TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

ĐỒ ÁN TỐT NGHIỆP

Xây dựng mô hình tổng hợp tiếng nói tiếng Việt nhúng đặc trưng người nói

NGUYỄN THỊ NGỌC ÁNH
anh.ntn160282@sis.hust.edu.vn

Ngành Công nghệ thông tin
Chuyên ngành Hệ thống thông tin

Giảng viên hướng dẫn: PGS. TS. Thân Quang Khoát
Chữ ký của GVHD

Bộ môn: Hệ thống thông tin
Viện: Công nghệ thông tin và Truyền thông

HÀ NỘI, 12/2020

PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

1. Thông tin về sinh viên

Họ và tên sinh viên: Nguyễn Thị Ngọc Ánh

Điện thoại liên lạc: 0342612379 Email: anh.ntn160282@sis.hust.edu.vn

Lớp: CNTT2.03 - K61 Hệ: Kỹ sư chính quy

Đồ án tốt nghiệp được thực hiện tại: Trường Đại học Bách khoa Hà Nội

Thời gian làm ĐATN: từ ngày 15/09/2020 đến ngày 23/12/2020

2. Mục đích nội dung của ĐATN

Xây dựng mô hình học sâu cho bài toán tổng hợp tiếng nói tiếng Việt nhúng đặc trưng người nói.

3. Các nhiệm vụ cụ thể của đồ án tốt nghiệp

- Tìm hiểu cơ sở lý thuyết học sâu và lĩnh vực tổng hợp tiếng nói.
- Xây dựng mô hình học sâu cho bài toán tổng hợp tiếng nói nhúng đặc trưng người nói.
- Chuẩn bị dữ liệu, huấn luyện mô hình và đánh giá kết quả.

4. Lời cam đoan của sinh viên

Tôi – Nguyễn Thị Ngọc Ánh cam kết Đồ án tốt nghiệp (ĐATN) là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của PGS.TS. Thân Quang Khoát. Các kết quả đạt được và nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác. Tất cả những tham khảo trong ĐATN đều được ghi rõ ràng và đầy đủ nguồn gốc trong danh mục tài liệu kham khảo. Tôi xin hoàn toàn chịu trách nhiệm nếu vi phạm quy chế của nhà trường.

Hà Nội, ngày tháng năm

Tác giả ĐATN

Nguyễn Thị Ngọc Ánh

Xác nhận của giảng viên về mức độ hoàn thành ĐATN và cho phép bảo vệ

.....

.....

Hà Nội, ngày tháng năm

Giảng viên hướng dẫn

PGS.TS. Thân Quang Khoát

Lời cảm ơn

Đầu tiên, tôi xin cảm ơn bố mẹ luôn định hướng, lắng nghe và ủng hộ tôi trên mọi bước đường mình chọn, giúp tôi tự tin theo đuổi những điều mình muốn.

Tiếp đến, tôi xin gửi lời cảm ơn tới thầy Thân Quang Khoát, người đã hướng dẫn tôi trong suốt thời gian qua để tôi có thể hoàn thành đồ án này.

Tôi cũng xin cảm ơn anh Đỗ Văn Hải cùng các đồng nghiệp tại trung tâm không gian mạng Viettel đã luôn giúp đỡ tôi trong quá trình hoàn thành đồ án.

Thêm nữa, tôi muốn gửi lời cảm ơn tới những người bạn đã tin tưởng và đồng hành cùng tôi trên chặng đường khám phá tri thức và hoàn thiện bản thân.

Cuối cùng, tôi xin cảm ơn mái trường Bách khoa thân yêu, cảm ơn những người thầy, người cô đã mang đến cho tôi một môi trường học tập thân thiện, một phần thanh xuân đáng nhớ.

Xin chân thành cảm ơn!

Sinh viên thực hiện
Ký và ghi rõ họ tên

Tóm tắt nội dung đồ án

Hiện nay, lĩnh vực tổng hợp tiếng nói đã được nghiên cứu và phát triển ở rất nhiều nơi trên thế giới, nhiều công nghệ và phương pháp khác nhau được thử nghiệm, triển khai thành công, thậm chí có những công trình đã đạt đến mức khó có thể phân biệt được với giọng đọc của con người. Tại Việt Nam, các hệ thống tổng hợp tiếng nói ngày càng được áp dụng rộng rãi trong nhiều lĩnh vực của cuộc sống, tuy nhiên chi phí để xây dựng một hệ thống tổng hợp giọng nói chất lượng vô cùng tốn kém. Chính vì lý do này, đồ án tập trung nghiên cứu công nghệ tổng hợp tiếng nói dựa trên mạng nơ ron học sâu cho tổng hợp giọng nói tiếng Việt với mục đích mở rộng sự đa dạng giọng nói cho các hệ thống tổng hợp tiếng nói bằng lượng chi phí về dữ liệu cũng như thời gian tối thiểu.

Để làm được điều này, đồ án đề xuất mô hình tổng hợp giọng nói một người dựa trên mô hình đa người nói nhúng đặc trưng người nói bằng phương pháp học chuyển tiếp. Sau khi tìm hiểu và xây dựng mô hình, đồ án tiếp tục thử nghiệm với các bộ dữ liệu người nói tiếng Việt với thời lượng khác nhau, tiến hành so sánh với một số phương pháp tổng hợp tiếng nói bằng mô hình học sâu khác, chứng minh được phương pháp đề xuất có hiệu quả trong việc giảm kích thước dữ liệu và thời gian huấn luyện trong khi vẫn giữ được chất lượng giọng nói sinh ra.

MỤC LỤC

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG

DANH MỤC TỪ VIẾT TẮT VÀ THUẬT NGỮ

Từ viết tắt /thuật ngữ	Từ đầy đủ	Ý nghĩa
TTS	Text-to-Speech	Văn bản thành giọng nói
HMM	Hidden Markov model	Mô hình Markov ẩn
GMM	Gaussian mixture model	Mô hình Gaussian hỗn hợp
ANN	Artificial neural network	Mạng nơ ron nhân tạo
DNN	Deep neural network	Mạng nơ ron học sâu
CNN	Convolutional Neural Network	Mạng nơ ron tích chập
RNN	Recurrent Neural Network	Mạng nơ ron hồi quy
LSTM	Long-Short Term Memory	Mạng bộ nhớ dài-ngắn
GRU	Gated Recurrent Unit	Đơn vị tái phát
end-to-end		Từ đầu đến cuối
Seq2seq	Sequence to sequence	Mô hình tuần tự
GE2E	generalized end-to-end	
MOS	Mean opinion score	Điểm ý kiến trung bình
F0	Fundamental frequency	Tần số cơ bản
GTA	Ground truth audio	Âm thanh thực
mel-spectrogram		Một biểu diễn tham số âm học mức thấp
sample rate		Tốc độ lấy mẫu
phoneme		Âm vị
ĐATN	Đồ án tốt nghiệp	

CHƯƠNG 1. MỞ ĐẦU

1.1 Giới thiệu về tổng hợp tiếng nói

Tổng hợp tiếng nói là quá trình tạo ra giọng nói của người từ đầu vào là văn bản hoặc các mã hóa việc phát âm. Ở thời điểm hiện tại, khi nhắc đến hệ thống tổng hợp tiếng nói, đa số ám chỉ hệ thống nhận đầu vào là một văn bản và tạo ra tín hiệu tiếng nói tương ứng ở đầu ra.

Nghiên cứu về tổng hợp tiếng nói đã bắt đầu từ rất lâu, năm 1779 nhà khoa học người Đan Mạch Christian Kratzenstein đã xây dựng mô phỏng đơn giản hệ thống cấu âm của con người, mô hình này đã có thể phát ra được âm thanh của một số nguyên âm dài. Đến tận thế kỷ 19 các nghiên cứu tổng hợp tiếng nói vẫn còn ở mức đơn giản, phải sang thế kỷ 20 khi mà có sự lớn mạnh của hệ thống điện, điện tử thì mới thực sự xuất hiện những hệ thống tổng hợp tiếng nói chất lượng, có thể kể đến như hệ thống VODER lần đầu được giới thiệu năm 1939 [1]. Phải tới khi Bell Labs công bố nghiên cứu của họ về việc tổng hợp đa ngôn ngữ dựa trên các hướng tiếp cận "Xử lý ngôn ngữ tự nhiên" năm 1997 thì lĩnh vực này mới bắt đầu được khai thác. Nhìn chung, đến thời điểm này chất lượng của các hệ thống tổng hợp vẫn còn rất tệ, phải đến đầu những năm 2000 chất lượng và độ tự nhiên mới có sự nhảy bậc khi áp dụng tổng hợp thống kê dựa trên các mô hình Markov ẩn. Gần đây những nghiên cứu về mạng nơ ron học sâu được dẫn đầu bởi Google đã cho thấy những bước tiến nổi bật khi áp dụng vào tổng hợp tiếng nói, chất lượng đã đạt đến độ rất cao và khó có thể phân biệt là người hay máy nói. Công nghệ tổng hợp tiếng nói hướng đến các mục tiêu đa dạng hơn như tổng hợp lời nói có cảm xúc hay hệ thống có sự đa dạng giọng nói.

1.2 Ứng dụng của tổng hợp tiếng nói

Các hệ thống tổng hợp tiếng nói ngày càng được áp dụng ngày càng rộng rãi trong nhiều lĩnh vực của cuộc sống. Một ứng dụng phổ biến của hệ thống tổng hợp tiếng nói là được tích hợp trong các ứng dụng trợ lý ảo trên điện thoại và máy tính như Siri của Apple[1], Google Assistant của Google[2] hay Cortana của Microsoft[3] để nâng cao trải nghiệm tương tác giữa người sử dụng và máy.

Phổ biến tại Việt Nam là ứng dụng trong các lĩnh vực sách nói, báo nói khi số lượng phương tiện sách báo được xuất bản và phát hành mỗi năm ngày càng lớn. Một ứng dụng khác của hệ thống tổng hợp tiếng nói là tổng đài chăm sóc khách hàng tự động. Cùng với sự phát triển của công nghệ nhận dạng tiếng nói và xử lý

[1] https://www.apple.com/ios/siri
[2] https://assistant.google.com
[3] https://www.microsoft.com/en-us/cortana

ngôn ngữ tự nhiên, ứng dụng giúp thực hiện nhiều cuộc gọi cùng một lúc mà không bị giới hạn bởi các nguồn lực như con người hay chi phí, thời gian.

Với nhiều ứng dụng trong hiện tại và tiềm năng phát triển trong tương lai, giá trị mà công nghệ tổng hợp tiếng nói mang lại vô cùng lớn, do đó việc phát triển công nghệ này vô cùng cần thiết.

1.3 Vấn đề đặt ra

Hiện nay, có nhiều công ty có sản phẩm sử dụng công nghệ tổng hợp tiếng nói, càng có nhiều nhu cầu về sự đa dạng của tiếng nói tổng hợp. Để phát triển một giọng nói tổng hợp mới đáp ứng yêu cầu về chất lượng bằng công nghệ học sâu như Tacotron-2, cần tốn hàng chục giờ dữ liệu giọng nói tự nhiên với cách phát âm chính xác, ngữ điệu sống động và hạn chế tiếng ồn xung quanh, tương đương với việc thu thập một lượng lớn dữ liệu chất lượng cao đối với nhiều người nói bằng cách thu âm hàng trăm giờ với phát thanh viên cùng văn bản có độ phủ tốt trong phòng thu chuyên dụng là không khả thi. Đồng thời việc phát triển một hệ thống tổng hợp tiếng nói cũng rất tốn kém không chỉ về chi phí mà còn về thời gian.

Do đó, đồ án này hướng đến những kỹ thuật giúp giảm lượng dữ liệu giọng nói huấn luyện cần thiết trong khi vẫn giữ được chất lượng cho hệ thống tổng hợp tiếng nói tiếng Việt dựa trên mạng nơ ron học sâu. Để làm được điều này, đồ án đề xuất hướng tiếp cận tổng hợp giọng nói một người dựa trên mô hình đa người nói nhúng đặc trưng người nói bằng phương pháp học chuyển tiếp. Kết quả thử nghiệm cho thấy đồ án đã đạt được mục tiêu đề ra, giúp giảm dữ liệu huấn luyện trong khi vẫn giữ được chất lượng giọng nói tổng hợp.

Nội dung của đồ án được trình bày theo bố cục gồm 6 chương:

- Chương 1: Giới thiệu khái quát bài toán tổng hợp tiếng nói và ứng dụng thực tiễn cũng như những thách thức đặt ra hiện nay.
- Chương 2: Các cơ sở lý thuyết về học sâu và các phương pháp tổng hợp tiếng nói.
- Chương 3: Tình hình phát triển của tổng hợp tiếng nói cũng như đề ra hướng tiếp cận giúp giải quyết khó khăn mà bài toán đang gặp phải.
- Chương 4: Chi tiết cụ thể về kiến trúc mô hình tổng hợp tiếng nói đề xuất.
- Chương 5: Cách thức cài đặt, thử nghiệm và đánh giá kết quả hệ thống tổng hợp giọng nói.
- Chương 6: Kết luận và một số hướng phát triển trong tương lai có thể thử nghiệm để cải thiện những điểm chưa giải quyết.

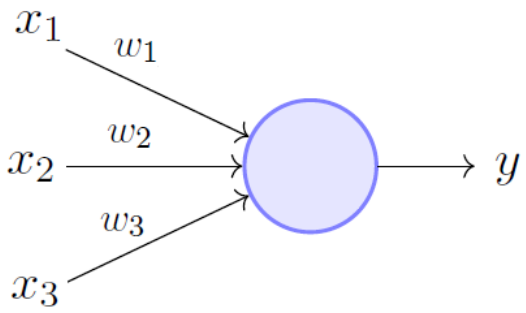
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1 Tổng quan về học sâu

Học sâu là một chi của lĩnh vực học máy, có khả năng khác biệt ở một số khía cạnh quan trọng so với học máy truyền thống, cho phép máy tính giải quyết một loạt các vấn đề phức tạp không thể giải quyết được như máy bay không người lái, ô tô tự hành, dự đoán hành vi,.... Nội dung phần này sẽ chủ yếu trình bày về hướng tiếp cận lý thuyết học sâu sử dụng mạng nơ ron nhân tạo, vì đây là phương pháp được áp dụng cho việc xây dựng hệ thống tổng hợp tiếng nói tiếng Việt của đề tài.

2.1.1 Mạng nơ ron nhân tạo

Mạng nơ ron nhân tạo (Artificial Neural Network, viết tắt là ANN) là một mô hình xử lý thông tin được mô phỏng dựa trên hoạt động của hệ thống thần kinh của sinh vật, bao gồm số lượng lớn các nơ ron được gắn kết để xử lý thông tin. ANN giống như bộ não con người, được học bởi kinh nghiệm (thông qua huấn luyện), có khả năng lưu giữ những kinh nghiệm hiểu biết (tri thức) và sử dụng những tri thức đó trong việc dự đoán các dữ liệu chưa biết. ANN được xây dựng giống như bộ não con người với các nơ ron nhân tạo chính là các nút (đơn vị) được kết nối với nhau. Mỗi nơ ron chính là một tế bào thần kinh có nhiệm vụ xử lý thông tin bằng cách xử lý tín hiệu đầu vào và có thể phát tín hiệu với các nơ ron được kết nối với nó. Nhiều nơ ron thường được tổ chức có hệ thống tạo thành các lớp (layer). Các đơn vị ở lớp đầu vào (input layer) nhận được nhiều dạng và cấu trúc thông tin khác nhau, truyền qua các lớp dựa trên hệ thống trọng số nội bộ và mạng nơ ron cố gắng tìm hiểu về thông tin để tạo ra một kết quả ở lớp đầu ra (output layer). Giống như con người cần các quy tắc và hướng dẫn để đưa ra thực hiện một phép xử lý tính toán, ANN cũng sử dụng một tập hợp các quy tắc học tập được gọi là lan truyền ngược để hoàn thành kết quả đầu ra của chúng.



Hình 2.1 Perceptron [2]

Mạng nơ ron nhân tạo là nền tảng của trí tuệ nhân tạo (AI) và giải quyết các vấn đề được cho là khó theo tiêu chuẩn thống kê hoặc bằng con người. Nó đã được ứng dụng trong mọi lĩnh vực hoạt động, phá vỡ các cách hoạt động truyền thống. Nhiều nhiệm vụ được giải quyết bằng ANN như dịch thuật, phương tiện không người lái,

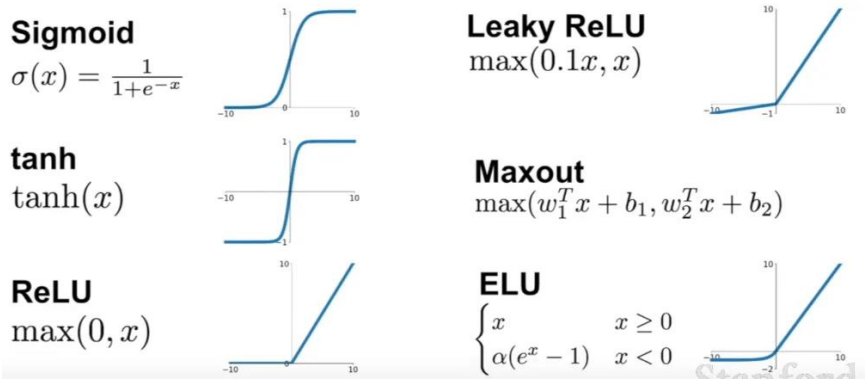
quản lý giao dịch tự động, chuẩn đoán y tế, đề xuất dựa trên cá nhân hoá, nhận dạng tiếng nói, cảnh báo quân sự, ...

2.1.2 Logistic regression

Logistic regression là mô hình mạng nơ ron nhân tạo có một đơn vị nơ ron. Đây là cấu trúc nơ ron đơn giản nhất chỉ với lớp đầu vào và lớp đầu ra. Mô hình của logistic regression là được mô tả bằng PT 2.1.

$$\hat{y} = \theta(\omega^T x + b) \qquad PT\ 2.1$$

Trong đó: w là trọng số được cập nhật, x là dữ liệu nhận từ đầu vào, b là bias và θ là hàm kích hoạt (activation function). Một số hàm kích hoạt thường được sử dụng trong mạng nơ ron nhân tạo là hàm sigmoid, hàm tanh, hàm ReLU. Hình 2.2 mô tả hình dạng một số hàm kích hoạt phổ biến.

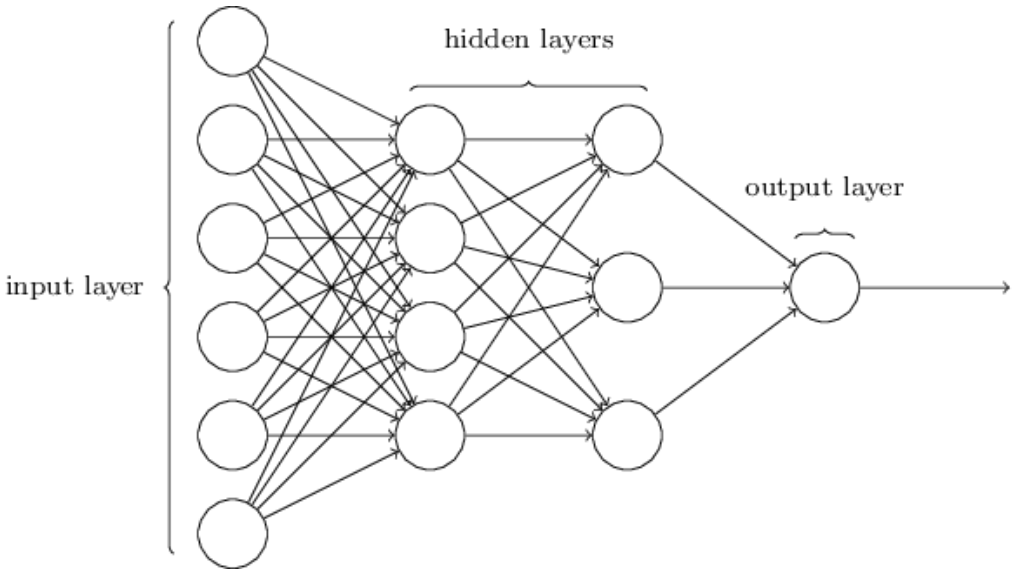


Hình 2.2 Một số hàm kích hoạt phổ biến. [3]

2.1.3 Mạng nơ ron học sâu

Mạng nơ ron học sâu (Deep Neural Network, viết tắt là DNN) là một mạng nơ ron nhân tạo (ANN) thường có ít nhất ba lớp, với nhiều lớp đơn vị ẩn ở giữa lớp đầu vào và đầu ra. Mạng nơ ron học sâu có thể mô hình hóa mối quan hệ giữa đầu vào và đầu ra bằng các phép biến đổi tuyến tính hay phi tuyến phức tạp.

