ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KĨ THUẬT THÔNG TIN

BÁO CÁO ĐỒ ÁN
MÔN XỬ LÝ THÔNG TIN GIỌNG NÓI – IE313
Đề tài: SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition

GVHD: ThS. Nguyễn Thành Luân

Nhóm sinh viên thực hiện:

1. Huỳnh Anh Dũng	MSSV: 22520278
2. Nguyễn Hoàng Hiệp	MSSV: 22520452
3. Hà Huy Hoàng	MSSV: 22520460
4. Nguyễn Duy Hoàng	MSSV: 22520467
5. Nguyễn Hồ Nam	MSSV: 22520915

Tp. Hồ Chí Minh, 06/2025

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

……, ngày…….....tháng……năm 2025

Người nhận xét

(Ký tên và ghi rõ họ tên)

BẢNG PHÂN CÔNG, ĐÁNH GIÁ THÀNH VIÊN:

Bảng 1: Bảng phân công, đánh giá thành viên

Họ và tên	MSSV	Phân công	Đánh giá
Huỳnh Anh Dũng	22520278	**Tuần 1:** Tìm data và tạo file manifest cho tiếng Anh **Tuần 2:** Tìm mô hình thay thế mô hình LSA và viết báo cáo phần 3.4.1 **Tuần 3:** Chạy thực nghiệm cho bộ LibreSpeech-100 có sử dụng Augment **Tuần 4:** Chạy DeepFusion trên VIVOS Corpus	**Tuần 1:** 100% **Tuần 2:** 100% **Tuần 3:** 100% **Tuần 4:** 100%
Nguyễn Hoàng Hiệp	22520452	**Tuần 1:** Tìm data và tạo file manifest cho tiếng Việt **Tuần 2:** Viết báo cáo phần 3.3.3 và 3.3.4 **Tuần 3:** Chạy thực nghiệm cho bộ LibreSpeech-100 không sử dụng Augment **Tuần 4:** Chạy DeepFusion trên LibreSpeech-100	**Tuần 1:** 100% **Tuần 2:** 100% **Tuần 3:** 100% **Tuần 4:** 100%
Hà Huy Hoàng	22520460	**Tuần 1:** Viết phần 3.1 và 3.2 **Tuần 2:** Chạy thực nghiệm cho bộ VIVOS sử dụng Augment **Tuần 3:** Viết báo cáo phần thực nghiệm **Tuần 4:** Hoàn thành các hạng mục cần nộp và viết phần làm thêm 3.4.4.3	**Tuần 1:** 100% **Tuần 2:** 100% **Tuần 3:** 100% **Tuần 4:** 100%
Nguyễn Duy Hoàng	22520467	**Tuần 1:** Viết báo cáo chương 1 **Tuần 2:** Thực nghiệm chạy các data trên các mô hình khác nhau **Tuần 3:** Làm slide và viết báo cáo đề mục 4. **Tuần 4:** Train và tìm cách kết nối LM theo shallow fusion	**Tuần 1:** 100% **Tuần 2:** 100% **Tuần 3:** 100% **Tuần 4:** 100%
Nguyễn Hồ Nam	22520915	**Tuần 1:** Viết báo cáo chương 2 **Tuần 2:** Viết báo cáo phần 3.3.2 **Tuần 3:** Chạy thực nghiệm cho bộ VIVOS không sử dụng Augment **Tuần 4:** Train và tìm cách kết nối LM theo kiến trúc của NeMo theo shallow fusion	**Tuần 1:** 100% **Tuần 2:** 100% **Tuần 3:** 100% **Tuần 4:** 100%

LỜI MỞ ĐẦU

Ngày nay, công nghệ nhận dạng giọng nói tự động (Automatic Speech Recognition – ASR) đang đóng vai trò quan trọng trong nhiều ứng dụng thực tiễn như trợ lý ảo, điều khiển bằng giọng nói, chuyển đổi giọng nói thành văn bản và nhiều lĩnh vực khác. Tuy nhiên, một trong những thách thức lớn trong quá trình huấn luyện mô hình ASR là sự đa dạng và phức tạp của tín hiệu âm thanh trong môi trường thực tế. Vì vậy, các phương pháp tăng cường dữ liệu (data augmentation) ngày càng được nghiên cứu và ứng dụng nhằm cải thiện hiệu suất của hệ thống.

