từ giọng nói và văn bản được ghi lại, loại bỏ nhu cầu phiên âm. Sử dụng mô hình tự giám sát và khuôn khổ wav2vec 2.0 của Facebook, phương pháp phân nhóm => phân đoạn các bản ghi âm thành các đơn vị tương ứng lỏng lẻo với âm thanh cụ thể.
* Để học cách nhận dạng các từ trong một đoạn ghi âm, Facebook đã đào tạo một mạng đối thủ chung (GAN) bao gồm 1 bộ đào tạo và 1 bộ phân biệt
* Bộ tạo âm thanh lấy các phân đoạn âm thanh và dự đoán âm vị (đơn vị âm thanh) tương ứng với một âm thanh trong ngôn ngữ. Nó được đào tạo bằng cách cố gắng đánh lừa người phân biệt, đánh giá xem các trình tự dự đoán có thực tế hay không.
* Đối với bộ phân biệt, nó học cách phân biệt giữa đầu ra nhận dạng giọng nói của trình tạo và văn bản thực từ các ví dụ về văn bản từ các nguồn đã được “phiên âm hóa”, ngoài đầu ra của trình tạo.

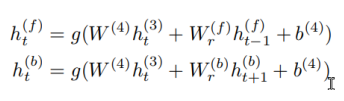
1. **Mô hình DeepSpeech**

* DeepSpeech được phát triển bởi phòng nghiên cứu Baidu Research. DeepSpeech nổi lên từ năm 2014, là một trong những mô hình đầu tiên áp dụng end-to-end cho bài toán nhận dạng tiếng nói.
* Mô hình DeepSpeech bao gồm các thành phần như:
* Mạng RNN đơn giản thay bởi Long short term memory (LSTM) phức tạp.
* Sử dụng CTC layer để tự động căn lề giữa đầu vào và đầu ra của mạng.
* Kết hợp mô hình ngôn ngữ.
* Do sử dụng CTC layer, nên tốc độ hội tụ của việc huấn luyện khá chậm. Đồng thời, kết quả nhận dạng cho độ chính xác thấp hơn so với phương pháp dạng lai như Kaldi.
* Mạng end-to-end được thực hiện theo kiến trúc deep-speech 2 của google, là mạng dựa trên kiến trúc thuần không kết nối lặp và mạng nơron kết nối lặp mà không sử dụng kiến trúc có bộ nhớ phức tạp như LSTM. Phương pháp này làm giảm độ phức tạp và cải thiện hiệu năng của hệ thống nhận dạng. Bằng việc kết hợp với hàm kích hoạt là ReLu, kiến trúc này giải quyết được vấn đề vanishing gradient trong kiến trúc RNN truyền thống và học được các phụ thuộc dài của LSTM.
* Kiến trúc mạng bao gồm 5 tầng ẩn. 3 tầng đầu là mạng không lặp, là các tầng truyền thẳng cơ bản. Với hàm kích hoạt ReLu

g(z) = min{max{0, z}, 20}



* Tầng thứ 4 là tầng lặp 2 chiều (bi-directional recurrent layer):



* Tầng cuối cùng trong kiến trúc mạng là tầng CTC. Đây là tầng quan trọng nhất trong các kiến trúc mạng End-to-End ban đầu. Tầng CTC có chức năng thực hiện tự động alignment các nhãn dữ liệu đầu vào và nhãn dữ liệu đầu ra. Điều này giúp các tín hiệu tiếng nói và kí tự chữ có thể tự động được học mà không cần sự gán nhãn của con người.

1. **Mô hình Wav2Vec Vietnam**

* Được đào tạo trên 13 nghìn giờ âm thanh youtube tiếng Việt (dữ liệu không có nhãn) và được tinh chỉnh trên 250 giờ có nhãn của bộ dữ liệu VLSP ASR trên âm thanh giọng nói được lấy mẫu 16kHz.
* Sử dụng kiến trúc wav2vec2 cho mô hình được đào tạo trước: Lần đầu tiên, việc học các biểu diễn mạnh mẽ chỉ từ âm thanh giọng nói, sau đó tinh chỉnh giọng nói được phiên âm có thể vượt trội hơn các phương pháp bán giám sát tốt nhất trong khi đơn giản hơn về mặt khái niệm.
* Đối với giai đoạn tinh chỉnh, wav2vec2 được tinh chỉnh bằng cách sử dụng Connectionist Temporal Classification (CTC), đây là một thuật toán được sử dụng để huấn luyện mạng thần kinh cho các vấn đề theo trình tự và chủ yếu trong Nhận dạng giọng nói tự động.

| Model | #params | Dữ liệu trước khi đào tạo | Dữ liệu sau khi tinh chỉnh |
| --- | --- | --- | --- |
| base | 95M | 13k hours | 250 hours |

* Trong một hệ thống ASR chính thức, cần có 2 thành phần: mô hình âm thanh và mô hình ngôn ngữ.
* Mô hình tinh chỉnh ctc-wav2vec hoạt động như 1 mô hình âm thanh.
* Mô hình ngôn ngữ cung cấp mô hình 4-grams được đào tạo trên 2GB văn bản nói.

1. **Mô hình Jaber and Quartet**
2. **Mô hình Whisper**

* Whisper là mô hình nhận diện giọng nói đa năng được đào tạo trên tập dữ liệu lớn và đa dạng giúp thực hiện các tác vụ nhận diện giọng nói và dịch ngôn ngữ.
* Whisper có tối đa 5 mô hình máy học với dung lượng khác nhau từ bé (tiny) đến lớn (large), cho phép người dùng lựa chọn mô hình tối ưu nhằm cân đối tốc độ xử lý với độ chính xác.
* OpenAI mô tả Whisper như 1 mô hình biến đổi mã hóa-giải mã, một loại mạng thần kinh có thể sử dụng ngữ cảnh thu thập được từ dữ liệu đầu vào, tìm hiểu các liên kết tình huống, sau đó có thể được dịch sang ngôn ngữ cần thiết ở đầu ra của mô hình.
* Âm thanh đầu vào được chia thành các đoạn dài 30s, chuyển đổi thành biểu đồ quang phổ log-Mel và sau đó chuyển vào bộ mã hóa. Một bộ giải mã được đào tạo để dự đoán văn bản chú thích tương ứng, trộn lẫn với các mã thông báo đặc biệt.
* Tỉ lệ nhận diện sai của whisper phụ thuộc vào loại ngôn ngữ, bảng dưới đây thể hiện thống kê tỷ lệ sai của từng loại ngôn ngữ khi sử dụng mô hình lớn nhất của whisper. Với tiếng Việt, tỉ lệ này là 10.7% còn đối với tiếng Anh là 4.5%. Tỉ lệ lỗi thấp nhất với tiếng Tây Ban Nha, chỉ với 3.5%.