Hình 2.3 mô tả một mạng nơ ron đơn giản. Lớp ngoài cùng bên trái gọi là lớp đầu vào và các nơ ron trong lớp này được gọi là nơ ron đầu vào, đây cũng chính là nơi nhận đầu vào của mạng nơ ron. Lớp ngoài cùng bên phải là lớp đầu ra (output), lớp này trả về giá trị đầu ra tương ứng với những đầu vào được nhận từ lớp đầu vào. Hai lớp ở giữa được gọi là lớp ẩn (hidden layers), lớp này không nhận đầu vào cũng như đầu ra, một mạng nơ ron có thể có một hoặc nhiều lớp ẩn. Việc các lớp này xử lý theo cách nào thường phụ thuộc vào từng yêu cầu khác nhau. Số lượng các lớp ẩn là không giới hạn. Số lớp ẩn và cách xử lý ở từng lớp kể trên sẽ quyết định kết quả và hiệu quả của công việc cần xử lý.



Hình 2.3 Một mạng nơ ron hai lớp ẩn [4]

Ưu điểm của mạng nơ ron học sâu đó là khả năng giúp giải quyết một loạt các vấn đề phức tạp – chẳng hạn như nhận dạng hình ảnh, ngôn ngữ và lời nói – bằng cách cho phép máy móc tìm hiểu cách các tính năng trong dữ liệu kết hợp thành các dạng trừu tượng ngày càng cao hơn. Tuy nhiên nhược điểm của mạng nơ ron học sâu đó là tốn kém chi phí tính toán và lượng dữ liệu cung cấp rất lớn để có thể đào tạo mạng dự đoán được kết quả mong muốn.

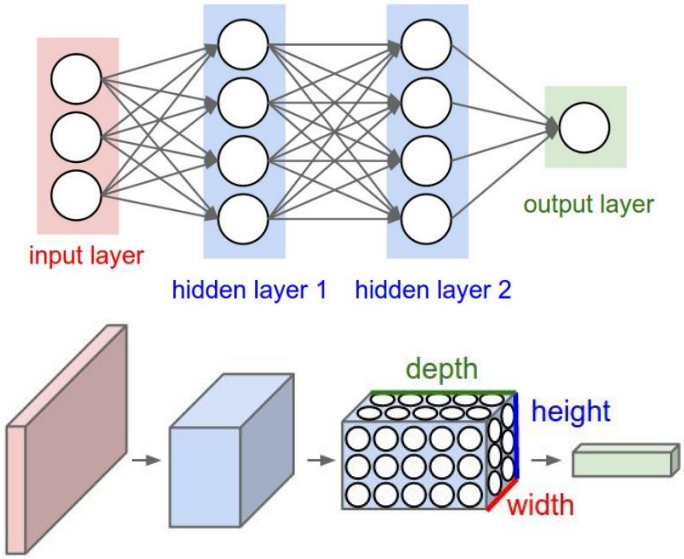
2.1.4 Mạng nơ ron tích chập

Mạng nơ ron tích chập (Convolutional Neural Network - CNN) [5] là một dạng đặc biệt lấy cảm hứng từ mạng nơ ron nhân tạo. Một thay đổi quan trọng của mạng nơ ron tích chập so với mạng nơ ron truyền thống là việc thay vì mỗi nơ ron chỉ nhận một giá trị, mạng nơ ron tích chập sắp xếp các nơ ron thành các không gian đa chiều và xây dựng véc tơ nơ ron dựa trên không gian nơ ron đa chiều này.

Một mạng nơ ron tích chập cũng có cấu trúc lớp cơ bản như mạng nơ ron nhân tạo thông thường, tuy nhiên thay thế lớp ẩn bằng lớp tích chập (convolutional layer) và bổ sung các lớp tổng hợp (pooling layer), lớp dropout (dropout layer) để tăng sự hiệu quả cho mô hình.

Lớp tích chập lấy dữ liệu đầu vào, thực hiện các phép chuyển đổi để tạo ra dữ liệu đầu vào cho lớp kế tiếp (đầu ra của lớp này là đầu vào của lớp sau). Phép biến đổi được sử dụng là phép tính tích chập. Mỗi lớp tích chập chứa một hoặc nhiều bộ lọc - bộ phát hiện đặc trưng (filter - feature detector) cho phép phát hiện và trích xuất những đặc trưng khác nhau của dữ liệu. Cụ thể hơn, một lớp tích chập chứa một tập hợp các bộ lọc có tham số cần phải học. Kích thước của các bộ lọc nhỏ hơn kích thước dữ liệu đầu vào của lớp tích chập. Lớp tích chập sẽ thực hiện phép tích chập giữa các bộ lọc này với khối đặc trưng đầu vào của lớp tích chập đó. Để áp dụng phép tích chập lên toàn bộ khối đặc trưng đầu vào, các bộ lọc sẽ được

"trượt" theo kích thước của khối đặc trưng này, mức độ "trượt" của bộ lọc được định nghĩa bằng giá trị stride. Đầu ra của phép tính toán này cũng là một khối đặc trưng với kích thước được quy định bằng cách thêm vào tham số padding, đây là tham số chỉ định số lượng các giá trị bằng 0 bao quanh khối đặc trưng đầu ra. Theo sau mỗi lớp tích chập là một hàm kích hoạt phi tuyến với mục đích phi tuyến hoá chuyển đổi giữa các lớp trong mạng.



Hình 2.4 So sánh giữa mạng nơ ron thông thường với mạng nơ ron tích chập [6]

Lớp tổng hợp được đặt giữa các lớp tích chập với mục đích giảm dần kích thước không gian của đặc trưng được trích xuất từ lớp tích chập, từ đó giảm số lượng tham số và khối lượng tính toán trong mạng, và do đó kiểm soát được hiện tượng học quá khớp (over fitting) khi mô hình quá phức tạp. Lớp tổng hợp sử dụng một bộ lọc kích thước $k \times k \times c$, tác động lên đặc trưng được trích xuất bằng cách trượt bộ lọc trên khối đặc trưng này và thực hiện phép toán lấy giá trị cực đại (max pooling) hoặc giá trị trung bình (average pooling) của đặc trưng đó.

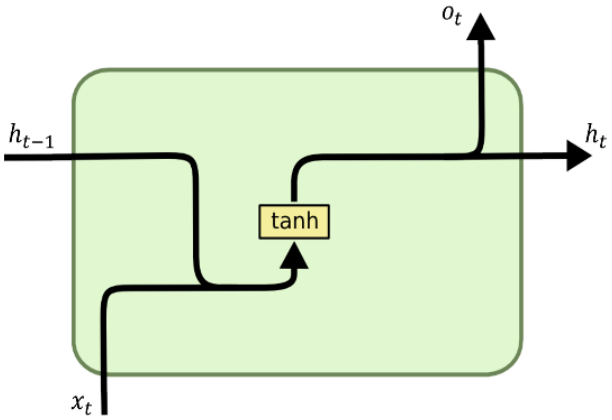
Lớp kết nối đầy đủ sẽ liên kết toàn bộ các nơ ron giữa hai tầng với nhau. Đây là một trong những dạng kết nối thường được sử dụng trong các mạng nơ ron nhân tạo, đặc biệt ở các mạng nơ ron tích chập, lớp kết nối đầy đủ thường được áp dụng ở những tầng cuối trong kiến trúc mạng. Sau các lớp tích chập và tổng hợp thì thường sẽ có hai lớp kết nối đầy đủ, đây là một lớp để tập hợp các đặc trưng mà ta đã tìm ra, đồng thời chuyển đổi dữ liệu đa chiều thành dữ liệu một chiều.

Thuật ngữ dropout [7] đề cập đến việc bỏ qua các đơn vị (unit) trong mạng nơ ron. Hiểu đơn giản là, trong một mạng nơ ron, kỹ thuật dropout là việc sẽ bỏ qua một vài đơn vị trong suốt quá trình huấn luyện trong mô hình, những đơn vị bị bỏ qua được lựa chọn ngẫu nhiên. Ở đây, chúng ta hiểu bỏ qua là đơn vị đó sẽ không tham gia và đóng góp vào quá trình huấn luyện (lan truyền tiến và lan truyền ngược). Khi chúng ta sử dụng lớp kết nối đầy đủ, các nơ ron sẽ phụ thuộc “mạnh” lẫn nhau trong suốt quá trình huấn luyện, điều này làm giảm sức mạng cho mỗi nơ ron và

dẫn đến bị quá khớp trên tập huấn luyện. Dropout loại bỏ bớt sự tham gia của các nơ ron, ép mạng nơ ron phải tìm ra các đặc trưng tốt hơn, đồng thời việc bỏ đi các nơ ron và kết nối giữa chúng cũng làm giảm thời gian huấn luyện cho mô hình.

2.1.5 Mạng nơ ron hồi quy

Mạng nơ ron hồi quy (RNN) [8] một mạng nơ ron nhân tạo trong đó các kết nối giữa các nút tạo thành một đồ thị có hướng dọc theo một trình tự thời gian. Mạng này chứa các vòng lặp bên trong cho phép thông tin có thể lưu lại được. Mô hình mạng nơ ron hồi quy RNN là mô hình được áp dụng rất rộng rãi trong các bài toán xử lý ngôn ngữ tự nhiên do mô hình RNN mô hình hóa được bản chất dữ liệu trong ngôn ngữ có đặc tính chuỗi và có sự phụ thuộc lẫn nhau giữa các thành phần (trạng thái) trong dữ liệu.

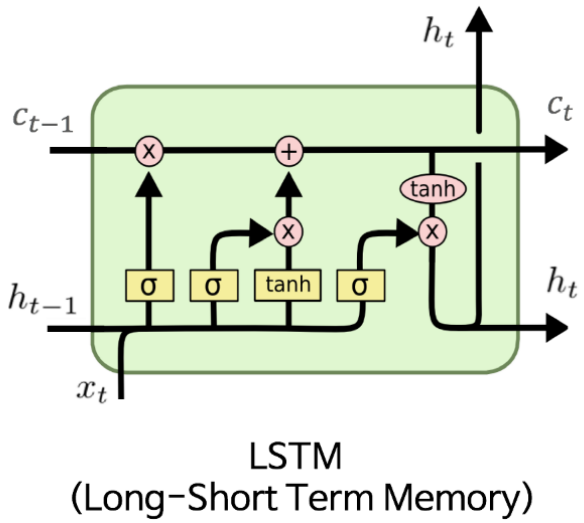


Hình 2.5 Một đơn vị nơ ron trong mạng nơ ron hồi quy [9]

Trong mạng RNN, các kết nối giữa các nút tạo thành một đồ thị có hướng dọc theo một chuỗi thời gian. Hình 2.5 mô tả một cấu trúc một nút mạng nơ ron hồi quy: x_t là đầu vào tại bước tính toán *t*. o_t là đầu ra tại bước *t*. h_t là trạng thái ẩn tại bước *t*, nó chính là bộ nhớ của mạng. h_t được tính toán dựa trên cả các trạng thái ẩn phía trước và đầu vào tại bước đó, tạo khả năng ghi nhớ các thông tin đã được tính toán ở những bước thời gian trước cho mạng.

RNN được gọi là hồi quy bởi lẽ chúng thực hiện cùng một tác vụ cho tất cả các phần tử của một chuỗi với đầu ra phụ thuộc vào cả các phép tính trước đó. Nói cách khác, RNN có khả năng nhớ các thông tin được tính toán trước đó. Trên lý thuyết RNN có thể sử dụng được thông tin của một văn bản rất dài, tuy nhiên thực tế thì nó chỉ có thể nhớ được một vài bước trước đó mà thôi.

Mạng bộ nhớ dài-ngắn (Long Short-Term Memory), thường được gọi là LSTM [10] là một dạng đặc biệt của RNN, được thiết kế để giải quyết các bài toán về phụ thuộc xa trong mạng RNN do bị ảnh hưởng bởi vấn đề đạo hàm (gradient) biến mất.

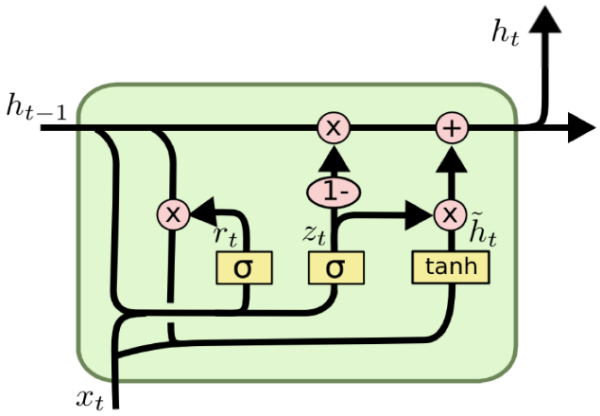


LSTM
(Long-Short Term Memory)

Hình 2.6 Nơ ron LSTM [9]

Ý tưởng của LSTM là bổ sung thêm trạng thái bên trong tế bào (cell internal state) và ba cổng sàng lọc thông tin đầu vào và đầu ra cho tế bào bao gồm cổng quên (forget gate) có nhiệm vụ loại bỏ những thông tin không cần thiết nhận được khỏi trạng thái trong tế bào, cổng vào (input gate) có nhiệm vụ chọn lọc những thông tin cần thiết nào được nhận vào trạng thái bên trong tế bào và cổng ra (output gate) có nhiệm vụ xác định những thông tin nào từ trạng thái bên trong tế bào được sử dung như đầu ra. Mạng LSTM có thể bao gồm nhiều tế bào LSTM liên kết với nhau.

GRU (Gated Recurrent Unit) là một phiên bản của LSTM với nguyên tắc hoạt động tương tự như LSTM nhưng có cấu tạo đơn giản hơn. GRU kết hợp trạng thái bên trong tế bào và trạng thái ẩn thành một, do đó nó chỉ có hai đầu vào và một đầu ra. Ngoài ra, GRU sử dụng hai cổng là cổng thiết lập lại (reset gate) và cổng cập nhật (update gate) để quyết định việc lưu trữ và loại bỏ thông tin.



GRU
(Gated Recurrent Unit)

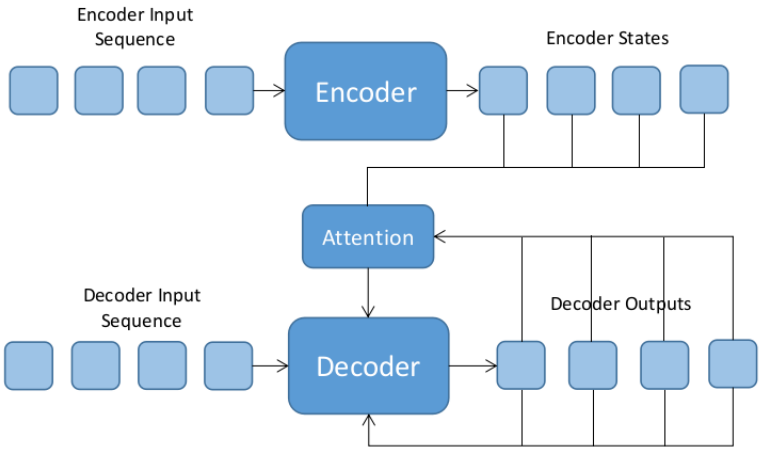
Hình 2.7 Nơ ron GRU [9]

Cả LSTM và GRU đều có những ưu nhược điểm của riêng mình. Thông thường LSTM có thể lưu trữ thông tin với dữ liệu dài hơn so với GRU. Tuy nhiên do cấu tạo đơn giản của mình, GRU thường xử lý nhanh hơn LSTM và có thể dễ dàng sử dụng để xây dựng các mạng có cấu trúc phức tạp. Do đó, việc sử dụng LSTM và GRU tùy thuộc vào từng yêu cầu bài toán cụ thể.

2.1.6 Mạng nơ ron tuần tự

Mạng học sâu là mô hình học máy rất mạnh mẽ, đạt được hiệu suất tuyệt vời đối với các vấn đề khó khăn như nhận diện hình ảnh. Tuy nhiên, các mạng này chỉ có thể được áp dụng cho các bài toán mà đầu vào và mục tiêu được mã hóa thành các vector phù hợp có chiều cố định, do đó xử lý không tốt cho các bài toán với đầu vào là các chuỗi có độ dài không biết trước như bài toán dịch máy hay tổng hợp giọng nói. Để giải quyết vấn đề này, mô hình mạng sequence-to-sequence ra đời.

Mô hình tuần tự (sequence-to-sequence) gồm hai phần chính là bộ mã hóa (Encoder) và bộ giải mã (Decoder). Cả hai thành phần này đều được hình thành từ các mạng nơ ron. Bộ mã hóa có nhiệm vụ chuyển đổi chuỗi dữ liệu đầu vào (input sequence) thành một biểu diễn với số chiều thấp cố định hay còn gọi là vectơ ngữ cảnh. Bộ giải mã có nhiệm vụ tạo ra đầu ra (output sequence) từ biểu diễn được tạo ra từ bộ mã hóa. Do bộ giải mã nhìn thấy cả chuỗi biểu diễn cố định sinh ra từ chuỗi đầu lẫn chuỗi đích, nên nó có thể sinh ra những dự đoán thông minh hơn về các đầu ra tương lai dựa trên đầu ra hiện tại. Trong mô hình sequence-to-sequence, ta có thể sử dụng các kiến trúc mạng khác nhau cho tầng mã hóa và giải mã như mạng RNN hay CNN.



Hình 2.8 Kỹ thuật attention

Mô hình sequence-to-sequence có thể được mở rộng để sử dụng kĩ thuật Attention để mô hình có thể tập trung vào các phần khác nhau của chuỗi đầu vào ở những thời điểm dự đoán khác nhau. Một vấn đề trong mô hình sequence-to-sequence đó là có hiệu năng kém với các chuỗi đầu vào và đầu ra dài bởi các biểu diễn bên trong có kích thước cố định trong lớp mã hóa. Hơn nữa, trạng thái ẩn cuối cùng của lớp mã hóa - trạng thái được sử dụng làm đầu vào của bộ giải mã, chứa phần lớn thông tin từ những phần tử cuối của lớp mã hóa, do đó nó có thể bị mất mát

thông tin từ những phần tử ở đầu. Kỹ thuật Attention được thêm vào để giải quyết vấn đề hạn chế này. Thay vì nén toàn bộ chuỗi đầu vào thành một vectơ ngữ cảnh cố định, kĩ thuật Attention sẽ lưu giữ toàn bộ các trạng thái từ bộ mã hoá và đưa cho từng phần tử của bộ giải mã giá trị trọng số trung bình của các trạng thái mã hoá. Ban đầu tất cả các trạng cuối cùng của chuỗi mã hoá đầu vào đều được giữ lại. Trong suốt quá trình giải mã chúng ta sẽ lấy trạng thái của mạng giải mã kết hợp với trạng thái của bộ mã hoá và truyền vào mạng truyền thẳng. Mạng này sẽ trả về danh sách các trọng số cho từng trạng thái mã hóa. Đầu vào mã hóa được nhân với các trọng số sau đó tính trung bình có trọng số của các trạng thái mã hóa. Kết quả ngữ cảnh này sau đó sẽ được chuyền đến lớp giải mã. Mạng giải mã bây giờ có thể sử dụng các phần khác nhau của chuỗi giải mã trong quá trình sinh chuỗi giả mã thay vì chỉ sử dụng một véc tơ ngữ cảnh cố định. Điều này cho phép mạng tập trung vào những phần quan trọng nhất của chuỗi đầu vào thay vì toàn bộ chuỗi đầu vào, do đó tạo ra các dự đoán thông minh hơn cho trạng thái tiếp theo trong chuỗi giải mã.

2.1.7 Chuẩn hóa theo lô

Chuẩn hóa theo lô (Batch normalization) là một kỹ thuật để cải thiện tốc độ, hiệu suất và tính ổn định của mạng nơ ron tích chập.