Trong đồ án môn *Xử lý thông tin giọng nói – IE313*, nhóm chúng em lựa chọn nghiên cứu đề tài "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition". Đây là một phương pháp đơn giản nhưng hiệu quả, được đề xuất để cải thiện khả năng tổng quát hóa của mô hình nhận dạng giọng nói bằng cách thao tác trực tiếp trên biểu diễn phổ Mel (Mel Spectrogram). Phương pháp này không đòi hỏi mô hình kiến trúc phức tạp hay thay đổi quy trình huấn luyện, nhưng vẫn đem lại hiệu quả rõ rệt.

Thông qua đề tài này, nhóm mong muốn tìm hiểu rõ hơn về các kỹ thuật tăng cường dữ liệu trong ASR, đặc biệt là SpecAugment, từ đó áp dụng và đánh giá hiệu quả của phương pháp trong các mô hình học sâu. Đồ án được thực hiện dưới sự hướng dẫn của ThS. Nguyễn Thành Luân, và là kết quả của sự nỗ lực hợp tác giữa các thành viên trong nhóm.

Nhóm xin chân thành cảm ơn Thầy đã tận tình hướng dẫn, tạo điều kiện thuận lợi để chúng em hoàn thành đồ án này.

DANH MỤC CÁC BẢNG, HÌNH ẢNH

Danh mục các bảng:

Danh mục hình ảnh:

MỤC LỤC

Chương 1. GIỚI THIỆU CHUNG

1.1 Giới thiệu về môn học

Môn học *Xử lý Thông tin Giọng nói* được giảng dạy tại Đại học Công nghệ Thông tin, trực thuộc Khoa Khoa học và Công nghệ Thông tin – UIT, VNU-HCM. Đây là môn học quan trọng nhằm cung cấp cho sinh viên nền tảng lý thuyết và thực hành về quá trình xử lý, phân tích và ứng dụng các thông tin có trong giọng nói của con người. Qua môn học, sinh viên sẽ được làm quen với các khái niệm cơ bản về sản xuất giọng nói, cảm nhận giọng nói, xử lý tín hiệu, tổng hợp giọng nói, nhận dạng giọng nói và thậm chí là dịch giọng nói.

Xử lý Thông tin Giọng nói không chỉ là một lĩnh vực nghiên cứu tiên phong mà còn đóng vai trò quan trọng trong việc tạo ra các ứng dụng công nghệ tiên tiến như trợ lý ảo, hệ thống dịch tự động và nhiều sản phẩm truyền thông khác. Qua đó, môn học mở ra cơ hội cho sinh viên phát triển khả năng phân tích và giải quyết các bài toán thực tiễn trong lĩnh vực trí tuệ nhân tạo và xử lý ngôn ngữ tự nhiên.

1.2 Định Hướng Nghiên Cứu:

Trong vài năm gần đây, việc sử dụng các kỹ thuật tăng cường dữ liệu (data augmentation) đã trở thành một xu hướng quan trọng nhằm cải thiện hiệu suất của các hệ thống Automatic Speech Recognition (ASR). Nhóm chúng tôi lựa chọn nghiên cứu hướng ứng dụng của data augmentation dựa trên ba bài báo có uy tín sau:

- **SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition (Interspeech 2019)** [1]
  Phương pháp SpecAugment đơn giản nhưng hiệu quả, đã được chứng minh qua việc cải thiện độ chính xác của hệ thống ASR bằng cách áp dụng các biến đổi trực tiếp trên dữ liệu âm thanh.
- **Data Augmentation Methods for End-to-End Speech Recognition on Distant-Talk Scenarios (Interspeech 2021**) [2]
  Bài báo này tập trung vào việc tăng cường dữ liệu cho các tình huống giao tiếp xa mic, nơi mà yếu tố nhiễu và hiệu ứng âm học môi trường có thể ảnh hưởng lớn đến hiệu năng nhận dạng.
- **Data Augmentation for End-to-End Silent Speech Recognition for Laryngectomees (Interspeech 2022)** [3]
  Ứng dụng của kỹ thuật augmentation trong nhận dạng giọng nói thầm, một bài toán đòi hỏi sự tinh chỉnh đặc thù nhằm hỗ trợ người dùng sau phẫu thuật cắt thanh quản.

  Nhờ sự kết hợp giữa các phương pháp trên, nhóm hướng tới mục tiêu phát triển một hệ thống ASR có khả năng tổng quát hóa cao, ứng dụng hiệu quả trong các điều kiện thực tế đa dạng và phức tạp. Các kỹ thuật tăng cường dữ liệu không

chỉ giúp mở rộng tập huấn luyện mà còn tăng cường độ bền vững của mô hình trước các biến đổi không mong đợi trong tín hiệu âm thanh.

1.3 Tổng quan bài toán

Bài toán Automatic Speech Recognition (ASR) hiện nay đang đối mặt với nhiều thách thức, đặc biệt khi hệ thống phải hoạt động trong các môi trường có điều kiện âm học không lý tưởng. Các vấn đề thường gặp bao gồm:

- **Tín hiệu âm thanh nhiễu và bị méo:**
  Khi nguồn thu âm không đạt chuẩn (xa mic, có tiếng ồn nền), chất lượng dữ liệu đầu vào giảm, ảnh hưởng tiêu cực đến quá trình nhận dạng.
- **Tình huống "silent speech":**
  Ở một số trường hợp, như ở đối tượng sau phẫu thuật cắt thanh quản, người nói có thể không tạo ra tín hiệu âm thanh rõ ràng. Điều này đòi hỏi các kỹ thuật đặc biệt để "học" được các đặc trưng ẩn từ dữ liệu không truyền thống.
- **Đa dạng hóa dữ liệu:**
  Sự đa dạng trong cách phát âm, giọng nói và môi trường thu âm làm tăng độ phức tạp của bài toán, đòi hỏi hệ thống ASR phải được huấn luyện trên tập dữ liệu phong phú và đại diện.

  Nhóm chúng tôi đề xuất áp dụng các kỹ thuật data augmentation từ ba bài báo đã nêu nhằm giải quyết các thách thức trên. Mục tiêu là xây dựng một mô hình nhận dạng giọng nói không chỉ chính xác trong điều kiện kiểm soát mà còn ổn định và hiệu quả trong các tình huống thực tế khó khăn. Qua đó, dự án không chỉ góp phần nghiên cứu về mặt học thuật mà còn có tiềm năng ứng dụng cao trong các hệ thống hỗ trợ người dùng trong môi trường giao tiếp đa dạng.

Chương 2. BÀI TOÁN LỰA CHỌN

2.1 Giới thiệu bài toán Automatic Speech Recognition (ASR)

Automatic Speech Recognition (ASR) là bài toán chuyển đổi tín hiệu âm thanh chứa giọng nói (speech audio signals) thành văn bản tương ứng một cách tự động thông qua các thuật toán hoặc mô hình học máy.