Mỗi lớp của mạng nơ ron tích chập có các đầu vào có phân phối tương ứng, phân phối này sẽ bị ảnh hưởng trong quá trình huấn luyện mô hình bởi tính ngẫu nhiên trong khởi tạo tham số và tính ngẫu nhiên trong dữ liệu đầu vào. Ảnh hưởng của các yếu tố ngẫu nhiên này đến phân phối đầu vào được gọi là hiện tượng dịch chuyển hiệp phương sai (covariate shift). Một cách đơn giản, nó là sự thay đổi giá trị trung bình (mean) và phương sai (variance) của phân phối. Trong quá trình huấn luyện của các mạng, khi các tham số của các lớp trước thay đổi, việc phân phối các đầu vào cho lớp hiện tại thay đổi theo, do đó lớp hiện tại cần phải điều chỉnh liên tục đối với các bản phân phối mới. Vấn đề này đặc biệt nghiêm trọng đối với các mạng sâu, bởi vì những thay đổi nhỏ trong các lớp ẩn nông hơn sẽ được khuếch đại khi chúng lan truyền trong mạng, dẫn đến sự thay đổi đáng kể trong các lớp ẩn sâu hơn. Do đó, batch normalization được đề xuất để giảm những thay đổi không mong muốn này, giúp làm tăng tốc độ huấn luyện và tăng độ chính xác cho mô hình. Bên cạnh việc giảm ảnh hưởng của hiện tượng dịch chuyển hiệp phương sai, batch normalization được cho là mang lại nhiều lợi ích khác. Việc thêm vào mạng lớp batch normalization, mạng có thể sử dụng tốc độ học (learning rate) cao hơn mà không gặp phải hiện tượng đạo hàm triệt tiêu (vanishing gradient).

Trong quá trình huấn luyện, chúng ta thường chia tập dữ liệu thành các phần có số lượng dữ liệu bằng nhau, mỗi tập này được gọi là một Batch. Về mặt toán học, batch normalization giúp chuẩn hoá phân phối của một batch về một phân phối mới với giá trị trung bình bằng 0 và độ lệch chuẩn bằng 1.

Quá trình thực hiện của batch normalization như sau đây.

– Với một batch dữ liệu X gồm *n* điểm dữ liệu $a = (a_1, a_2, ..., a_n)$, ta có giá trị trung bình và độ lệch chuẩn của phân phối dữ liệu được tính bằng:

$$\mu_x = \frac{1}{n}\sum_{i=1}^{n} a_i \qquad \textit{PT 2.2}$$

$$\sigma_x^2 = \frac{1}{n}\sum_{i=1}^{n}(a_i - \mu_x)^2 \qquad \textit{PT 2.3}$$

– Chuẩn hoá X về giá trị trung bình 0 và độ lệch chuẩn 1 bằng:

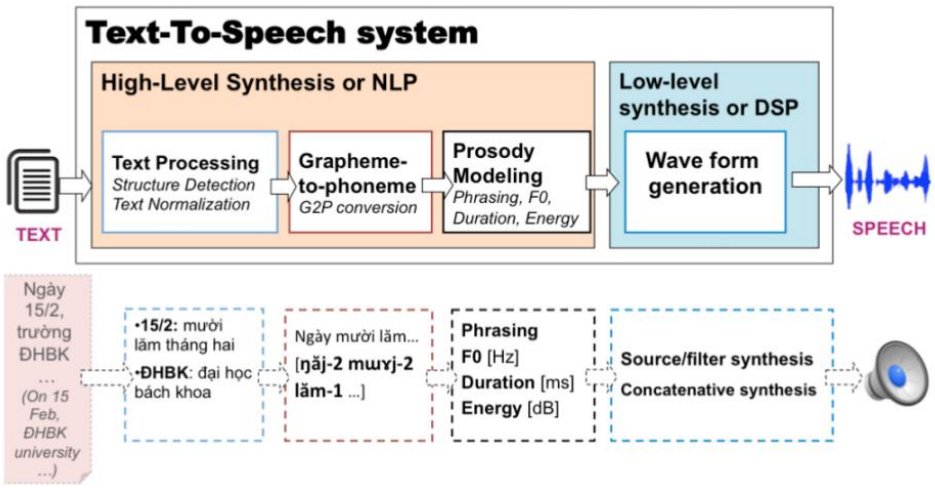
$$\hat{a}_i = \frac{a_i - u_X}{\sqrt{\sigma_X^2 + \epsilon}} \qquad \textit{PT 2.4}$$

– Để mạng có thể linh hoạt trong việc lựa chọn phân phối, tham số học được γ và β được thêm vào, hai tham số này cho phép mạng thay đổi phân phối tuỳ vào mục đích.

$$y_i = \gamma\hat{a_i} + \beta = BN_{\gamma,\beta}(a_i) \qquad \textit{PT 2.5}$$

2.2 Tổng quan về tổng hợp tiếng nói

Qua quá trình phát triển, đa số các hệ thống tổng hợp tiếng nói đều bao gồm hai khối chức năng. Bao gồm khối phân tích xử lý ngôn ngữ tự nhiên và khối xử lý tổng hợp tiếng nói [11]. Hình 2.9 là ví dụ về một hệ thống tổng hợp tiếng nói bao gồm hai khối chức năng này.



Hình 2.9 Sơ đồ chức năng tổng quát một hệ thống tổng hợp tiếng nói [12]

Khối xử lý văn bản chịu trách nhiệm chuẩn hóa văn bản và xử lý cách phát âm của các kí tự đầu vào. Tiếng nói ở dạng sóng tín hiệu sẽ được tạo ra bằng một kỹ thuật tổng hợp ở khối xử lý tổng hợp tiếng nói.

2.2.1 Khối xử lý ngôn ngữ tự nhiên

Khối xử lý ngôn ngữ tự nhiên có nhiệm vụ chuyển đổi chuỗi các ký tự văn bản đầu vào thành một dạng chuỗi các nhãn ngữ âm đã được thiết kế trước của hệ thống tổng hợp tiếng nói. Nghĩa là chuyển đổi chuỗi văn bản đầu vào thành dạng biểu diễn ngữ âm, xác định cách đọc nội dung văn bản. Quá trình này cũng đòi hỏi khả năng dự đoán ngôn điệu từ văn bản đầu vào với thông tin ngữ âm và ngữ điệu tương ứng. Thông tin ngữ âm cho biết những âm nào sẽ được phát âm ra trong ngữ cảnh cụ thể nào. Thông tin ngữ điệu mô tả điệu tính của các âm được phát âm [11]. Quy trình xử lý ngôn ngữ tự nhiên thường bao gồm ba bước:

- Xử lý và chuẩn hóa văn bản (Text Processing).
- Phân tích cách phát âm: chuyển đổi hình vị sang âm vị (Grapheme to phoneme).
- Phát sinh các thông tin ngôn điệu, ngữ âm cho văn bản (Prosody modeling).

Chuẩn hóa văn bản có nhiệm vụ chuyển hóa văn bản đầu vào ban đầu thành một văn bản dạng chuẩn, có thể đọc được một cách dễ dàng. Có thể kể đến như chuyển đổi các số, từ viết tắt, ký tự đặc biệt, … thành dạng viết đầy đủ và chính xác.

Phân tích cách phát âm là quá trình xác định cách phát âm chính xác cho từ. Các hệ thống tổng hợp tiếng nói dùng hai cách cơ bản để xác định cách phát âm cho từ ngữ, quá trình này còn được gọi là chuyển đổi văn bản sang chuỗi âm vị. Phương pháp thứ nhất và đơn giản nhất là dựa vào từ điển. Sử dụng một từ điển lớn có chứa tất cả các từ vựng của một ngôn ngữ và chứa cách phát âm đúng tương ứng cho từng từ. Cách phát âm đúng cho từng từ được xác định bằng cách tìm kiếm trong từ điển và thay đoạn văn bản bằng chuỗi âm vị tương ứng trong từ điển. Phương pháp thứ hai là dựa trên các quy tắc ngôn ngữ. Sử dụng các quy tắc ngôn ngữ để tìm ra cách phát âm tương ứng. Mỗi phương pháp đều có ưu nhược điểm khác nhau, phương pháp dựa trên từ điển nhanh và chính xác, nhưng hệ thống sẽ không hoạt động nếu từ phát âm không có trong từ điển, và cần lưu một số lượng từ vựng lớn. Phương pháp sử dụng quy tắc phát âm phù hợp cho tất cả các văn bản nhưng nếu ngôn ngữ có nhiều trường hợp bất quy tắc có thể làm tăng độ phức tạp.

Quá trình tạo ra các thông tin ngôn điệu cho văn bản là việc xác định vị trí trọng âm của từ cần phát âm, sự lên xuống của giọng nói ở các vị trí khác nhau trong câu và xác định các biến thể khác nhau của âm phụ thuộc vào ngữ cảnh khi được phát âm liên tục. Ngoài ra quá trình này còn phải xác định các điểm dừng nghỉ lấy hơi khi phát âm hoặc đọc một đoạn văn bản [13]. Thông tin thời gian được đo bằng đơn vị mili giây (ms), và được ước tính bằng cách sử dụng các quy tắc hoặc thuật toán học máy.

2.2.2 Khối tổng hợp tín hiệu tiếng nói

Khối xử lý tổng hợp tiếng nói có nhiệm vụ tạo ra tiếng nói từ thông tin về ngữ âm, ngữ điệu do khối xử lý ngôn ngữ tự nhiên cung cấp. Trong thực tế có hai cách tiếp cận cơ bản liên quan đến công nghệ tổng hợp tiếng nói: tổng hợp tiếng nói sử dụng mô hình nguồn âm và tổng hợp dựa trên việc ghép nối các đơn vị âm.

Chất lượng giọng nói tổng hợp được đánh giá qua hai tiêu chí. Đó là tính dễ hiểu và tính tự nhiên. Tính dễ hiểu có nghĩa nội dung của tiếng nói tổng hợp có dễ hiểu hay không. Tính tự nhiên của tiếng nói tổng hợp là sự so sánh độ giống nhau giữa giọng nói tổng hợp và giọng nói tự nhiên của con người.

Hệ thống tổng hợp tiếng nói được đánh giá lý tưởng cần phải vừa dễ hiểu vừa tự nhiên. Mục tiêu của việc xây dựng hệ thống tổng hợp tiếng nói là tối đa hóa hai khía cạnh này [14].

2.3 Các phương pháp tổng hợp tiếng nói

2.3.1 Tổng hợp mô phỏng hệ thống phát âm

Tổng hợp mô phỏng hệ thống phát âm là công nghệ tổng hợp giọng nói được mô phỏng trên máy tính có thể mô phỏng cơ quan phát âm của con người và quá trình tạo ra âm thanh của nó. Vì mục tiêu của phương pháp này là mô phỏng quá trình tạo tiếng nói sao cho càng giống cơ chế của con người càng tốt, nên về mặt lý thuyết đây được xem là phương pháp tổng hợp tiếng nói cơ bản nhất, nhưng cũng vì vậy mà phương pháp này khó thực hiện nhất và khó có thể tổng hợp được tiếng nói chất lượng cao. Do những hạn chế trong vấn đề mô phỏng các tham số tiếng nói và năng lực tính toán, mà tổng hợp mô phỏng hệ thống phát âm đã không đạt được nhiều thành công mong đợi như các phương pháp tổng hợp tiếng nói khác. Để thực hiện được phương pháp tổng hợp mô phỏng hệ thống phát âm đòi hỏi thời gian, chi phí và công nghệ. Phương pháp này khó có thể ứng dụng tại Việt Nam thời điểm hiện nay.

2.3.2 Tổng hợp tần số formant

Tổng hợp tiếng nói formant là phương pháp tổng hợp tiếng nói không sử dụng mẫu giọng thật nào khi chạy, thay vào đó tín hiệu tiếng nói được tạo ra bởi một mô hình tuyến âm. Mô hình này mô phỏng hiện sự cộng hưởng của các cơ quan phát âm thông qua một bộ gồm nhiều bộ lọc. Các bộ lọc này gọi là các bộ lọc cộng hưởng formant, có thể được kết hợp nối tiếp, song song hoặc kết hợp cả hai.

Trong quá trình tổng hợp âm thanh giọng nói, phương pháp tổng hợp formant không cần trực tiếp sử dụng bất kỳ mẫu giọng thực nào. Thay vào đó, tín hiệu âm thanh được tổng hợp dựa trên một mô hình âm thanh. Các thông số như tần số cơ bản, sự phát âm và âm lượng được thay đổi theo thời gian để tạo ra tín hiệu giọng nói nhân tạo. Tuy nhiên, phương pháp này vẫn cần âm thanh giọng nói thực ở bước phân tích để có được đặc trưng formant, trường độ hay năng lượng tiếng nói.

Hiện nay, với những công cụ thích hợp chúng ta hoàn toàn có thể xác định tần số formant cho các âm vị của tiếng Việt. Phương pháp này có ưu điểm là tiết kiệm được bộ nhớ, có khả năng điều khiển mềm dẻo các tham số âm học của tiếng nói. Nhược điểm của phương pháp này là khó xây dựng, cần nghiên cứu sâu sắc về ngữ âm của ngôn ngữ, phức tạp trong việc xác định các tham số điều khiển bộ tổng hợp, hạn chế về tính tự nhiên, độ giống tiếng người của tiếng nói tạo ra, chất lượng tiếng nói không tự nhiên (nói nghe như tiếng robot, khác hoàn toàn giọng nói con người) và phụ thuộc nhiều vào chất lượng của quá trình phân tích tiếng nói của từng ngôn ngữ. Ngoài ra, tổng hợp formant yêu cầu chuẩn bị trước các tham số chính xác trước khi tiến hành tổng hợp tiếng nói, khiến cho quá trình tổng hợp thiếu linh hoạt.

Tại Việt Nam, phương pháp tổng hợp formant cũng đã có vài công trình nghiên cứu và đã có các kết quả đưa vào ứng dụng thực tế. Chẳng hạn, phần mềm "đọc văn bản tiếng Việt" [15] hay Phần mềm tổng hợp tiếng nói tiếng Việt VnSpeech [16] tổng hợp tiếng nói theo hướng tiếp cận này. Hệ thống tổng hợp formant có thể đọc được hầu hết các âm tiết tiếng Việt ở mức nghe rõ, tuy vậy, nó có nhược điểm là mức độ tự nhiên không cao.

2.3.3 Tổng hợp ghép nối

Tổng hợp ghép nối (hay còn gọi là lựa chọn đơn vị âm) là một trong số các phương pháp tổng hợp mới phát triển sau này, kết hợp (ghép nối) các mẫu tiếng nói tự nhiên thu âm sẵn lại với nhau để tạo ra câu nói tổng hợp. Các âm tiết sau khi được tạo thành sẽ được tiếp tục ghép lại với nhau tạo thành đoạn tiếng nói. Vì các thuộc tính tự nhiên của tiếng nói được lưu trữ trong các đơn vị âm, nên phương pháp này có khả năng tổng hợp tiếng nói có độ dễ hiểu và độ tự nhiên cao. Tuy nhiên, giọng nói tự nhiên được ghi âm có sự thay đổi từ lần phát âm này sang lần phát âm khác, và công nghệ tự động hóa việc ghép nối các đoạn của sóng âm thỉnh thoảng tạo ra những tiếng cọ xát không tự nhiên ở phần ghép nối. Ngoài ra, tập các đơn vị âm luôn bị hạn chế về số lượng cũng như nội dung. Điều này dẫn đến tiếng nói tổng hợp nghe "thô ráp", các đơn vị âm ghép nối với nhau thường không phù hợp ngữ cảnh. Không chỉ vậy, cần phải có một không gian lưu trữ rất lớn kèm theo tốc độ tính toán và truy vấn đủ mạnh để có thể lưu trữ tất cả các đơn vị âm cần thiết cho nhiều người nói khác nhau, với nhiều ngữ cảnh và đặc trưng trạng thái. Điều này không hiệu quả về mặt kinh tế. Hạn chế này khiến tính linh hoạt của tổng hợp ghép nối bị ảnh hưởng và phương pháp này chỉ có thể "bắt chước" một giọng người nói cụ thể trong tập dữ liệu đơn vị âm rất lớn của người đó. Do hạn chế về chất lượng của tiếng nói tổng hợp dựa vào formant, nên phương pháp tổng hợp ghép nối được tập trung đầu tư, nghiên cứu.

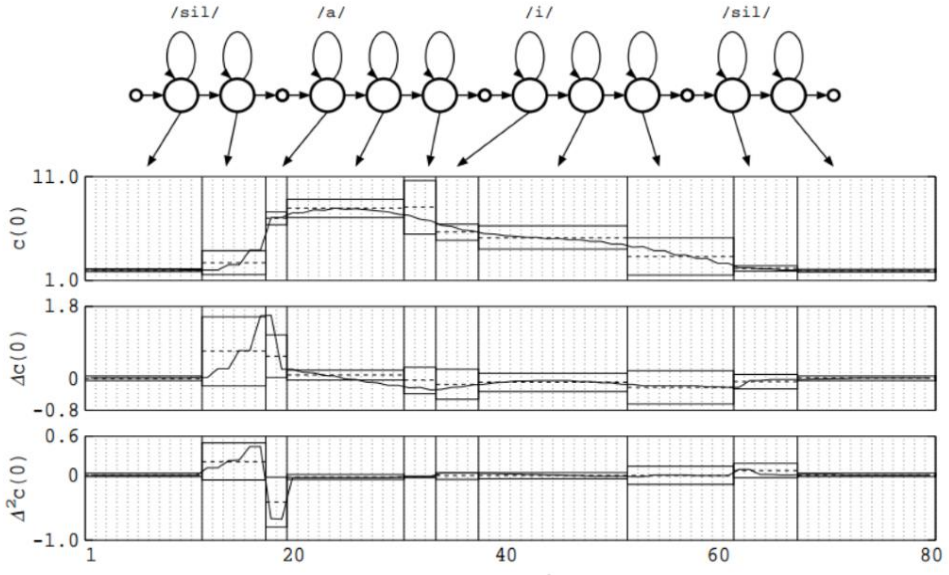
Có ba kiểu tổng hợp ghép nối, cụ thể là:

- Tổng hợp chuyên biệt (Domain-specific).
- Tổng hợp âm kép (diphone).

- Tổng hợp chọn đơn vị (unit selection).

2.3.4 Tổng hợp dùng tham số thống kê (HMM)

Phương pháp tổng hợp tiếng nói dùng tham số thống kê HMM là phương pháp dựa trên mô hình Markov ẩn (HMM) [17]. Ở đây, HMM là một mô hình thống kê, được sử dụng để mô hình hóa các tham số tiếng nói của một đơn vị ngữ âm, trong một ngữ cảnh cụ thể, được trích rút đồng thời từ cơ sở dữ liệu tiếng nói. Nhờ tập các HMM này, hệ thống sau đó có thể phát sinh ra các tham số tiếng nói, tuỳ thuộc vào nội dung văn bản đầu vào, để tạo ra tiếng nói dưới dạng sóng nhờ các tham số được phát xạ này.

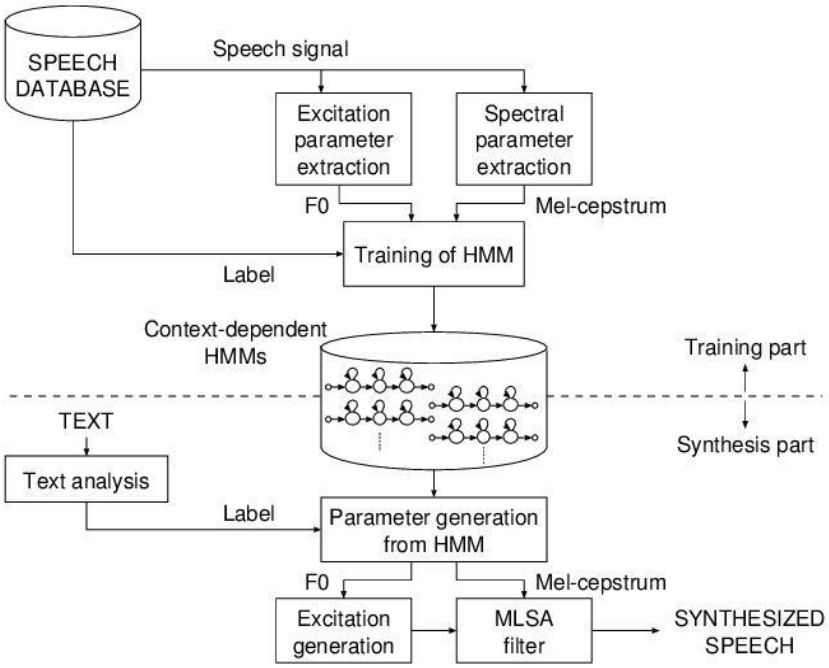


Hình 2.10 Ví dụ về thống kê và các tham số HMM cấp câu [18]

Hình 2.10 mô tả cách áp dụng mô hình Markov ẩn trong tổng hợp tiếng nói, trong đó mỗi mô hình Markov ẩn được sử dụng để mô hình hóa một âm vị, và các mô hình Markov ẩn được móc nối với nhau để mô hình hóa chuỗi âm vị, các tham số HMM cấp câu được tạo bằng cách ghép hai HMM cấp âm vị (cụ thể là âm vị /a/ và /i/).

Mô hình Markov ẩn là một mô hình học máy dựa trên thống kê, do đó hệ thống tổng hợp tiếng nói dựa trên mô hình Markov ẩn hoạt động bao gồm hai quá trình là quá trình huấn luyện và quá trình tổng hợp. Hình 2.11 mô tả quá trình tổng hợp và huấn luyện một hệ thống tổng hợp tiếng nói dựa trên mô hình Markov ẩn.

Quá trình tổng hợp dựa trên mô hình Markov ẩn sẽ là quá trình mà nhận đầu vào là một đoạn văn bản, chuyển hóa đoạn văn bản này thành chuỗi âm vị, sau đó dựa vào các mô hình markov ẩn mô hình hóa chuỗi các âm vị tương ứng ta sẽ tìm ra được các tham số mel và tần số cơ bản F0. Từ các tham số mel xây dựng nên chuỗi các bộ lọc MLSA (Mel Log Spectral Approximation) và kết hợp với tín hiệu kích thích được tạo từ F0 sẽ tạo ra được tín hiệu tiếng nói [19] [20].



Hình 2.11 Quá trình huấn luyện và tổng hợp một hệ thống tổng hợp tiếng nói dựa trên mô hình markov ẩn [21]

Trong quá trình huấn luyện, trước tiên các tham số phổ (ví dụ như các hệ số mel-cepstral) và tham số kích thích (ví dụ như tần số cơ bản F0) được trích xuất từ dữ liệu tiếng nói mẫu. Sau đó các tham số đã được trích xuất được mô hình hóa bằng các mô hình HMM phụ thuộc ngữ cảnh. Mô hình trường độ phụ thuộc ngữ cảnh cũng được tính toán trong giai đoạn này. Các HMM phụ thuộc ngữ cảnh dùng để mô hình hoá phổ và F0 được huấn luyện dùng một kỹ thuật gom cụm dựa trên cây quyết định dùng tiêu chí độ dài mô tả cực tiểu (minimum description length - MDL).

Hệ thống tổng hợp tiếng nói dựa trên mô hình Markov ẩn có khả năng tạo ra các mẫu tiếng nói với các phong cách khác nhau, mang đặc điểm của người nói khác nhau, thậm chí mang cảm xúc của người nói. Ưu điểm của phương pháp này là cần ít bộ nhớ lưu trữ và tài nguyên hệ thống hơn so với tổng hợp ghép nối, có thể điều chỉnh tham số để thay đổi ngữ điệu. Tuy nhiên, một số nhược điểm của hệ thống này đó là độ tự nhiên trong tiếng nói tổng hợp của hệ thống bị suy giảm hơn so với tổng hợp ghép nối, phổ tín hiệu và tần số cơ bản được ước lượng từ các giá trị trung bình của các mô hình Markov ẩn được huấn luyện từ dữ liệu khác nhau, điều này khiến cho tiếng nói tổng hợp nghe có vẻ đều đều mịn và không được tự nhiên.

2.3.5 Tổng hợp bằng phương pháp lai ghép

Tổng hợp tiếng nói bằng phương pháp lai ghép là phương pháp tổng hợp bằng cách kết hợp giữa tổng hợp ghép nối chọn đơn vị và tổng hợp dựa trên mô hình Markov

ẩn, nhằm tận dụng ưu điểm của từng phương pháp và áp dụng trong hệ thống. Như đã nói, hệ thống tổng hợp lai ghép kết hợp ưu nhược điểm của từng hệ thống thành phần, tùy theo thành phần nào đóng vai trò chủ đạo mà có thể phân loại các hệ thống tổng hợp lai ghép thành hai loại sau: Tổng hợp hướng ghép nối và tổng hợp hướng HMM.

Hệ thống tổng hợp hướng ghép nối sử dụng các HMM để hỗ trợ quá trình ghép nối, ý tưởng chính của phương pháp này như sau: Đơn vị dùng để lựa chọn trong tổng hợp ghép nối chọn đơn vị cũng sẽ là đơn vị được tổng hợp ra. Đường biên giữa các đơn vị sẽ được làm mịn bằng các mô hình markov ẩn. Âm thanh sau cùng được làm mịn bằng phương pháp làm mịn phổ.

Khác với hệ thống tổng hợp hướng ghép nối, hệ thống tổng hợp hướng HMM sử dụng các thuật toán sinh tham số từ các HMM và phần tổng hợp ghép nối được sử dụng để tăng cường chất lượng chuỗi tham số này.

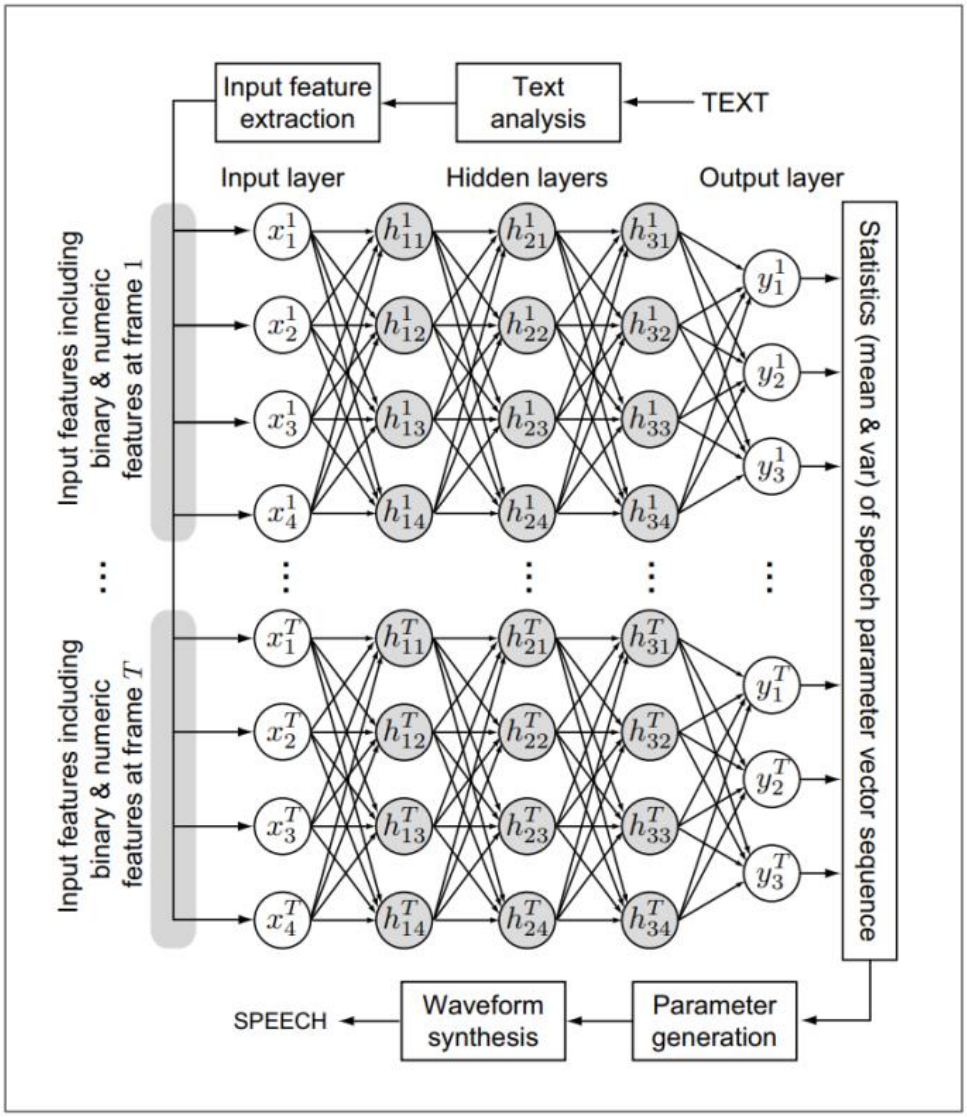
Hai hướng tổng hợp lai ghép nêu trên đều có ưu nhược điểm khác nhau, và được sử dụng tùy vào yêu cầu chất lượng tiếng nói hay yêu cầu cụ thể về hệ thống. Ưu điểm cơ bản của hệ thống lai ghép hướng ghép nối đó là giảm tác động không mong muốn do dữ liệu không đủ và giảm sự phụ thuộc vào dữ liệu, hay cũng chính là cải thiện các nhược điểm của tổng hợp ghép nối. Mặc dù đã giải quyết cơ bản những vấn đề về ghép nối nhưng vấn đề trở ngại tại những điểm ghép nối vẫn còn tồn tại.

2.3.6 Tổng hợp bằng học sâu

Tổng hợp tiếng nói dựa trên mạng nơ ron nhân tạo đã phát triển mạnh mẽ trong vài năm trở lại đây. Phương pháp này dựa trên việc sử dụng mạng nơ ron học sâu DNN để mô hình hóa mô hình âm học. Trong đó văn bản đầu vào sẽ được chuyển hóa thành một véc tơ đặc trưng ngôn ngữ các véc tơ đặc trưng này mang các thông tin về âm vị, ngữ cảnh xung quanh âm vị, thanh điệu,…Sau đó mô hình âm học dựa trên DNN lấy đầu vào là véc tơ đặc trưng ngôn ngữ và tạo ra các đặc trưng âm học tương ứng ở đầu ra. Từ các đặc trưng âm học này sẽ tạo thành tín hiệu tiếng nói nhờ một bộ tổng hợp tín hiệu tiếng nói (thường là vocoder).

Mô hình âm học được sử dụng có thể là mạng nơ ron học sâu (DNN), mạng nơ ron tích chập (CNN), mạng nơ ron hồi tiếp (RNN) hay mạng bộ nhớ dài-ngắn hạn (LSTM). Mô hình thông dụng nhất là DNN. Hình 2.12 mô tả một kiến trúc tổng quan của một hệ thống tổng hợp tiếng nói dựa trên mạng nơ ron học sâu DNN. Trong đó, văn bản cần tổng hợp sẽ trải qua quá trình phân tích văn bản (Text analysis) để trích xuất các đặc trưng ngôn ngữ và được chuyển đổi thành các véc tơ đặc trưng bởi bộ trích xuất đặc trưng (Input feature extraction). Các véc tơ này được đưa vào mạng theo từng khung (frame), đầu ra của mạng sẽ là các đặc trưng. Các đặc trưng đầu ra này chứa các thông tin về phổ và tín hiệu kích thích, thông qua bộ tạo tham số (Parameter Generation) sẽ được chuyển thành các tham số đặc

trưng âm học và được đưa qua bộ tạo tín hiệu tiếng nói (Waveform generation) để tạo ra tín hiệu tiếng nói thực.



Hình 2.12 Một mô hình tổng hợp tiếng nói dựa vào mạng nơ ron học sâu. [22]

Đến thời điểm hiện tại, một mô hình học sâu mới hơn là sequence-to-sequence (seq2seq) đang dần thay thế cho các mô hình DNN truyền thống. Năm 2014 mạng seq2seq lần đầu tiên được giới thiệu bởi nhóm nghiên cứu của Google [23] . Mặc dù ban đầu của mô hình này chỉ được ứng dụng trong bài toán dịch máy, tuy nhiên hiện tại mô hình seq2seq cũng được áp dụng nhiều trong nhiều bài toán khác như nhận diện giọng nói, tóm tắt văn bản, chú thích video, hỏi đáp, tổng hợp tiếng nói.

Trong tổng hợp tiếng nói, các hệ thống sử dụng mô hình seq2seq cho thấy chất lượng giọng nói tổng hợp có độ tự nhiên gần tương đương giọng nói thật của con người, vượt trội hơn hẳn so với các cách tổng hợp cũ. Kết quả đánh giá ở Bảng 2.1 cho thấy rằng mô hình Tacotron 2 [24] đạt độ tự nhiên cao nhất trong các mô hình tổng hợp tiếng nói sử dụng cùng bộ dữ liệu huấn luyện North American English, gần tương đương với giọng nói tự nhiên.

Bảng 2.1 Đánh giá một số mô hình tổng hợp tiếng nói trên bộ dữ liệu North American English.

Mô hình	MOS
Giọng nói tự nhiên	4.582 ± 0.053.
Tacotron 2 [24]	4.526 ± 0.066.
Wavenet (Linguistic) [24]	4.210 ± 0.081.
Wavenet (L+F) [25]	4.341 ± 0.051.
HMM-driven concatenative [25]	3.860 ± 0.137.
LSTM-RNN parametric [25]	3.670 ± 0.098.

Ưu điểm của các hệ thống tổng hợp tiếng nói dựa trên phương pháp học sâu đó chính là độ tự nhiên của giọng nói tổng hợp. Đặc biệt trong trường hợp sử dụng mạng nơ ron học sâu có thể mô hình hóa một cách mạnh mẽ mối quan hệ phi tuyến, phức tạp giữa các đặc trưng ngôn ngữ học của văn bản và đặc trưng âm học của tín hiệu tiếng nói. Tuy nhiên việc sử dụng mạng nơ ron học sâu cũng có những hạn chế đó là vì sự mạnh mẽ của nó nên nó rất nhạy cảm với các thông tin sai lệch và không tốt như nhiễu, và nó cũng cần rất nhiều dữ liệu để huấn luyện mô hình, cùng với đó là thời gian huấn luyện mất hàng chục tiếng thậm chí hàng tuần và yêu cầu cao về hiệu năng máy tính. Do đó để xây dựng những hệ thống tổng hợp tiếng nói dựa trên mạng nơ ron học sâu rất tốn kém về chi phí.

CHƯƠNG 3. HƯỚNG TIẾP CẬN

3.1 Tình hình phát triển về tổng hợp tiếng nói tiếng Việt

Việt Nam đang trong thời kỳ phát triển nhanh chóng của công nghệ thông tin. Điều đó cho phép chúng ta có những nền tảng khoa học kỹ thuật và nền tảng cơ sở vật chất để có thể nghiên cứu cũng như triển khai các ứng dụng về khoa học công nghệ trong cuộc sống. Trong nhiều năm trở lại đây, tổng hợp tiếng Việt đã có những thành tựu đáng kể, các hệ thống tổng hợp tiếng nói tiếng Việt được ra đời như VietVoice, VnSpeech, Vais, hệ thống tổng hợp tiếng nói của tập đoàn FPT hay hệ thống tổng hợp tiếng nói Hoa súng. Trong đó các hệ thống tổng hợp tiếng nói tiếng Việt được xây dựng dựa theo các hướng phổ biến đó là tổng hợp ghép nối, tổng hợp sử dụng tham số thống kê HMM và tổng hợp dựa trên mạng nơ ron học sâu.

Về phương pháp tổng hợp tiếng nói ghép nối cho tiếng Việt, rất nhiều hệ thống đã được phát triển. Chẳng hạn như hệ thống Hoa súng [26], được phát triển lần đầu tiên năm 2007. Dữ liệu để xây dựng hệ thống này có tên là VNSpeech Corpus, được thu thập và chắt lọc từ nhiều nguồn khác nhau, chẳng hạn như truyện, sách, … Dữ liệu này bao gồm nhiều loại khác nhau như các từ với đầy đủ sáu thanh điệu, các số, câu thoại, đoạn văn ngắn, … Đến năm 2011 hệ thống được mở rộng, sử dụng kỹ thuật lựa chọn âm vị không đồng nhất. Phiên bản này cũng sử dụng bộ dữ liệu tương tự ở phiên bản trước. Kết quả ban đầu cho thấy phiên bản thứ hai của hệ thống Hoa súng có sự cải thiện về mặt chất lượng, tuy nhiên dữ liệu kiểm thử không được thiết kế để bao trùm toàn bộ đơn vị âm, thêm nữa không có sự kết nối giữa quá trình chọn đơn vị âm và quá trình chọn đơn vị như một bán âm tiết trong việc tính toán chi phí mục tiêu và chi phí ghép nối. Kết quả là tổng chi phí không được tối ưu hóa cho những câu cần bán âm tiết.

Đối với phương pháp tổng hợp tiếng nói sử dụng tham số thống kê, tại Việt Nam cũng đã có nhiều hệ thống tổng hợp tiếng nói phát triển dựa trên phương pháp này. Có thể kể đến như sản phẩm của tập đoàn FPT hay hệ thống tổng hợp tiếng nói tiếng Việt Mica TTS[4] (Viện Mica Đại học Bách Khoa Hà Nội).

Trong những năm gần đây, nhiều hệ thống tổng hợp tiếng nói dựa trên mạng nơ ron học sâu được ra đời. Tại Việt Nam, các mô hình này cũng nhanh chóng được ứng dụng vào tổng hợp giọng nói tiếng Việt. Điển hình có thể kể đến sản phẩm tổng hợp giọng nói Viettel AI[5], sản phẩm trợ lý ảo của Zalo[6] hay hệ thống tổng hợp tiếng nói của Google[7]. Những hệ thống này có độ tư nhiên cao cũng như được

[4] http://sontinh.mica.edu.vn/tts2
[5] https://viettelgroup.ai/service/tts
[6]https://kiki.zalo.ai/
[7] https://assistant.google.com/

ứng dụng rộng rãi như báo nói (Dân trí [8], Báo mới [9], …), trợ lý ảo (Kiki, Google Assistant) hay tổng đài trả lời tự động (Viettel Cyber Callbot[10]).

3.2 Hướng tiếp cận

Với sự phát triển của các kỹ thuật tổng hợp tiếng nói, các yêu cầu phát triển của hệ thống tổng hợp tiếng nói cũng liên tục tăng lên. Ngoài tính tự nhiên, người ta cũng mong đợi hệ thống tổng hợp tiếng nói có thể tạo ra giọng nói của bất kỳ người nói nào với dữ liệu đào tạo tối thiểu. Trước vấn đề đặt ra này, thích ứng giọng nói đã trở thành một trong những hướng nghiên cứu trong lĩnh vực tổng hợp tiếng nói. Thích ứng giọng nói là kỹ thuật hệ thống tổng hợp tiếng nói được điều chỉnh các tham số của một mô hình cho phù hợp với đặc điểm âm thanh của một người nói bằng cách sử dụng một lượng ít mẫu âm thanh của người nói đó. Đây không phải là một chủ đề mới mà là một chủ đề đã được nghiên cứu kỹ lưỡng từ lâu, đặc biệt là trên hệ thống tổng hợp tiếng nói dựa trên mô hình Markov ẩn.

Các hệ thống tổng hợp giọng nói dựa trên mô hình Markov ẩn đã được phát triển bằng những kỹ thuật thích ứng tiếng nói khác nhau nhằm cải thiện tính tự nhiên và tính dễ hiểu của giọng nói tổng hợp. Các kỹ thuật có thể chia thành hai loại gồm MLLR và MAP. Nhóm các kỹ thuật MLLR (Maximum likelihood linear regression) cố gắng tìm hiểu một phép biến đổi tuyến tính có thể thay đổi giọng nói trung bình thành âm thanh giọng nói mục tiêu. Các kỹ thuật MAP (Maximum a posteriori) sử dụng các mô hình độc lập với người nói (speaker-independent) làm mô hình mẫu trước khi ước tính mô hình giọng mục tiêu. Các kỹ thuật thích ứng tiếng nói này đã được chứng minh là có hiệu quả trong việc sử dụng một lượng nhỏ dữ liệu thích ứng để bắt chước giọng nói của con người. Tuy nhiên các hệ thống theo phương pháp không đảm bảo về phong độ giọng nói thích ứng bởi sự ảnh hưởng của trạng thái của mô hình ban đầu hay các tiêu chí ước tính. Với tiếng Việt đã có một số hệ thống thích ứng giọng nói được nghiên cứu ví dụ mô hình thích ứng giọng nói dựa trên mô hình HMM.

Đối với các hệ thống dựa trên mạng nơ ron học sâu, phương pháp phổ biến được sử dụng để đào tạo một mô hình thích ứng giọng nói là dùng một véc tơ mã hóa người nói là một phương pháp được sử dụng phổ biến. Mô hình Deep Voice thêm véc tơ mã hóa người nói vào nhiều phần của mạng để tạo ra mô hình nhiều người nói và thích ứng giọng nói cho người nói mới. Mô hình Voiceloop sử dụng mô hình âm học để huấn luyện một bộ mã hóa người nói có thể dùng mẫu âm thanh cũng như nhãn người nói để tạo ra giọng nói thích ứng.

Trong tiếng Việt đã có một số hệ thống thích ứng giọng nói được nghiên cứu, chẳng hạn như mô hình thích ứng giọng nói dựa trên mô hình dựa trên mô hình

[8] https://dantri.com.vn/
[9] https://baomoi.com/
[10] https://viettelgroup.ai/product/cyberbot

Markov ẩn. Các hệ thống tổng hợp tiếng nói tiếng Việt bằng học sâu mới phát triển gần đây nên chưa có nhiều nghiên cứu về chủ đề thích ứng giọng nói.