Các bước cơ bản của ASR:

- Thu nhận dữ liệu âm thanh
- Tiền xử lý âm thanh
- Nhận dạng và chuyển đổi
- Xử lý hậu kỳ

2.2 Trình bày bài báo nhóm lựa chọn

2.2.1 SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition [1]

Tác giả: Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, Quoc V. Le

Hội nghị: INTERSPEECH 2019

Nội dung chính:

Bài báo đề xuất một phương pháp tăng cường dữ liệu đơn giản nhưng hiệu quả trong nhận dạng tiếng nói tự động, bằng cách biến đổi trực tiếp các đặc trưng phổ âm thanh giúp cải thiện hiệu suất nhận dạng

2.2.2 Data Augmentation Methods for End-to-End Speech Recognition on Distant-Talk Scenarios [2]

Tác giả: Emiru Tsunoo, Kentaro Shibata, Chaitanya Narisetty, Yosuke Kashiwagi, Shinji Watanabe

Hội nghị: Interspeech 2021

Nội dung chính:

Bài báo đề xuất ba phương pháp tăng cường dữ liệu nhằm cải thiện độ chính xác của hệ thống nhận dạng tiếng nói đầu-cuối (E2E ASR) trong môi trường nói từ xa và nhiễu: sử dụng dữ liệu tổng hợp TTS, chuyển đổi âm thanh bằng Cycle-GAN, và tạo nhãn giả bằng mô hình ASR huấn luyện trước. Thử nghiệm trên bộ dữ liệu CHiME-6 và CHiME-4 cho thấy từng phương pháp đều giúp nâng cao hiệu suất so với SpecAugment, và sự kết hợp ba phương pháp mang lại giảm lỗi nhận dạng đáng kể.

2.2.3 Data Augmentation for End-to-end Silent Speech Recognition for Laryngectomees [3]

Tác giả: Beiming Cao, Kristin Teplansky, Nordine Sebkhi, Arpan Bhavsar, Omer T. Inan, Robin Samlan, Ted Mau, Jun Wang

Hội nghị: Interspeech 2022

Nội dung chính:

Bài báo trình bày nghiên cứu mới về tăng cường dữ liệu cho mô hình SSR đầu-cuối hỗ trợ bệnh nhân cắt thanh quản. Phương pháp che khung thời gian liên tiếp cho hiệu quả tốt nhất và là cách tiếp cận mới trực tiếp trên tín hiệu vận động phát âm.

2.3 Tóm tắt những nội dung chính của bài báo

2.3.1 SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition [1]

- **Giới thiệu chung:**

Bài báo đề xuất SpecAugment, một phương pháp tăng cường dữ liệu cho nhận dạng tiếng nói. Phương pháp này áp dụng trực tiếp lên phổ log mel thay vì âm thanh thô. SpecAugment giúp cải thiện hiệu suất mô hình Listen, Attend and Spell (LAS) mà không cần thêm dữ liệu. Nó đạt kết quả vượt trội trên các bộ dữ liệu chuẩn như LibriSpeech 960h và Switchboard 300h.

- **Phương pháp chính:**

- Tác giả đã xây dựng các kĩ thuật tang cường áp dụng trực tiếp lên log mel spectrogram:
  - **Time Warping**: làm biến dạng trục thời gian bằng cách dịch các điểm dữ liệu trên trục thời gian.
  - **Frequency Masking**: che đi một dải tần số liên tục bất kỳ.
  - **Time Masking**: che đi một khoảng thời gian liên tục trong dữ liệu.
- Các kỹ thuật này được áp dụng theo cách thủ công như **LB**, **LD**, **SM**, **SS**, tùy vào bộ dữ liệu và mức độ biến dạng mong muốn.
- Ngoài ra, tác giả còn áp dụng mô hình LAS, tối ưu với lịch học có ba giai đoạn (tăng tốc, giữ nguyên, giảm dần), kết hợp label smoothing và shallow fusion để kết hợp với mô hình ngôn ngữ (Language Model).

- **Ưu điểm, hạn chế:**

1. **Ưu điểm:**
   - Đơn giản, hiệu quả: chỉ áp dụng lên spectrogram, không cần xử lý âm thanh thô hay dữ liệu mới.
   - Tăng khả năng khái quát hóa: chuyển từ bài toán overfitting sang underfitting.
   - Tương thích tốt với mô hình End to End như LAS

- Hiệu suất cao: đạt hiệu suất cao trên cả LibriSpeech và Switchboard ngay cả khi không dùng mô hình ngôn ngữ

2. **Hạn chế:**
   - Time wrapping ít ảnh hưởng và tính toán tốn kém
   - Label Smoothing có thể bất ổn khi kết hợp với learning rate schedule có giảm tốc
   - Các policy augmentation hiện tại vẫn đang thủ công chưa tự động

- **Kết luận:**

SpecAugment là một bước tiến quan trọng trong tang cường dữ liệu cho nhận dạng tiếng nói, chứng minh rằng các kỹ thuật đơn giản nếu áp dụng đúng cách sẽ mang lại hiệu quả cao.

2.3.2 Data Augmentation Methods for End-to-End Speech Recognition on Distant-Talk Scenarios [2]

- **Giới thiệu chung:**

Bài báo tập trung vào việc cải thiện hiệu quả của các hệ thống End to End ASR trong môi trường nói từ xa và có nhiều nhiễu (distant-talk) – một bối cảnh đặc biệt do khó khăn thiếu dữ liệu và chất lượng ghi âm kém. Tác giả áp dụng và kết hợp ba phương pháp tăng cường dữ liệu để cải thiện độ chính xác mô hình RNN-T và Conformer trên các tập dữ liệu CHiME-6 và CHiME-4, hai tập dữ liệu tiêu chuẩn trong nghiên cứu nhận dạng tiếng nói trong môi trường thực.

- **Phương pháp chính:**

- Ba phương pháp tăng cường dữ liệu chính được đề xuất là:
  - **TTS Augmentation:** Tạo dữ liệu nói mới từ văn bản bằng mô hình chuyển văn bản thành tiếng nói (Transformer TTS), kết hợp điều kiện giọng nói và phong cách (x-vector và GST).
  - **Cycle-Gan Augmentation:** Dùng Cycle-GAN để chuyển đổi đặc trưng âm thanh sạch sang đặc trưng nhiễu tương tự điều kiện thực tế, đặc biệt hữu ích khi không có dữ liệu ghép cặp.
  - **Pseudo-label Augmentation:** Sử dụng mô hình ASR huấn luyện trước để gán nhãn giả cho các đoạn âm thanh không rõ hoặc bị nhiễu, làm mềm phân phối nhãn trong tập huấn luyện.
- Các phương pháp trên được kết hợp cùng với SpecAugment và đánh giá hiệu quả thông qua độ chính xác (WER) trên CHiME-6 và CHiME-4.

- **Ưu điểm, hạn chế:**

1. **Ưu điểm:**
   - Cải thiện đáng kể WER, đặc biệt khi kết hợp cả ba phương pháp
   - Không yêu cầu dữ liệu song song cho Cycle-Gan, phù hợp với thực tế
   - Có thể áp dụng linh hoạt cho cả môi trường có nhiều nhiễu và dữ liệu ít

2. Hạn chế:

- TTS yêu cầu mô hình được huấn luyện trước, không thể huấn luyện từ tập dữ liệu nhiễu nhỏ.
- Pseudo-labels có thể chứa lỗi nếu không lọc kỹ, cần thêm bước lọc theo chỉ số CER.
- Chi phí tính toán đáng kể khi kết hợp nhiều phương pháp

- **Kết luận:**

Việc kết hợp nhiều phương pháp tăng cường như: TTS, Cycle-GAN, và Pseudo-label có thể cải thiện đáng kể hiệu suất nhận dạng tiếng nói đầu – cuối trong môi trường thực tế, nhiễu và dữ liệu hạn chế.