Kỹ thuật thích ứng giọng nói đã có nhiều phương pháp được sử dụng, ví dụ như: học chuyển tiếp, sử dụng véc tơ mã hóa người nói, học đóng góp đơn vị ẩn (Learning hidden unit contribution - LHUC), biến đổi không gian đặc trưng (Feature space transformation - FST). Trong phần này sẽ giới thiệu hai hướng tiếp cận đó là học chuyển tiếp và sử dụng véc tơ mã hóa người nói.

Đồ án này hướng đến cách tiếp cận tách mô hình người nói khỏi mô hình tổng hợp giọng nói bằng cách đào tạo độc lập mạng nhúng phân biệt người nói, từ đó nắm bắt không gian của đặc điểm người nói và đào tạo một mô hình tổng hợp giọng nói trên một tập dữ liệu dựa trên cách biểu diễn người nói được học bởi mạng nhận dạng người nói.

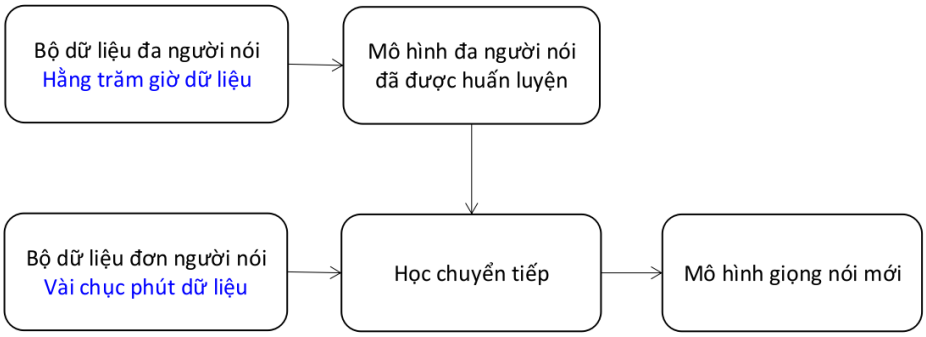
3.2.1 Học chuyển tiếp

Học chuyển tiếp là một cách tiếp cận trong học máy, giải quyết một nhiệm vụ cụ thể bằng cách sử dụng toàn bộ hoặc một phần của mô hình đã được huấn luyện trước về một nhiệm vụ khác. Học chuyển tiếp giúp giảm thời gian đào tạo mô hình đồng thời cũng giúp nâng cao hiệu suất cho quá trình đào tạo mô hình mới. Ví dụ, trong bài toán tổng hợp tiếng nói, một mô hình biến đổi các đặc trưng âm học thành giọng nói được đào tạo cho một ngôn ngữ có thể được sử dụng cho bài toán tổng hợp tiếng nói của ngôn ngữ khác khi chỉ cần đào tạo lại trên một lượng nhỏ dữ liệu hoặc không cần đào tạo lại. Kỹ thuật học chuyển tiếp có thể mang lại các lợi ích như:

- Giải quyết vấn đề thiếu dữ liệu: Các mô hình học sâu đòi hỏi rất nhiều dữ liệu để giải quyết một nhiệm vụ hiệu quả. Tuy nhiên, không thường xuyên có quá nhiều dữ liệu có sẵn. Trong trường hợp đó, một tác vụ mục tiêu cụ thể có thể được giải quyết bằng cách sử dụng lại một mô hình được đào tạo trước cho một nhiệm vụ ban đầu tương tự.
- Giải quyết vấn đề về tốc độ: Học chuyển tiếp cắt giảm một tỷ lệ lớn thời gian đào tạo và cho phép xây dựng các giải pháp học sâu khác nhau ngay lập tức.
- Tăng hiệu năng bắt đầu: Mô hình có thể đạt hiệu năng cao hơn khi sử dụng kiến trúc đã được huấn luyện trước thay vì phải khởi tạo các tham số một cách ngẫu nhiên từ đầu.
- Tăng khả năng hội tụ cho mô hình đào tạo.

Có hai cách tiếp cận phổ biến cho quá trình học chuyển tiếp. Phương pháp đầu tiên là chỉ cập nhật một phần các trọng số của mô hình cũ trong quá trình đào tạo mô hình thích ứng. Việc điều chỉnh có thể chỉ được cập nhật ở các lớp ẩn hoặc chỉ trong lớp đầu ra. Phương pháp thứ hai là trong quá trình huấn luyện mô hình thích ứng trên tập dữ liệu mới, toàn bộ trọng số của mô hình cũ sẽ được cập nhật. Phương pháp này kỳ vọng rằng việc sử dụng mô hình cũ làm điểm khởi đầu cho mô hình

thích ứng có thể cải thiện tốc độ hội tụ và chất lượng của mô hình thích ứng. Đây là hướng tiếp cận phổ biến trong học sâu.



Hình 3.1 Phương pháp học chuyển tiếp

Do sự phức tạp của mô hình cũng như số lượng tham số quá lớn, đồ án này hướng đến cách tiếp cận thứ nhất: Hệ thống tổng hợp giọng nói được điều chỉnh cho phù hợp với các đặc điểm âm thanh của một người nói cụ thể bằng cách sử dụng một mẫu nhỏ giọng nói của người dùng đó. Trước tiên mô hình được huấn luyện trên tập dữ liệu nhiều người nói. Mô hình sẽ tiếp tục được đào tạo trên tập dữ liệu giọng nói mục tiêu sau khi hoàn thành quá trình đào tạo trên tập dữ liệu nhiều người nói để có được mô hình cho giọng nói thích ứng. Cách tiếp cận này hiệu quả hơn về dữ liệu so với việc đào tạo một mô hình TTS riêng biệt cho mỗi người nói, ngoài ra còn nhanh hơn và ít tốn kém hơn về mặt tính toán.

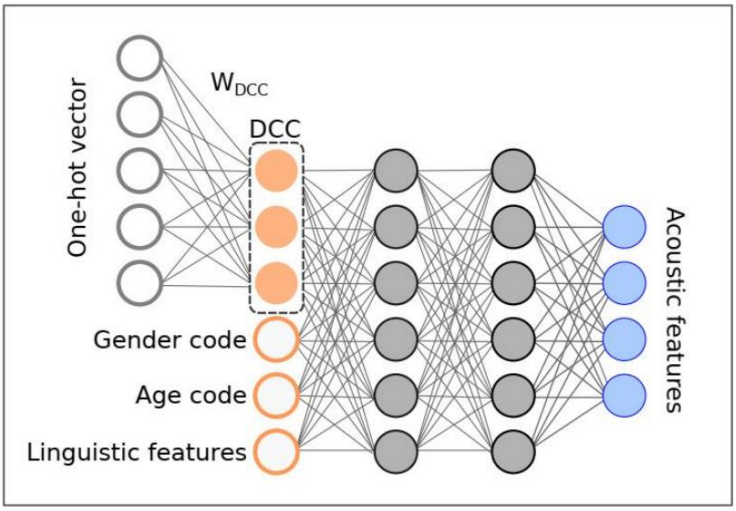
3.2.2 Véc tơ mã hóa người nói

Sử dụng véc tơ mã hóa người nói là một phương pháp đã được sử dụng từ lâu trong bài toán nhận dạng tiếng nói. Phương pháp này được xác nhận về tính hiệu quả đáng kể trong việc xác minh người nói độc lập với văn bản. Véc tơ mã hóa người nói là một véc tơ sử dụng các đặc trưng đầu vào để xác định người nói, mỗi một người nói có một véc tơ duy nhất. Từ ý tưởng này, véc tơ mã hóa người nói cũng được sử dụng trong bài toán tổng hợp tiếng nói.

Để mã hóa các đặc trưng của một người nói có nhiều cách thức thực hiện. Phương án đơn giản nhất là sử dụng one-hot vec tơ. Véc tơ one-hot sẽ có dạng $X = [x_1, x_2, \ldots, x_n]$. Trong đó n là số lượng người nói trong tập dữ liệu. Với mỗi câu nói trong tập dữ liệu, x_i bằng 0 nếu đó không phải là câu nói của người nói thứ i và ngược lại, bằng 1 nếu người thứ i là người nói. Véc tơ này sẽ được thêm vào véc tơ các đặc trưng đầu vào khác hoặc được kết nối trực tiếp với các lớp ẩn.

Một phương pháp khác đó là sử dụng véc tơ nhúng, từ mô hình DNN ban đầu sẽ được thêm một tập các nút (node) để mã hóa thông tin người nói. Các nút này nhận đầu vào (input) dưới dạng các véc tơ one-hot mã hóa người nói, đầu ra (output) được kết nối với các lớp ẩn khác của mô hình DNN. Các trọng số của các nút đặc trưng người nói sẽ được cập nhật đồng thời với các trọng số của mô hình DNN trong quá trình huấn luyện. Tại quá trình thích ứng, véc tơ nhúng S sẽ được tính toán nhờ thuật toán lan truyền ngược. Một cách khác để trích rút véc tơ nhúng là

sử dụng mạng DNN phân loại người nói. Véc tơ thu được từ mô hình phân loại người nói đã huấn luyện sẽ được trích ra từ trước lớp phân loại cuối cùng và được sử dụng này để làm véc tơ mã hóa người nói. Bên cạnh véc tơ mã hóa người nói, các thông tin về người nói như độ tuổi, giới tính, quê quán, … cũng được thêm vào véc tơ đặc trưng nhằm tăng hiệu quả học tập cho mô hình.



Hình 3.2 Các thông tin về tuổi và giới tính được thêm vào cùng với véc tơ mã hóa người nói [27]

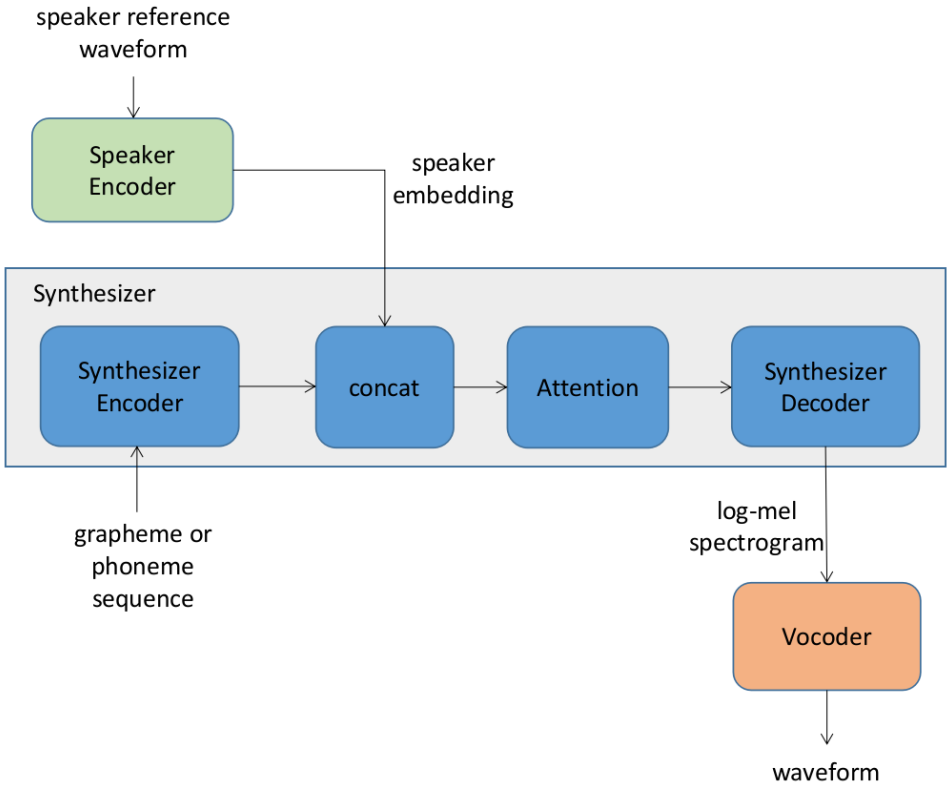
Trong đồ án này, véc tơ mã hóa người nói được sử dụng là véc tơ đặc trưng người nói được trích xuất từ nhiệm vụ xác nhận người nói.

CHƯƠNG 4. PHƯƠNG PHÁP ĐỀ XUẤT

Được xây dựng dựa trên kiến trúc "Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis" [28]: một hệ thống dựa trên mạng nơ ron để tổng hợp văn bản thành giọng nói (TTS) có thể tạo ra âm thanh giọng nói bằng giọng nói của những người nói khác nhau. Phát triển từ ý tưởng này, đồ án hướng đến việc xây dựng mô hình tổng hợp giọng nói một người dựa trên mô hình đa người nói bằng phương pháp học chuyển tiếp.

Hệ thống gồm 3 thành phần được đào tạo độc lập, trong đó các bước tương ứng với các mô đun được liệt kê theo thứ tự:

- Speaker encoder (Mạng mã hóa người nói): dựa trên nhiệm vụ xác nhận người nói (speaker verification), từ giọng nói tạo ra tham chiếu (embedding vector) có số chiều cố định cho người nói tương ứng. Tham chiếu này là một biểu diễn có ý nghĩa giọng của người nói, trong cùng một không gian biểu diễn, các giọng nói giống nhau sẽ được biểu diễn gần nhau.
- Synthesizer (Bộ tổng hợp giọng nói): dựa trên mô hình Tacotron 2, kết hợp tham chiếu giọng nói và dữ liệu văn bản, tạo ra mel-spectrogram.
- Vocoder: tổng hợp mẫu tiếng nói dạng sóng theo miền thời gian từ các đặc trưng học được từ synthesizer, tạo ra âm thanh từ mel-spectrogram mang giọng của người nói tương ứng.

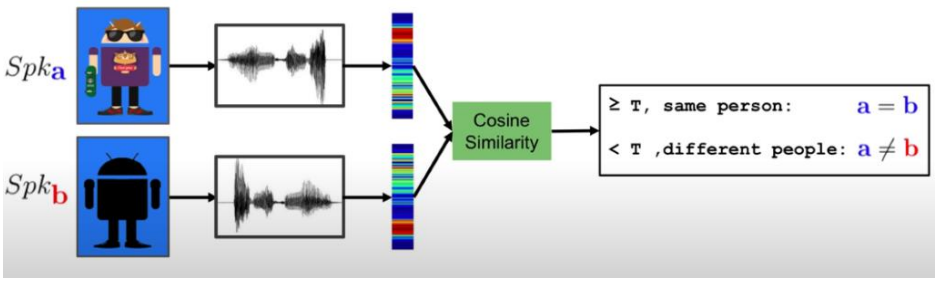


Hình 4.1 Kiến trúc mô hình

4.1 Speaker encoder

Mạng mã hóa người nói (speaker encoder) ánh xạ một chuỗi các khung biểu đồ log-mel spectrogram được tính từ một đầu vào âm thanh có thời lượng tùy ý, đến một véc tơ nhúng có chiều cố định, được gọi là véc tơ biểu diễn đặc trưng người nói [29], không phụ thuộc vào nội dung âm thanh và tiếng ồn xung quanh. Mạng được đào tạo để tối ưu việc véc tơ nhúng các lời nói phát ra từ cùng một người nói có độ giống nhau cao, trong khi các lời nói phát ra từ các người nói khác nhau ở xa nhau trong không gian nhúng. Mặc dù mạng không được tối ưu hóa trực tiếp để học cách biểu diễn nắm bắt các đặc điểm của người nói liên quan đến tổng hợp tiếng nói, nhưng việc đào tạo nhiệm vụ phân biệt người nói dẫn đến véc tơ nhúng phù hợp để điều chỉnh mạng tổng hợp dựa trên nhận dạng của người nói.

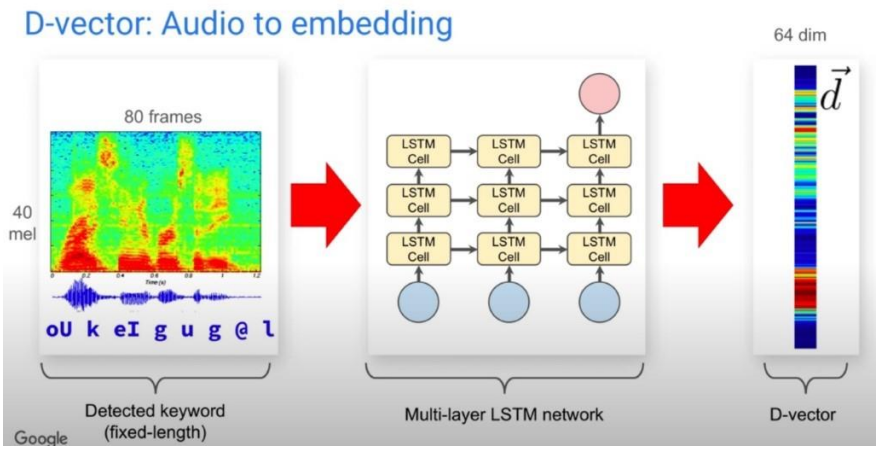
Mạng mã hóa người nói được đào tạo dựa trên nhiệm vụ xác thực người nói. Xác thực người nói là quá trình xác minh xem lời nói có thuộc về một người nói cụ thể hay không, dựa trên các phát biểu đã biết của người nói đó. Bản mẫu để nhận dạng của một người được lấy bằng cách lấy véc tơ nhúng từ một câu nói của họ. Tại thời điểm xác minh, người nói tự nhận dạng mình bằng một câu nói ngắn và hệ thống sẽ so sánh véc tơ nhúng của câu nói đó với các véc tơ nhúng đã biết (lời nói đã đăng ký) của người đó. Người nói được xác thực qua một ngưỡng tương đồng nhất định.



Hình 4.2 Phương pháp xác nhận người nói [30]

Các bước tính toán véc tơ nhúng đặc trưng người nói được thực hiện như sau:

Đầu vào của mạng là các khung phổ log-mel 40 kênh (channels) với độ rộng cửa sổ 25ms (window width) và bước 10ms (step) được đi qua một mạng gồm một chồng 3 lớp LSTM với 768 nút ẩn (hidden layers), lớp LSTM cuối cùng được kết nối với một lớp biến đổi tuyến tính ReLU như một phép biến đổi bổ sung của khung cuối cùng của mạng lên 256 đơn vị. Sau đó đầu ra của lớp này được chuẩn hóa L2 - normalizing tạo thành vecto nhúng 256 phần tử.



Hình 4.3 Biểu diễn một mẫu giọng nói thành véc tơ nhúng số chiều cố định [30]

Hàm mất mát generalized end-to-end (GE2E) [30] được sử dụng để tối ưu hóa mô hình.

Tại thời điểm huấn luyện, một batch được tạo từ $N \times M$ câu. Những câu nói này từ N người nói khác nhau và mỗi người nói có M câu nói. Mỗi véc tơ đặc trưng x_{ji} ($1 \leq j \leq N$ và $1 \leq i \leq M$) đại diện cho các đặc trưng được trích xuất từ câu nói i của người nói j.