2.3.3 Data Augmentation for End-to-end Silent Speech Recognition for Laryngectomees [3]

- **Giới thiệu chung:**

Bài báo nghiên cứu phương pháp tăng cường dữ liệu nhằm cải thiện hiệu suất của hệ thống nhận dạng Silent Speech Recognition (SSR) dành cho người bị cắt thanh quản (laryngectomees). Vì việc thu thập dữ liệu cử động phát âm rất khó và tốn công, bài toán SSR cho laryngectomees gặp nhiều giới hạn về dữ liệu. Bài báo nghiên cứu đã áp dụng và so sánh nhiều kỹ thuật tăng cường trực tiếp trên tín hiệu chuyển động cơ học từ thiết bị đo EMA.

- **Phương pháp chính:**

- Bao gồm 5 phương pháp chính:
  - **Consecutive Time Masking (CTM):** che một đoạn thời gian liên tục (giống SpecAugment)
  - **Intermittent Time Masking (ITM):** che từng đoạn nhỏ, rời rạc theo thời gian.
  - **Articulatory Dimension Masking (ADM):** che các chiều chuyển động của bộ máy phát âm (ví dụ: trục lưỡi lên-xuống).
  - Sinusoidal Noise Injection (SNI): chèn nhiễu dạng sóng sin vào tín hiệu.
  - **Random Scaling (RS):** kéo giãn/ngắn thời gian của mẫu tín hiệu mô phỏng thay đổi tốc độ nói.
- Các kỹ thuật trên được huấn luyện trên mô hình End to End DeepSpeech2 sử dụng CTC loss, với đầu vào là chuyển động 2D từ 4 điểm đo (lưỡi và môi), đầu ra là chuỗi phoneme.

- **Ưu điểm, hạn chế**

1. Ưu điểm:

- Các kỹ thuật được áp dụng trực tiếp lên tín hiệu thô, giúp đơn giản hóa quá trình huấn luyện.

- Nghiên cứu đầu tiên ứng dụng hệ thống tăng cường này vào SSR cho laryngectomees, mở hướng cho các hệ thống trợ giúp phục hồi giọng nói.

2. Hạn chế:

- CTM không hiệu quả trong thiết lập speaker-independent (SI): không bù được khoảng cách giữa người bình thường và người bị mất tiếng.
- Chỉ thử nghiệm trên 2 người alaryngeal, nên khó tổng quát hóa.
- Hiệu quả của các kỹ thuật này trên dữ liệu từ các thiết bị khác (ví dụ: sEMG, ultrasound) chưa được kiểm chứng.

- **Kết luận:**

Bài báo nghiên cứu cho thấy rằng tăng cường dữ liệu, cụ thể là CTM có thể cải thiện rõ rệt hiệu suất SSR, đặc biệt khi được áp dụng trong giai đoạn fine-tuning của mô hình speaker-adaptive. Kết quả mở ra tiềm năng phát triển SSI cho người bị mất giọng và thúc đẩy nhu cầu nghiên cứu sâu hơn trên quy mô dữ liệu và người tham gia lớn hơn.

Chương 3. BÀI TOÁN CỤ THỂ

3.1 Giới thiệu chung

- Automatic Speech Recognition (ASR) là một trong những bài toán trọng yếu trong lĩnh vực xử lý ngôn ngữ tự nhiên và trí tuệ nhân tạo, với ứng dụng rộng rãi từ trợ lý ảo, nhập liệu bằng giọng nói cho đến điều khiển thiết bị thông minh. Độ chính xác của các hệ thống ASR phụ thuộc mạnh mẽ vào khả năng tổng quát hóa của mô hình học sâu trên dữ liệu huấn luyện. Do đó, một trong những kỹ thuật phổ biến để cải thiện hiệu năng mô hình là data augmentation – tức là mở rộng dữ liệu bằng cách biến đổi hoặc tạo mới các mẫu từ dữ liệu gốc.
- Tuy nhiên, đa số các phương pháp data augmentation truyền thống đòi hỏi phải biến đổi tín hiệu âm thanh gốc (raw waveform), điều này vừa làm tăng chi phí tính toán, vừa làm phức tạp pipeline huấn luyện. Trong bối cảnh đó, nhóm tác giả Park et al. (2019) đề xuất một phương pháp đơn giản nhưng hiệu quả mang tên SpecAugment, áp dụng trực tiếp trên log mel spectrogram, giúp mô hình tổng quát tốt hơn mà không cần thay đổi kiến trúc mạng hoặc thêm dữ liệu mới trong bài báo **SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition [1]**.