Đưa vectơ đặc trưng x_{ji} qua mạng LSTM. Đầu ra của toàn bộ mạng nơ ron là $f(x_{ji}; w)$ trong đó w đại diện cho tất cả các tham số của mạng nơ ron. Véc tơ nhúng được định nghĩa là chuẩn hóa L2 của đầu ra mạng: e_{ji} là đại diện cho vectơ nhúng của câu nói i của người nói j.

$$e_{ji} = \frac{f(x_{ji};\omega)}{\left\|f(x_{ji};\omega)\right\|_2} \qquad PT\ 4.1$$

Tâm của các véc tơ nhúng từ người nói $[e_{j1}, \ldots, e_{jM}]$ được định nghĩa là c_j:

$$c_k = E_m[e_{km}] = \frac{1}{M}\sum_{m=1}^{M} e_{km} \qquad PT\ 4.2$$

Ma trận tương đồng $S_{ji,k}$ được định nghĩa là các độ tương đồng cosine được chia tỷ lệ giữa mỗi véc tơ nhúng e_{ji} và tất cả các tâm c_k ($1 \leq j, k \leq N$ và $1 \leq i \leq M$), trong đó w và b là các tham số có thể học được:

$$S_{ji,k} = w \cdot cos(e_{ji}, c_k) + b \qquad PT\ 4.3$$

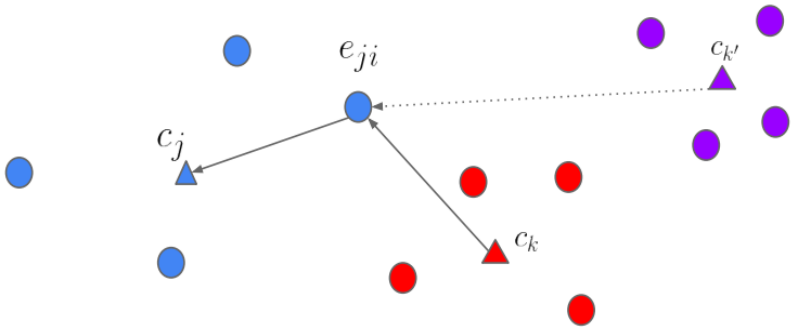
Trong quá trình đào tạo, mục tiêu hướng đến là véc tơ nhúng của mỗi câu nói ở gần với phần trung tâm của tất cả vectơ nhúng của người nói đó, đồng thời cách xa với phần trung tâm của những người nói khác. Để tối ưu hóa theo hướng này, đặt hàm softmax trên $S_{ji,k}$với $k = 1, \ldots, N$ để đầu ra bằng 1 nếu $k = j$, ngược lại đầu ra bằng 0. Do đó, tổn thất trên mỗi véc tơ nhúng e_{ji} có thể được định nghĩa là:

$$L(e_{ji}) = -S_{ji,j} + \log \sum_{k=1}^{N} \exp\left(S_{ji,k}\right). \qquad PT\ 4.4$$

Hàm mất mát này đẩy mỗi véc tơ nhúng gần với tâm của nó và kéo nó ra khỏi tất cả các tâm khác. Hàm mất mát GE2E cuối cùng L_G là tổng của tất cả các tổn thất trên ma trận tương tự ($1 \le j \le N$ và $1 \le i \le M$):

$$L_G(x; w) = L_G(S) = \sum_{j,i} L(e_{ji}) \qquad PT\ 4.5$$

Hình 4.4 mô tả biểu diễn các véc tơ nhúng trong quá trình huấn luyện: véc tơ nhúng màu xanh lam gần với tâm của người nói của chính nó (tam giác xanh lam) và xa các tâm khác (tam giác đỏ và tím), đặc biệt là tâm gần nhất (tam giác đỏ).



Hình 4.4 Biểu diễn các véc tơ nhúng đặc trưng người nói của các người nói khác nhau trong quá trình huấn luyện [30]

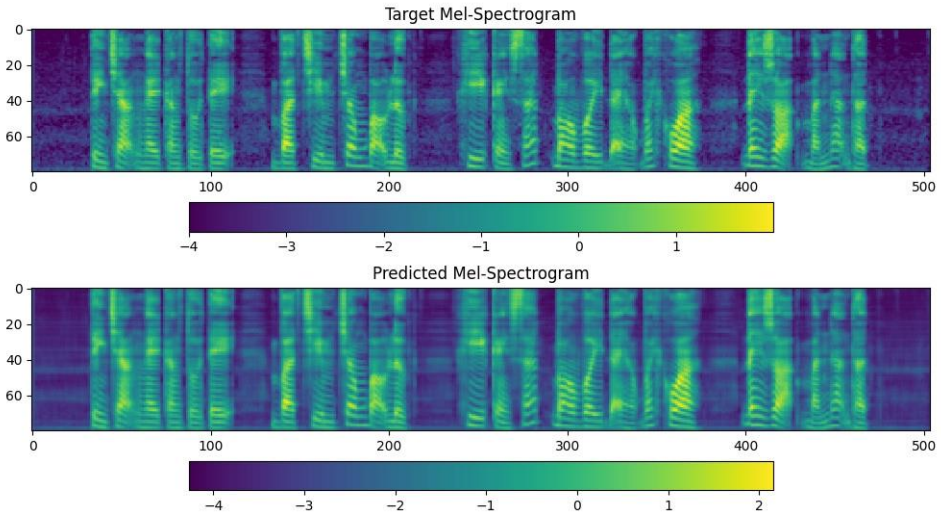
Dữ liệu véc tơ nhúng đặc trưng người nói mới sau quá trình này sẽ được gắn vào cùng dữ liệu văn bản để huấn luyện dự đoán biểu đồ âm phổ được mô tả tại Hình 4.8.

4.2 Synthesizer

Synthesizer là mạng tổng hợp có nhiệm vụ dự đoán mel-spectrogram từ văn bản đầu vào.

4.2.1 Kiến trúc synthesizer trong mô hình Tacotron 2 truyền thống

Tacotron 2 [31] là một mô hình end-to-end sequence to sequence cho hệ thống tổng hợp tiếng nói trực tiếp từ các ký tự. Mô hình có thể huấn luyện hoàn toàn chỉ từ các cặp dữ liệu <văn bản, âm thanh> mà không cần trích rút đặc trưng hay phải huấn luyện riêng biệt nhiều mô đun. Tacotron 2 sử dụng một cặp mạng nơ ron với vai trò khác nhau: một mạng tạo ra mel spectrogram - hình ảnh trực quan về những tần số âm thanh cụ thể, mạng còn lại (vocoder) sẽ tái hiện lại những spectrogram đó dưới dạng âm thanh. Phương pháp này sử dụng đặc trưng âm thanh mức thấp: mel-spectrogram, làm cầu nối giữa 2 thành phần (Seq2Seq và Vocoder) phía trên.

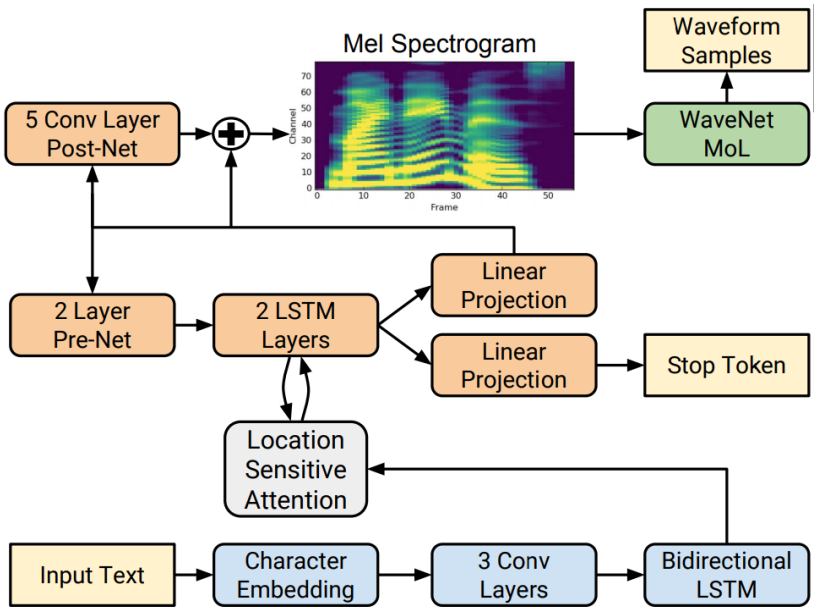


Hình 4.5 Biểu diễn mel-spectrogram của một đoạn âm thanh thật và âm thanh tổng hợp tương ứng cùng văn bản

Việc sử dụng một biểu diễn dễ dàng tính toán từ các dạng sóng miền thời gian cho phép huấn luyện hai thành phần riêng biệt. Biểu diễn mel-spectrogram cũng mượt mà hơn so với các mẫu dạng sóng và dễ huấn luyện hơn bằng cách sử dụng tổn thất lỗi bình phương (mean square error) vì nó không thay đổi theo pha trong mỗi khung hình. Mel-Spectrogram được tính toán thông qua các phép biến đổi phi tuyến tính trên đặc trưng phổ tần số tuyến tính, như một dạng tóm tắt tần số với số chiều nhỏ hơn. Đồng thời, tập trung hơn vào các tần số thấp giúp lời nói dễ hiểu hơn, còn các tần số cao thường được tạo ra từ âm xát và tiếng ồn, những phần không cần phải mô hình hóa. Đây cũng là một đặc trưng cơ bản được dùng trong các hệ thống nhận dạng tiếng nói.

Về cơ bản, Tacotron 2 sử dụng mô hình seq2seq được tối ưu hóa cho TTS để ánh xạ một chuỗi các chữ cái thành chuỗi các tính năng mã hóa âm thanh. Các tính năng này được biểu diễn dưới dạng một quang phổ âm thanh 80 chiều với các khung hình được tính toán sau mỗi 12,5 mili giây, không chỉ ghi lại cách phát âm

của các từ mà còn ghi lại các nét đặc trưng khác nhau trong lời nói của con người, bao gồm âm lượng, tốc độ và ngữ điệu. Cuối cùng, các tính năng này được chuyển đổi thành dạng sóng 16kHz bằng việc sử dụng kiến trúc WaveNet chỉnh sửa.



Hình 4.6 Kiến trúc mô hình Tacotron 2 với WaveNet vocoder [31]

Cấu trúc Synthesizer trong mô hình Tacotron 2 mang kiến trúc Seq2Seq chịu trách nhiệm chuyển đổi các văn bản thành các bản đồ phổ mel-spectrogram, gồm 3 phần: encoder, attention và decoder.

Bộ mã hóa Encoder có nhiệm vụ chuyển đổi một chuỗi ký tự thành một biểu diễn tính năng ẩn, gồm các phần:

- Character Embedding với nhiệm vụ mã hóa các ký tự riêng lẻ từ chuỗi văn bản dưới dạng véc tơ. Kích thước của mạng tuỳ thuộc vào số lượng từ có trong từ điển.
- 3 Conv Layers: kết quả đầu ra của mạng embeding sẽ được đưa vào 3 lớp Convolution 1D và mỗi lớp trong số đó chứa 512 bộ lọc kích thước 5 x 1 (đây là kích thước bộ lọc tốt trong ngữ cảnh này vì nó nắm bắt một ký tự nhất định, cũng như hai kí tự kế trước đó và hai kí tự kế tiếp theo) và sau cùng là lớp Batch Normalization và hàm kích hoạt ReLU. Các tầng tích chập này dùng để mô hình hóa ngữ cảnh dài (long-term context) của chuỗi ký tự đầu vào.
- Bidirectional LSTM: Đầu ra của lớp tích chập cuối cùng được đưa vào một mạng LSTM hai chiều chứa 512 nút (256 nút cho mỗi chiều) để sinh ra các đặc trưng được mã hóa.

Cơ chế Location Sensitive Attention giúp cho mô hình tập trung vào không chỉ các đặc trưng ở các bước trước đó mà còn là cả đặc trưng tại vị trí hiện tại, giúp biểu diễn được độ chính xác của phát âm của các từ và giữ lại các đặc trưng của âm thanh con người như âm lượng, tốc độ và ngữ điệu. Các biểu diễn đầu ra của Encoder sau khi được ghép với Speaker Embedding được mạng Attention sử dụng

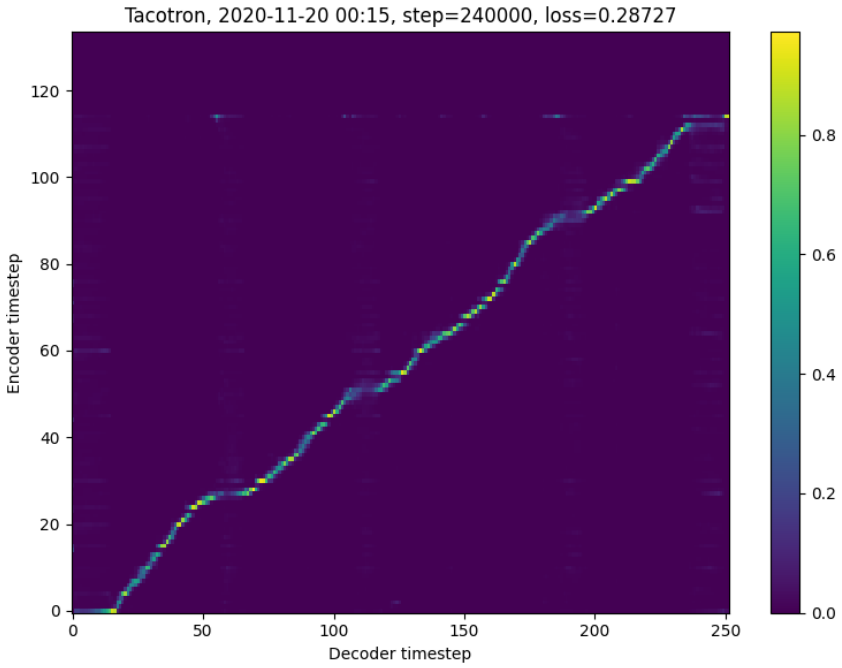
để rút gọn lại thành một véc tơ ngữ cảnh với chiều dài cố định cho mỗi bước ở Decoder là Attention context vector.

Bộ giải mã Decoder sử dụng chuỗi biểu diễn tính năng ẩn để dự đoán một biểu đồ âm phổ, là một mạng nơ ron tự hồi quy (autoregressive recurrent neural network) với các đặc điểm:

- Đầu ra của decoder từ bước trước được đưa qua một mạng pre-net nhỏ gồm 2 lớp kết nối hoàn chỉnh (fully connection layers) gồm 256 đơn vị ReLU ẩn mỗi lớp, xen kẽ các lớp dropout với tỉ lệ dropout bằng 0.5.
- Đầu ra của lớp prenet và Attention context vector (thu được từ cơ chế Attention) được nối với nhau và đưa qua 2 lớp LSTM đơn hướng (uni-direction) với 1024 đơn vị trong mỗi lớp.
- Đầu ra của 2 lớp LSTM kết hợp cùng Attention context vector được nối với nhau và được chuyển đến lớp kết nối hoàn chỉnh gồm 80 nơ ron tương ứng với số kênh quang phổ. Lớp decoder cuối cùng này dự đoán mel-spectrogram theo từng khung (frame). Đầu ra của nó đóng vai trò là đầu vào cho bước thời gian tiếp theo của bộ giải mã trong PreNet.
- Ngoài phép chiếu lên lớp kết nối hoàn chỉnh 80 nơ ron, việc ghép dữ liệu đầu ra của các lớp LSTM với Attention context vector được hướng vào một lớp được kết nối đầy đủ với một nơ ron với kích hoạt sigmoid - gọi là "Stop token". Nó dự đoán xác suất khung được tạo ở bước bộ giải mã là khung cuối cùng. Lớp này được thiết kế để tạo ra một biểu đồ quang phổ không cố định mà có độ dài tùy ý ở giai đoạn đầu ra mô hình. Ở giai đoạn đầu ra, phần tử này xác định số bước của bộ giải mã.
- Các mel-spectrogram được dự đoán được đưa qua mạng tích chập 5 lớp post-net dự đoán những đặc tính phần dư, kết hợp cùng prediction để tăng tính khôi phục tổng thể nâng cao chất lượng tái tạo âm thanh. Mỗi tầng trong post-net gồm có 512 bộ lọc với hình dạng là 5x1 đi kèm với chuẩn hóa theo lô (batch normalization), theo sau bởi 1 tầng tanh, trừ lớp cuối cùng.

Tại mỗi bước của bộ giải mã, mô hình cố gắng giải mã một khung phổ. Tuy nhiên, không phải toàn bộ thông tin từ bộ mã hóa sẽ được sử dụng ở mỗi bước. Ví dụ, nếu đầu vào là một chuỗi văn bản gồm 200 ký tự và biểu đồ quang phổ tương ứng là 800 khung hình, thì sẽ có 4 khung hình cho mỗi ký tự. Tuy nhiên, giọng nói được tạo ra trên cơ sở của một quang phổ như vậy sẽ hoàn toàn không tự nhiên. Trong câu nói sẽ có một số từ được phát âm nhanh hơn, một số từ khác chậm hơn, đôi khi có ngắt nghỉ giữa câu hoặc không. Đó là lý do tại sao cơ chế Attention là yếu tố quan trọng của toàn bộ hệ thống: nó thiết lập sự tương ứng giữa cao độ của bộ giải mã và thông tin từ bộ mã hóa để có được thông tin cần thiết để tạo ra một khung cụ thể. Và giá trị của trọng số chú ý càng lớn, thì càng phải “chú ý nhiều hơn” đến phần dữ liệu bộ mã hóa tương ứng khi tạo khung hình quang phổ.

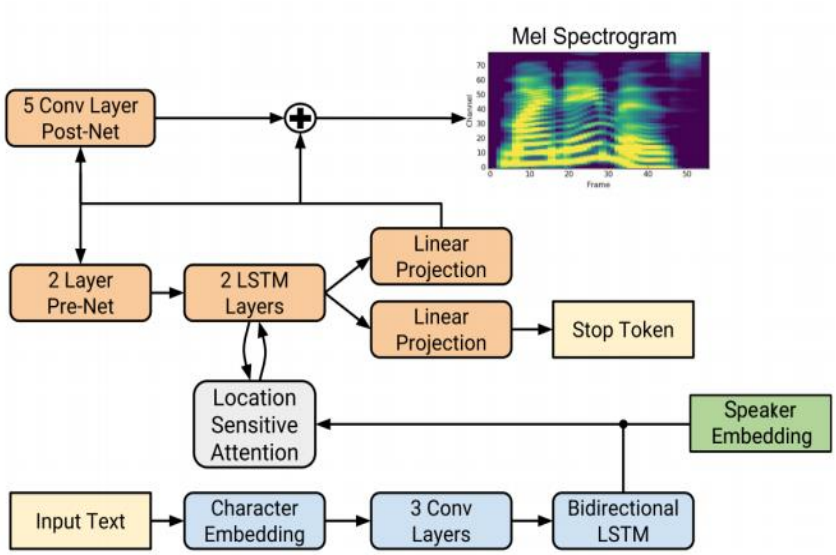
Rất khó để đưa ra bất kỳ đánh giá định lượng nào về hiệu suất của mô hình, tuy nhiên trong trường hợp của bộ Synthesizer, người ta nhận thấy rằng các căn chỉnh do cơ chế Attention tạo ra khá chính xác [28]. Hình 4.7 mô tả một ví dụ căn chỉnh giữa âm vị từ bộ mã hóa và khung mel-spectrogram của bộ giải mã, có thể nhận thấy rằng tuy có một số vị trí nhiễu, giữa bộ mã hóa và bộ giải mã của mo đun synthesizer đã có sự tương ứng nhất định.



Hình 4.7 Ví dụ alignment giữa encoder với decoder

4.2.2 Kiến trúc synthesizer nhúng đặc trưng người nói

Mô đun synthesizer kế thừa kiến trúc chung từ Tacotron 2, với véc tơ đặc trưng người nói được xem như đầu vào thứ hai của mạng.



Hình 4.8 Kiến trúc mô đun Synthesizer với nhúng đặc trưng người nói [28]

Tại đây, đặc tính âm thanh của giọng nói được trích xuất từ bộ mã hóa là speaker embedding được ghép với đầu ra của Synthesized Encoder, sử dụng làm đầu vào cho cơ chế Location Sensitive Attention.

Trong quá trình huấn luyện, đặc trưng âm thanh được trích xuất từ từng mẫu giọng nói có nhãn trong tập huấn luyện thay vì đặc trưng của một người nói nhằm mang lại độ tự nhiên tốt nhất. Tại thời điểm suy luận, với mô hình được chọn, hệ thống sẽ chọn ngẫu nhiên một bản mẫu âm thanh của người nói đích có sẵn trong tập huấn luyện để trích xuất đặc trưng hoặc lấy véc tơ đặc trưng của người nói bằng cách lấy trung tâm của một số hoặc tất cả véc tơ đặc trưng của người nói đó.

Tuy nhiên, thường có sự khác biệt lớn về âm điệu và cao độ trong các câu nói của cùng một người nói trong tập dữ liệu, đặc biệt khi số lượng câu được lấy mẫu hớn, khi họ nhại các ký tự khác nhau [32]. Từng câu nói có sự thay đổi nội bộ về âm điệu và cao độ thấp hơn, vì phạm vi chỉ giới hạn trong một câu. Do đó, véc tơ nhúng của một câu nói thể hiện chính xác đặc trưng âm học của người nói hơn là véc tơ nhúng của người nói.

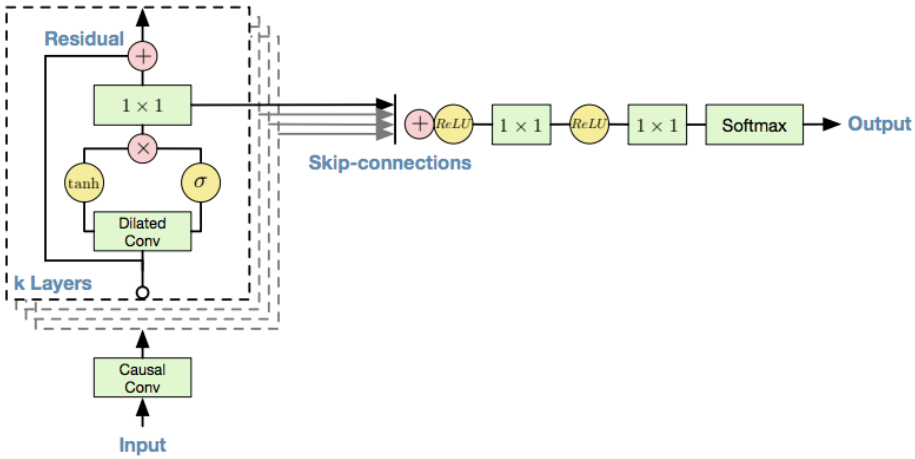
4.3 Vocoder

Vocoder (từ ghép giữa voice và encoder) là một hệ thống phân tích và tổng hợp tín hiệu tiếng nói của con người. Hệ thống này được sử dụng trong hai quá trình huấn luyện và tổng hợp tiếng nói: Ở quá trình huấn luyện, dữ liệu âm thanh được phân tích thành các đặc tính vốn có của giọng nói của con người (ví dụ như tần số cơ bản, phổ cường độ, ...) thay đổi theo thời gian, các đặc tính này được xem như đặc trưng huấn luyện trong mạng nơ ron học sâu. Ở quá trình tổng hợp, vocoder sử dụng các đặc trưng âm học có được từ bộ tổng hợp trước đó, đảo ngược quy trình huấn luyện chuyển đầu vào thành tín hiệu tiếng nói.

Xuất hiện lần đầu vào năm 1987, từ đó đến nay có nhiều loại Vocoder được phát triển để cải thiện chất lượng phân tích và tổng hợp tiếng nói như Straight, World [33], Magphase, …

Tacotron 2 sử dụng mạng WaveNet [25] chỉnh sửa như bộ Vocoder, đây là kiểu kiến trúc mạng học sâu cho phép sinh ra sóng âm dạng thô. Mô hình này hoàn toàn là dựa trên lý thuyết xác suất và tự hồi quy, dự đoán ra phân phối của mỗi mẫu âm thanh (audio sample) dựa trên mẫu trước đó, do đó có thể trực tiếp tạo ra âm thanh bằng cách dự đoán từng mẫu một theo kiểu tự động phản hồi.

Kiến trúc Wavenet có hai phần: một ngăn xếp tích chập và một mô-đun xử lý sau (post-processing). Ngăn xếp tích chập bao gồm các khối tích chập được giãn nở một chiều với hệ số giãn nở tăng theo cấp số nhân với độ sâu lớp, cho phép trường tiếp nhận rất lớn và độ phi tuyến mạnh cần thiết để tạo mô hình âm thanh thô và hoạt động như một bộ trích xuất tính năng đa tỷ lệ, trong khi mô-đun xử lý sau kết hợp thông tin từ các khối chập để dự đoán mẫu tiếp theo.



Hình 4.9 Kiến trúc một mạng Wavenet tiêu chuẩn [25]

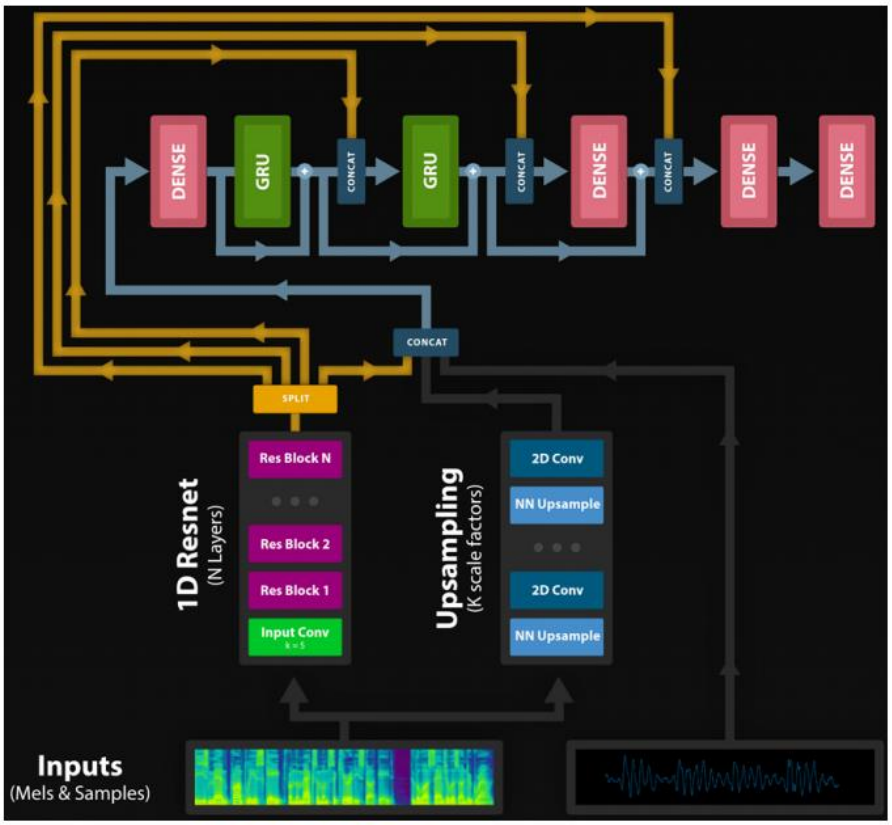
Với bài toán tổng hợp tiếng nói, WaveNet sinh ra tiếng nói có tính tự nhiên cao nhất so với những hệ thống nối âm hoặc hệ thống dựa trên tham số ở cả tiếng Anh và tiếng Trung Quốc phổ thông tốt nhất tại thời điểm nó xuất hiện (2016). Tuy nhiên, WaveNet cũng được biết đến là kiến trúc Vocoder học sâu chậm nhất tại thời điểm suy luận. WaveRNN (Wave Recurrent Neural Networks) được xây dựng dựa trên kiến trúc Wavenet. WaveRNN giúp tăng tốc độ suy luận của WaveNet trong khi vẫn giữ nguyên chất lượng [34].

Một lược đồ đơn giản để tính tốc độ suy luận của các mạng rơ ron tự động hồi quy được đề xuất bởi [34]:

$$T(u) = |\mathrm{u}| \sum_{i=1}^{N} (c(op_i) + d(op_i)). \qquad PT\ 4.6$$

Với vectơ mục tiêu u, có $|u|$ mẫu để dự đoán, N là số ma trận véc tơ để tính toán cho mỗi mẫu, $c(op_i)$ là thời gian tính toán ở lớp i và $d(op_i)$ là chi phí tính toán (thường cho phép I/O) cho lớp i. Với âm thanh huấn luyện được sử dụng trong đồ án này có sample rate là 16kHz tương đương chỉ 6s âm thanh $|u|$ chứa gần 100000 mẫu. Kiến trúc WaveNet tiêu chuẩn trong Tacotron 2 chiếm ba ngăn xếp gồm 10 khối residual mỗi lớp của hai lớp, dẫn đến N = 60. WaveRNN giúp tăng tốc độ suy luận so với WaveNet không chỉ bằng cách giảm N, mà còn giảm cả $c(op_i)$ và $d(op_i)$. Trong WaveRNN, toàn bộ 60 phép tích chập từ WaveNet được thay thế bằng một lớp GRU duy nhất. Các tác giả cho rằng chỉ riêng độ phi tuyến tính cao của một lớp GRU đã đủ gần để bao hàm sự phức tạp của toàn bộ mô hình WaveNet [35].

Quá trình chuyển tiếp của WaveRNN được thực hiện chỉ với N = 5 vectơ ma trận trong một lược đồ.



Hình 4.10 Kiến trúc mô hình WaveRNN [36]

Quá trình WaveRNN biến đổi mel-spectrogram đầu vào thành âm thanh tổng hợp được mô tả ở Hình 4.10:

- Đầu vào của mạng là phân đoạn sóng âm *t-1* và GTA (ground truth audio) mel-spectrogram tương ứng phân đoạn sóng âm *t*.
- Mel-spectrogram *t* đi qua mạng upsampling (để đạt được độ dài trùng với độ dài âm thanh đích) và đi qua mạng resnet (Resividual Net) sinh ra các đặc trưng. Kết quả này được lặp lại để khớp với thời lượng của đoạn sóng âm.
- Đầu ra mạng Resnet là conditioning vector được chia thành bốn phần bằng nhau theo chiều kênh. Phần đầu tiên được nối với biểu đồ quang phổ đã được lấy mẫu và với phân đoạn sóng âm của bước thời gian trước đó *t-1*. Véc tơ kết quả qua các bước biến đổi với các kết nối bỏ qua (skip connection): hai lớp GRU và một lớp dày đặc (dense layer). Giữa mỗi bước, các phần còn lại conditioning vector lần lượt được nối với sóng âm trung gian.
- Cuối cùng, hai lớp dense tạo ra sự phân phối trên các giá trị rời rạc tương ứng với mã hóa của âm thanh thu được đầu ra là phân đoạn sóng âm *t*.

CHƯƠNG 5. THỬ NGHIỆM VÀ ĐÁNH GIÁ

5.1 Xây dựng cơ sở dữ liệu

Thông tin về các bộ dữ liệu âm thanh được miêu tả ở Bảng 5.1. Tất cả các bộ dữ liệu này đều được thu thập và sử dụng tại Trung tâm Không gian mạng Viettel.

Bảng 5.1 Thông tin về các bộ dữ liệu

Tên bộ dữ liệu	Số người nói	Tổng thời lượng	Số câu nói
vtr500	1400	398 giờ	275637
vtr_clean	143	52 giờ	35754
tts	11	153 giờ	107059
tts_full	154	189 giờ	137317
target_M_full	1	24.29 giờ	15170
target_M_small	1	20 phút	180
target_F_full	1	8.84 giờ	7710
target_F_small	1	20 phút	263

Trong đó:

- Bộ dữ liệu vtr500 là bộ dữ liệu được chọn ra từ bộ dữ liệu hơn 500 giờ cho bài toán nhận diện tiếng nói. Bộ dữ liệu này có chất lượng thu âm thấp do người nói tự ghi âm bằng điện thoại, chất lượng giọng nói không đồng đều, chất lượng thấp (có chứa tiếng thở, nhiễu nền, giọng hụt hơi, lưỡi ngắn, nói ngọng, nền ồn, dội âm, thì thầm, âm lượng ko đều, rè, chép miệng).
- Bộ dữ liệu tts là bộ dữ liệu có chất lượng cao, được ghi âm từ phòng thu chuyên nghiệp và phát thanh viên chuyên nghiệp. Tuy nhiên vì vấn đề số lượng người nói ít, đồ án kết hợp bộ dữ liệu vtr_clean (chứa khoảng 50 giờ dữ liệu có chất lượng trung bình khá được trích chọn từ bộ dữ liệu vtr500) để kết hợp tạo thành bộ dữ liệu đa người nói tts_full có chất lượng tốt hơn nhằm tăng chất lượng cho mô hình thích ứng.
- Bộ dữ liệu target_M_full là bộ dữ liệu giọng nam M chất lượng khá, được phát thanh viên nam miền Bắc ghi âm từ phòng ghi hình không có vọng âm.
- Bộ dữ liệu target_F_full có chất lượng trung bình, giọng nữ F miền Nam, được ghi âm bằng điện thoại, dữ liệu có tiếng vọng âm khá to và rõ.
- Bộ dữ liệu target_F_small và target_M_small là dữ liệu được chọn ngẫu nhiên lần lượt từ bộ target_F_full, target_M_full.

Dữ liệu là một trong những phần quan trọng nhất ảnh hưởng đến chất lượng tổng hợp tiếng nói, do đó để có được một hệ thống tổng hợp tiếng nói chất lượng cần phải chuẩn bị bộ dữ liệu chất lượng. Để xây dựng được bộ cơ sở dữ liệu có chất

lượng tốt, cần phải xử lý dữ liệu thu thập được (tiền xử lý dữ liệu), gồm 2 bước: tiền xử lý dữ liệu âm thanh và chuẩn hóa văn bản.

Dữ liệu thu thập được có rất nhiều vấn đề như nhiễu nền, giọng đọc lúc to lúc nhỏ, nhiều từ vay mượn hay tệp âm thanh quá dài. Do đó cần tiền xử lý dữ liệu huấn luyện:

- Giọng đọc lúc to lúc nhỏ dẫn đến kết quả tổng hợp tiếng nói cũng bị như vậy, thậm trí còn trầm trọng hơn khi lúc thì quá to và lúc thì quá nhỏ. Để giải quyết vấn đề này cần cân bằng cường độ âm thanh của dữ liệu huấn luyện.
- Có các từ viết tắt và từ tiếng nước ngoài trong văn bản, nếu sử dụng các âm vị để phiên âm từ này thì sẽ gây méo tín hiệu tiếng nói của âm vị đó, do đó các câu có chứa những từ này cần được loại bỏ.
- Đưa các dữ liệu âm thanh về cùng tần số lấy mẫu (sampling rate) 16kHz.

Quá trình tiền xử lý văn bản đầu vào giúp cho văn bản đầu vào có thể đọc được một cách rõ ràng, nhất quán. Quá trình tiền xử lý sẽ chuẩn hóa các thành phần không chuẩn như từ viết tắt, số, ngày tháng, tách các dấu câu thành các ký tự độc lập riêng biệt.

- Văn bản đầu vào sẽ được chia tách thành các các phần nhỏ hơn dựa theo khoảng trắng. Từng thành phần này được tìm kiếm trong từ điểm âm tiết. Nếu có trong từ điển thì đó là thành phần có thể đọc được. Nếu không có sẽ tục được tìm kiếm trong từ điển từ mượn, từ viết tắt
- Những thành phần không có trong từ điển âm tiết nếu được tìm thấy ở trong từ điển viết tắt sẽ được chuyển thành một chuỗi các từ chuẩn theo từ điển âm tiết. Nếu không tìm thấy sẽ được chuyển áp dụng biểu thức chính quy.
- Các thành phần mà không xuất hiện trong cả hai từ điển nêu ở trên như: ngày tháng, chữ viết tắt, tỉ số, … sẽ được tìm kiếm bằng biểu thức chính quy các mẫu có sẵn phù hợp rồi thay thế. Ví dụ thành phần ngày tháng có dạng "…/..." sẽ được thay thế bằng "ngày … tháng …".

5.2 Huấn luyện mô hình

Trong đồ án, các mô hình được thử nghiệm bao gồm:

- Mô hình học chuyển tiếp giọng nói mới ít dữ liệu chất lượng khá/trung bình từ mô hình đa người nói chất lượng cao/thấp
- Đào tạo giọng nói mới từ mô hình Tacotron 2 đơn người nói với nhiều/ít dữ liệu
- Đào tạo giọng nói mới từ mô hình DNN đơn người nói ít dữ liệu

Trong đó, kiến trúc mô hình Tacotron2 và mô hình học chuyển tiếp đều được mô tả cụ thể ở Chương 4. Với hai hướng thích ứng giọng nói mới, kiến trúc mô hình chuyển tiếp đều được giữ nguyên.

Mô hình DNN còn lại được phát triển bởi Centre for Speech Technology Research (CSTR), Đại học Edinburgh. Đây là một mô hình DNN để tổng hợp lời nói tham số thống kê, sử dụng kết hợp với bộ xử lý văn bản Festival và bộ mã hóa giọng nói WORLD vocoder. Mô hình này có khả năng tổng hợp giọng nói từ nguồn dữ liệu khan hiếm.

Các mô hình được huấn luyện trên máy tính có cấu hình CPU E5-2640 32 nhân, tần số 2.6 GHz. RAM 128 Gb. GPU Quadro K22000 với 4 Gb GPU Memory.

Các mô hình học chuyển tiếp và Tacotron-2 nhận âm thanh đầu vào với tốc độ lấy mẫu (sampling rate) 16kHz, mel-spectrogram với 80 kênh, hop_size = 200 (tương đương 12.5ms), window_size = 800 (tương đương 50ms). Số khung được sinh ra tại mỗi bước decoder = 2 nhằm giúp tăng tốc độ hội tụ. Có thể thấy rằng số bước bộ giải mã (250) tại Hình 4.7 khớp với số khung dự đoán (500) tại Hình 4.5 bởi số khung được giải mã tại mỗi bước (2).

Thông tin về quá trình huấn luyện được miêu tả tại Bảng 5.2. Do có giới hạn ràng buộc về môi trường cũng như thời gian, các mô hình seq2seq được triển khai với batch = 16 thay vì 64 như đề xuất của tác giả và quá trình huấn luyện dừng lại khi chưa thực sự hội tụ tốt.

Bảng 5.2 Thông tin về các mô hình giọng nói được huấn luyện

Mô hình huấn luyện	Bộ dữ liệu	Thời gian huấn luyện	Mã mô hình
Tacotron2 nhúng đặc trưng người nói	vtr500	11 ngày	vtr500
	tts + vtr_clean	10 ngày	tts
	vtr500 -> target_M_small	1.5 ngày	Iy
	vtr500 -> target_F_small	1.5 ngày	Ix
	tts -> target_M_small	1.5 ngày	IIy
	tts -> target_F_small	1.5 ngày	IIx
Tacotron2	target_M_full	6 ngày	IIIy
	target_F_full	5 ngày	IIIx
	target_M_small	3 ngày	IVx
	target_F_small	3 ngày	IVy
DNN	target_M_small	20 phút	Vy
	target_F_small	20 phút	Vx

5.3 Đánh giá kết quả

Mục tiêu của đánh giá này là để kiểm tra xem việc áp dụng phương pháp thêm véc tơ nhúng đặc trưng âm học của người nói vào tổng hợp tiếng nói kết hợp cùng phương pháp học chuyển tiếp có thực sự giúp tổng hợp được âm thanh tiếng nói chất lượng cao hay không.

5.3.1 Phương pháp đánh giá

Đồ án này quan tâm đến việc đánh giá độ tự nhiên, dễ hiểu của giọng nói và độ giống giọng của âm thanh được tạo.

a) Độ tự nhiên và độ dễ hiểu

Hai tính chất quan trọng của chất lượng hệ thống tổng hợp giọng nói là mức độ tự nhiên và mức độ dễ nghe. Mức độ tự nhiên của giọng nói tổng hợp chỉ đến giọng nói tổng hợp được phát âm giống giọng người thật nói một cách tự nhiên. Mức độ dễ nghe chỉ đến việc câu phát âm có thể hiểu được dễ dàng không.

Tiêu chí để đánh giá độ tự nhiên và độ dễ hiểu được dựa trên điểm ý kiến trung bình (Mean opinion score - MOS)

Phương pháp đánh giá: Mời 21 người tham gia đánh giá và cho điểm chất lượng hệ thống trên tập dữ liệu đánh giá. Thông tin về người nghe được thể hiện ở Bảng 5.3.

Bảng 5.3 Thông tin người nghe đánh giá hệ thống tổng hợp tiếng nói

Thông tin người nghe						
Giới tính		Phương ngữ			Chuyên gia	
Nam	Nữ	Bắc	Trung	Nam	Đúng	Không
13	8	15	4	2	5	16

Tập dữ liệu đánh giá cho mỗi mô hình giọng nói là tập gồm mười tệp âm thanh được tổng hợp từ mười văn bản có độ dài khác nhau, được so sánh cùng giọng nói gốc có cùng nội dung văn bản, không xuất hiện trong tập dữ liệu huấn luyện.

Mỗi người đánh giá sẽ được nghe 60 tệp được chọn ra từ các mô hình đối với mỗi giọng và chấm điểm cho từng tệp.

Kết quả đánh giá được cho trên thang điểm 5 với các mức:

- 1 – Rất tồi (nghe khó chịu, khó hiểu).
- 2 – Tồi (nghe khó chịu, nhiều yếu tố nhân tạo, chỉ nghe hiểu được một số ít từ).
- 3 – Khá (chưa được tự nhiên, khá nhiều yếu tố nhân tạo, nghe vẫn có một chút khó chịu/không nghe rõ vài từ).
- 4 – Tốt (tương đối tự nhiên, giống giọng người thật, nghe rõ ràng từng từ).
- 5 – Rất tốt (rất tự nhiên, rất giống giọng người thật).

Kết quả cuối cùng là điểm trung bình của tất của người nghe.

b) Độ tương đồng

Độ tương đồng được đo bằng điểm nhận dạng người nói được tính toán dựa trên khoảng cách cosine giữa véc tơ đặc trưng người nói được trích xuất từ tệp âm thanh gốc và tệp âm thanh được tổng hợp từ các mô hình với công thức ở PT **5.1**.