3.2 Các công trình nghiên cứu liên quan

Trước khi SpecAugment ra đời, đã có nhiều nỗ lực nhằm cải thiện hiệu suất của các hệ thống nhận dạng giọng nói tự động (ASR) thông qua kỹ thuật tăng cường dữ liệu. Một số phương pháp phổ biến bao gồm:

- **Noise addition**: Thêm nhiễu vào tín hiệu để mô phỏng môi trường thực.
- **Speed perturbation**: Thay đổi tốc độ của tín hiệu âm thanh.
- **Pitch shifting**: Thay đổi tần số cơ bản để tạo ra sự đa dạng trong phát âm.
- **Vocal tract length perturbation (VTLP)**: Thay đổi chiều dài ống âm thanh nhằm mô phỏng giọng nói của các cá thể khác nhau.

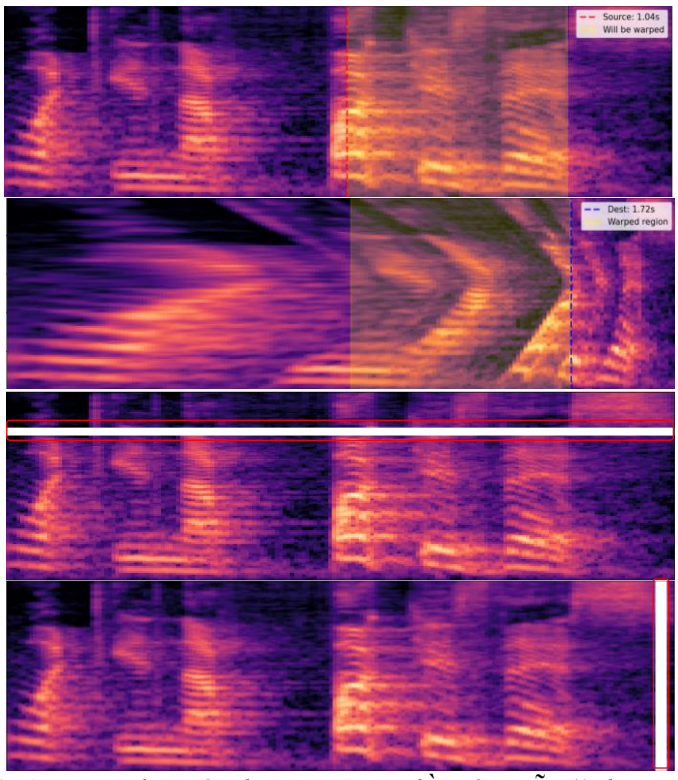
Tuy nhiên, những kỹ thuật này thường đòi hỏi xử lý tín hiệu ở mức sóng âm (waveform) khá phức tạp và làm tăng đáng kể chi phí tính toán.Để khắc phục điều này, các nhà nghiên cứu đã bắt đầu hướng đến các phương pháp tăng cường dữ liệu đơn giản hơn bằng cách tác động trực tiếp lên đặc trưng đầu vào của các mạng học sâu, chẳng hạn như phổ tần số (spectrogram). Một kỹ thuật tiền thân có liên quan là frequency masking trong các mô hình âm học CNN, như được đề cập trong **[4]**. Ở đó, các khối tần số liền kề được gom nhóm sẵn thành các "bin" và bị gán giá trị bằng 0 một cách ngẫu nhiên theo từng minibatch. Ngược lại, SpecAugment chọn ngẫu nhiên cả kích thước và vị trí của các vùng che theo tần số cho từng mẫu đầu vào riêng lẻ trong minibatch, giúp tăng mức độ đa dạng hóa. Bên cạnh đó, nhiều nghiên cứu khác cũng đã đề xuất các phương pháp loại bỏ có cấu trúc trong phổ tần số nhằm mục tiêu tăng cường mô hình, như được trình bày trong **[5]**.