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}} \quad PT\ 5.1$$

Mỗi tệp âm thanh tổng hợp được trong tập dữ liệu đánh giá sẽ được tính tương đồng với tệp âm thanh gốc do người thật nói có cùng nội dung văn bản, sau đó điểm tương đồng của mỗi mô hình sẽ được lấy bằng trung bình bằng điểm đánh giá tương đồng của các câu tổng hợp được từ mô hình.

5.3.2 Kết quả đánh giá

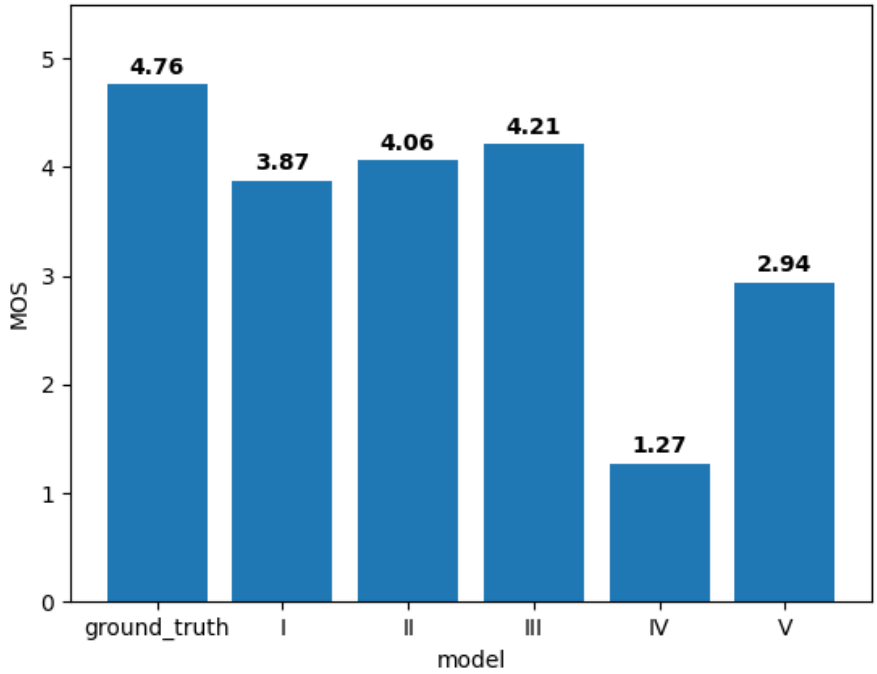
Các hệ thống được đánh giá bao gồm mười hệ thống tổng hợp tiếng nói được xây dựng theo ba mô hình DNN, Tacotron 2 và Tacotron2 nhúng đặc trưng người nói cho mỗi giọng nam M và giọng nữ F. Các hệ thống này được huấn luyện từ cùng một bộ dữ liệu, và cùng một tập đánh giá nêu trên.

Từ Bảng 5.4 nhận thấy, điểm đánh giá về độ tự nhiên và độ dễ hiểu của các mô hình tổng hợp tiếng nói giọng nam có xu hướng cao hơn so với các mô hình tổng hợp giọng nữ có cùng kiến trúc, điều này thể hiện tầm quan trọng của độ sạch dữ liệu. Để giọng nói tổng hợp đạt được chất lượng cao, dữ liệu huấn luyện cần có chất lượng tốt.

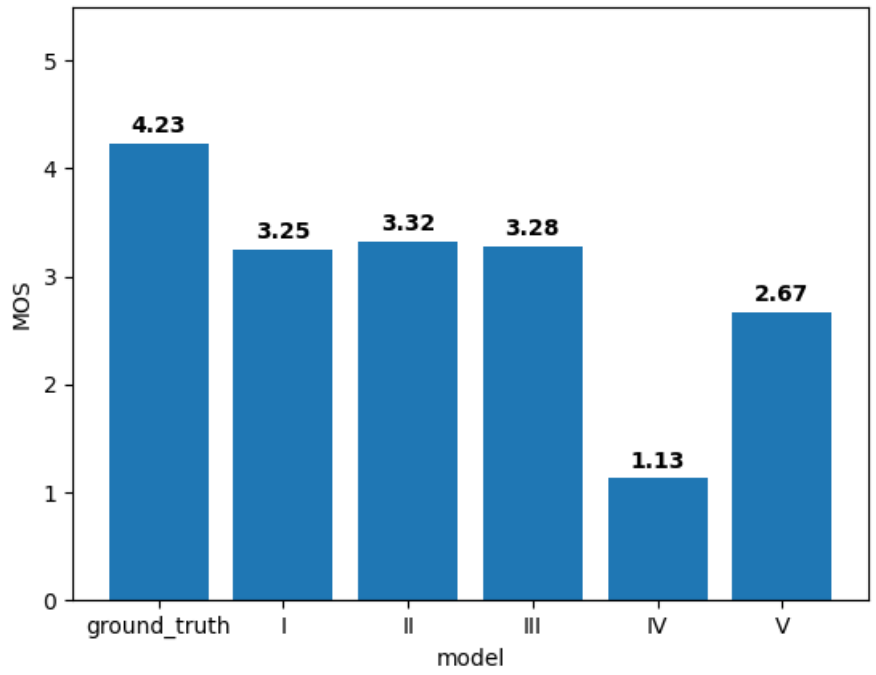
Bảng 5.4 Đánh giá điểm MOS các mô hình cùng kiến trúc giữa giọng nam và giọng nữ

Mô hình	Giọng nam	Giọng nữ
Người thật nói	4.76	4.23
I	3.87	3.25
II	4.06	3.32
III	4.21	3.28
IV	1.27	1.13
V	2.94	2.67

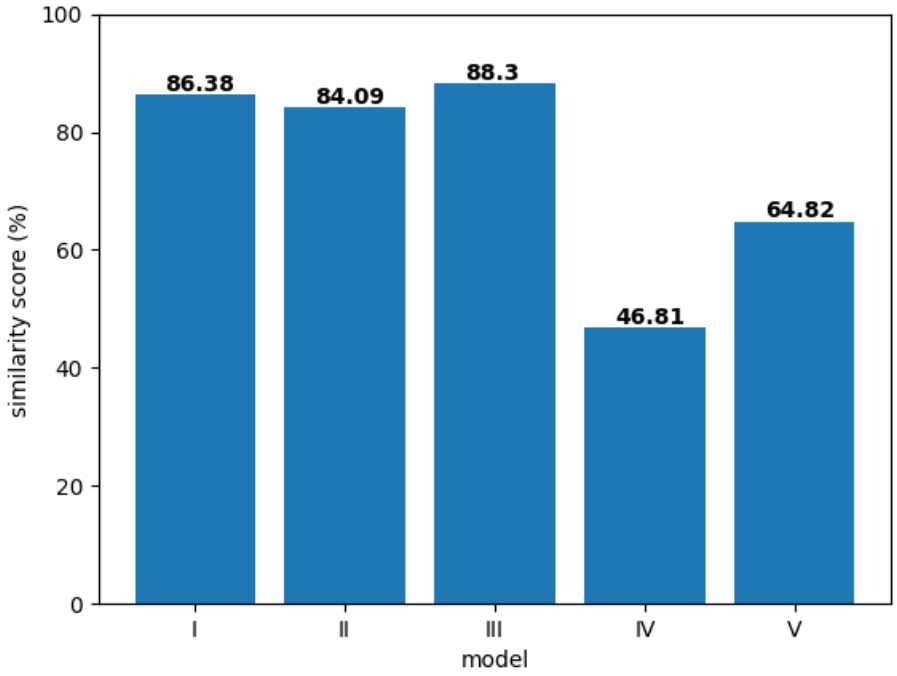
Kết quả đánh giá thể hiện rõ trong các Hình 5.1, Hình 5.2, Hình 5.3, Hình 5.4, điều này cho thấy rõ ràng là việc áp dụng nhúng đặc trưng người nói đã góp phần cải thiện chất lượng hệ thống tổng hợp tiếng nói cả về độ hiểu, độ tự nhiên và vẫn giữ được đặc trưng âm thanh của người nói chỉ với một lượng ít dữ liệu.



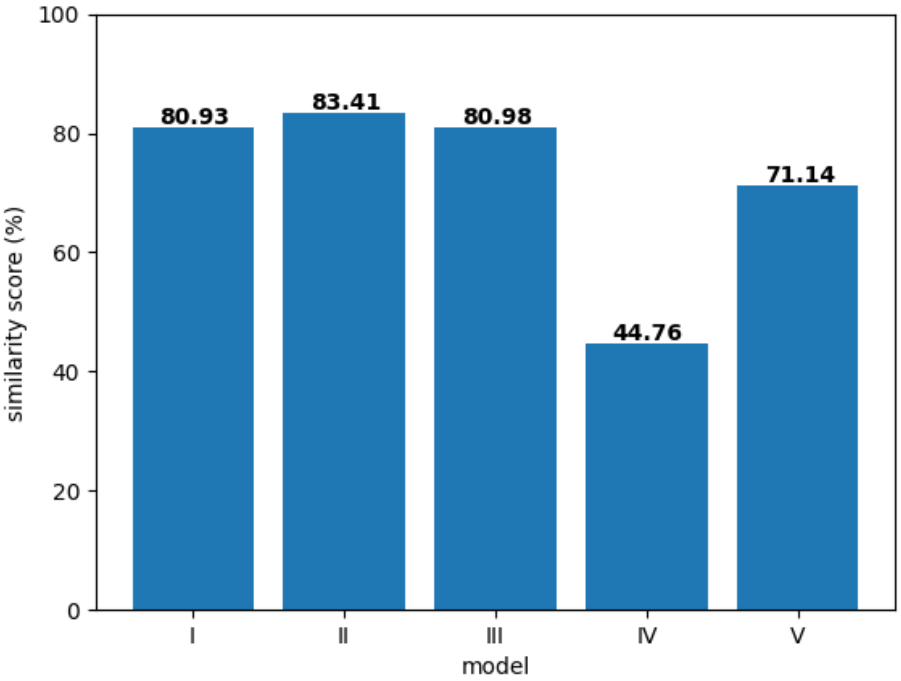
Hình 5.1 Đánh giá độ tự nhiên và độ dễ hiểu cho các mô hình giọng nam



Hình 5.2 Đánh giá độ tự nhiên và độ dễ hiểu cho các mô hình giọng nữ



Hình 5.3 Đánh giá độ tương đồng các mô hình giọng nam với giọng nói thật



Hình 5.4 Đánh giá độ tương đồng các mô hình giọng nữ với giọng nói thật

Đối với mô hình giọng nam chất lượng dữ liệu khá, ta có thể thấy trong trường hợp dữ liệu hạn chế (chỉ 20 phút), mô hình tổng hợp tiếng nói Tacotron2 kết quả tệ, hệ thống thậm chí không thể tổng hợp được một câu nói rõ ràng tự nhiên, mô hình DNN có thể phát âm rõ ràng nhưng chưa được tự nhiên. Sau khi áp dụng phương pháp học chuyển tiếp nhúng đặc trưng người nói, kết quả được cải thiện rõ rệt, đạt mức độ tự nhiên, dễ hiểu và độ tương đồng xấp xỉ so với mô hình được huấn luyện bằng lượng dữ liệu mục tiêu lớn (25 giờ).

Đối với mô hình giọng nữ chất lượng trung bình, mô hình thích ứng nhúng đặc trưng người nói sử dụng dữ liệu hạn chế (20 phút) thậm chí cho giúp tăng độ tương đồng và chất lượng âm thanh tổng hợp so với mô hình Tacotron 2 sử dụng nhiều dữ liệu (9 giờ). Điều này cho thấy việc sử dụng mô hình Tacotron 2 nhúng đặc trưng người nói có thể cải thiện chất lượng giọng nói tổng hợp từ giọng mẫu chất lượng trung bình bằng cách học chuyển tiếp từ mô hình được huấn luyện trên tập dữ liệu đa người nói chất lượng cao.

Tổng quát có thể thấy, mô hình Tacotron2 nhúng đặc trưng người nói giúp giảm thời gian huấn luyện mô hình và lượng dữ liệu cần thiết để huấn luyện nhưng vẫn giúp giữ được độ tự nhiên gần tương đương lượng dữ liệu lớn.

CHƯƠNG 6. KẾT LUẬN

6.1 Tổng kết

Đồ án đã đạt được mục tiêu đề ra khi đã xây dựng hệ thống tổng hợp giọng nói tách mô hình người nói khỏi mô hình tổng hợp, giúp giảm dữ liệu huấn luyện trong khi vẫn giữ được chất lượng giọng nói tổng hợp. Mặc dù vậy đồ án vẫn còn nhiều hạn chế như: giọng nói tổng hợp chưa đạt được mức độ tự nhiên khi so sánh với giọng nói thật, kết quả đánh giá được thực hiện bởi hầu hết tình nguyện viên không phải là chuyên gia có thể có nhiều sai sót. Bên cạnh đó, phương pháp đề xuất chưa được triển khai thực tế, dẫn dến các đánh giá về hiệu năng cũng như hiệu quả kinh tế chưa được thực hiện.

Trong quá trình thực hiện đồ án, tác giả đã thu được nhiều kiến thức và kinh nghiệm như:

- Tìm hiểu và làm chủ được công nghệ thích ứng giọng nói, xây dựng thành công hệ thống thích ứng giọng nói tiếng Việt đầu tiên sử dụng công nghệ học sâu.
- Hướng tới bài toán tổng hợp tiếng nói nhiều người nói. Từ đó phát triển hệ thống có khả năng tổng hợp nhanh giọng nói mới cũng như thêm nhiều giọng nói mới.

6.2 Hướng phát triển trong tương lai

Mặc dù phương pháp đề xuất đã đạt được mục tiêu đề ra là giúp giảm lượng dữ liệu huấn luyện trong quá trình tổng hợp giọng nói, tuy nhiên giọng nói tổng hợp vẫn chưa đạt chất lượng cao khi so sánh với giọng nói thật của con người. Vì vậy hướng phát triển tiếp theo của đồ án là tiếp tục cải thiện chất lượng của giọng nói tổng hợp bằng các phương pháp như:

- Thêm các giải pháp mới cho vấn đề chuẩn hóa văn bản đầu vào, tự động thêm dấu câu ngắt nghỉ giúp tăng độ tự nhiên trong tốc độ đọc của giọng tổng hợp.
- Tăng chất lượng âm thanh huấn luyện đầu vào đối với dữ liệu âm thanh chất lượng trung bình/thấp, giúp giọng nói tổng hợp tự nhiên hơn.
- Áp dụng các kiến thức thu thập được trong quá trình thực hiện đồ án vào các sản phẩm thực tế tại đơn vị đang làm việc để tăng hiệu suất sản phẩm cũng như nhận được đánh giá phản hồi thực tế từ phía người sử dụng, cung cấp góc nhìn khách quan hơn về hiệu quả của phương pháp.
- Thử nghiệm phương pháp với bài toán tổng hợp tiếng nói có cảm xúc.

TÀI LIỆU THAM KHẢO

[1] D. Suendermann, H. Höge, and A. Black, "Challenges in Speech Synthesis," *Speech Technology*, pp. 19-32, 2010.

[2] "Perceptron: The Artificial Neuron," [Online]. Available: https://mc.ai/perceptron-the-artificial-neuron. [Accessed May 2019].

[3] "Summary of commonly used activation functions in the field of machine learning," [Online]. Available: https://www.programmersought.com. [Accessed 11 2020].

[4] "Neural Networks and Deep Learning - Part 1: The basic of Neural Networks," [Online]. Available: https://viblo.asia/p/neural-networks-and-deep-learning-part-1-the-basic-of-neural-networks-OREGwLwlelN. [Accessed 12 2020].

[5] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, "Gradientbased learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, p. 2278–2324, 1998.

[6] Standford University, "Cs231n: Convolutional neural networks for visual recognition.," [Online]. Available: http://cs231n.stanford.edu/. [Accessed 12 2020].

[7] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *J. Mach. Learn. Res*, vol. 15, p. 1929–1958, 2014.

[8] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cogn. Model*, vol. 5, no. 3, p. 1, 1988.

[9] "Understanding LSTM Network," [Online]. Available: https://colah.github.io/posts/2015-08-Understanding-LSTMs. [Accessed 21 May 2019].

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735-1780, 1997.

[11] P. T. Sơn and P. T.Nghĩa, "Một số vấn đề về tổng hợp tiếng nói tiếng Việt," p. 4, 2014.

[12] T. T. T. Nguyen, "HMM-based Vietnamese Text-To-Speech: Prosodic Phrasing Modeling, Corpus Design System Design, and Evaluation," PhD Thesis, Paris, 2015.

[13] Q. Nguyễn Hồng, Phân tích văn bản cho tổng hợp tiếng nói tiếng Việt, Đại Học Bách khoa Hà Nội, 2006.

[14] P. Taylor, Text-to-speech synthesis, Cambridge university press, 2009.

[15] Lê Hồng Minh, "Một số kết quả nghiên cứu và phát triển hệ phần mềm chuyển văn bản thành tiếng nói cho tiếng Việt bằng tổng hợp formant," *Kỷ yếu Hội thảo Khoa học Quốc gia lần thứ nhất*, pp. 292-301, 2003.

[16] Nguyễn Hữu Minh, "Xác định khoảng ngừng giữa các âm tiết, cường độ và trường độ của âm tiết cho bộ phát âm tiếng Việt," *PhD Thesis*, 2009.

[17] S. J. Kim, "HMM-based Korean speech synthesizer with two-band mixed excitation model for embedded applications," in *PhD Thesis, Ph. D. dissertation, School of Engineering, Information and Communication University*, Korea, 2007.

[18] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech Synthesis Based on Hidden Markov Models," vol. 101, pp. 1234-1252, 2013.

[19] T. Masuko, "HMM-Based Speech Synthesis and Its Applications," p. 185, 202.

[20] T. Fukada, K. Tokuda, T. Kobayashi, and S. Imai, "An adaptive algorithm for mel-cepstral analysis of speech," vol. 1, pp. 137-140, 1992.

[21] Heiga Zen, Tomoki Toda, "An overview of nitech HMM-based speech synthesis system for Blizzard Challenge," in *in Ninth European Conference on Speech Communication and Technology (Eurospeech)*, 2005.

[22] Heiga Zen, Andrew Senior, Mike Schuster, "Statistical parametric speech synthesis using deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[23] Ilya Sutskever, Oriol Vinyals, Quoc V Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014.

[24] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, Rif A Saurous, Yannis Agiomvrgiannakis, Yonghui Wu, "Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[25] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," 2016.

[26] D. D. Tran, "Synthèse de la parole à partir du texte en langue vietnamienne," PhD Thesis, Grenoble INPG, 2007.

[27] Hieu-Thi Luong, Shinji Takaki, Gustav Eje Henter, Junichi Yamagishi, "Adapting and Controlling DNN-Based Speech Synthesis Using Input

Codes," *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4902-4909, 2017.

[28] Ye Jia, Yu Zhang, Ron J. Weiss, Quan Wang, Jonathan Shen, Fei Ren, Zhifeng Chen, Patrick Nguyen, Ruoming Pang, Ignacio Lopez Moreno, Yonghui Wu, "Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis," *Neural Information Processing Systems 31*, no. 1806.04558, pp. 4485-4495, 2018.

[29] Ehsan Variani; Xin Lei; Erik McDermott; Ignacio Lopez Moreno; Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," *International Conference on Acoustic, Speech and Signal Processing (ICASSP),* 2014.

[30] Li Wan, Quan Wang, Alan Papir, Ignacio Lopez Moreno, "Generalized End-to-End Loss for Speaker Verification," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[31] Jonathan Shen, R. Pang, Ron J. Weiss, M. Schuster, Navdeep Jaitly, Z. Yang, Z. Chen, Yu Zhang, Yuxuan Wang, R. Skerry-Ryan, R. A. Saurous, Yannis Agiomyrgiannakis, Y. Wu, "Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4779-4783, 2018.

[32] Jemine, Corentin, "Real-Time Voice Cloning," Master Thesis, Belgique, 2019.

[33] Masanori Morise, Fumiya Yokomori, Kenji Ozawa, "WORLD: A Vocoder-Based High-Quality Speech Synthesis System for Real-Time Applications," *IEICE Trans. Inf. Syst,* vol. E99.D, no. 7, pp. 1877-1884, 2016.

[34] Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, Norman Casagrande, Edward Lockhart, Florian Stimberg, Aaron van den Oord, Sander Dieleman, Koray Kavukcuoglu, "Efficient Neural Audio Synthesis," 2018.

[35] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," *ArXiv*, vol. 1409, 2014.

[36] "Pytorch implementation of Deepmind's WaveRNN model," [Online]. Available: https://github.com/fatchord/WaveRNN. [Accessed 22 10 2020].