3.3 Phương pháp

3.3.1 Các chiến lược Augment

Bài báo hướng đến việc xây dựng một chính sách tăng cường dữ liệu tác động trực tiếp lên phổ log-mel, nhằm giúp mạng học được các đặc trưng hữu ích. Dựa trên mục tiêu rằng các đặc trưng này cần phải có khả năng chống chịu với các biến dạng theo chiều thời gian, sự mất mát một phần thông tin tần số, và sự mất mát cục bộ các đoạn ngắn trong tín hiệu lời nói, chúng tôi đã lựa chọn các dạng biến dạng sau để tạo thành chính sách tăng cường dữ liệu:

- **Time warping:** biến dạng thời gian thông qua hàm `sparse_image_warp` của TensorFlow. Với một phổ log-mel có τ bước thời gian, coi nó như một hình ảnh trong đó trục thời gian nằm ngang và trục tần số nằm dọc. Một điểm ngẫu nhiên trên đường ngang đi qua trung tâm của hình ảnh, nằm trong phạm vi các bước thời gian $(W, \tau - W)$, sẽ được làm biến dạng sang trái hoặc phải với một khoảng cách ω được chọn từ phân phối đều trong khoảng từ 0 đến tham số biến dạng thời gian W. Có sáu điểm neo được cố định tại các biên - bao gồm bốn góc và hai điểm giữa của các cạnh dọc.
- **Frequency masking:** được áp dụng bằng cách che f kênh mel liên tiếp, cụ thể là các kênh từ $[f_0, f_0 + f)$, trong đó f được chọn ngẫu nhiên từ phân phối đều trong khoảng từ 0 đến tham số che tần số F, f_0 được chọn từ khoảng $[0, v - f]$ với v là số lượng kênh tần số mel.
- **Time masking:** được áp dụng bằng cách che t bước thời gian liên tiếp, cụ thể là đoạn $[t_0, t_0 + t]$, trong đó:
    - t được chọn ngẫu nhiên từ phân phối đều trong khoảng từ 0 đến tham số T (tham số che thời gian).
    - t_0 được chọn từ khoảng $[0, \tau - t)$ với τ là tổng số bước thời gian trong spectrogram.
    - Sử dụng thêm một tham số p (một hằng số nhỏ) để giới hạn vùng thời gian bị che mất t không quá dài để tránh mất quá nhiều thông tin vì vậy bài báo áp dụng ngưỡng độ rộng tối đa là $p \times \tau$.



Hình 1. *Augment được áp dụng trên một đầu vào mẫu (ảnh trên cùng). Từ trên xuống dưới, ảnh thể hiện phổ log mel của đầu vào mẫu, time warp, frequency mask and time mask áp dụng lần lượt.*

Việc masking nhiều lần (nhiều vùng bị che) trên trục tần số và trục thời gian là có thể. Các tác giả bài báo đã đưa ra các chiến lược tăng cường (augmentation policies) được tìm ra thủ công bằng tay: LibriSpeech basic (LB), LibriSpeech double (LD), Switchboard mild (SM) và Switchboard strong (SS), các tham số được tổng hợp trong ***Bảng 1***.

Policy	W	F	m_F	T	p	m_T
None	0	0	-	0	-	-
LB	80	27	1	100	1.0	1
LD	80	27	2	100	1.0	1
SM	40	15	2	70	0.2	2
SS	40	27	2	70	0.2	2

Bảng 1. *Các tham số của augment cho các chiến lược. m_F và m_T thể hiện cho số frequency và time mask được áp dụng*

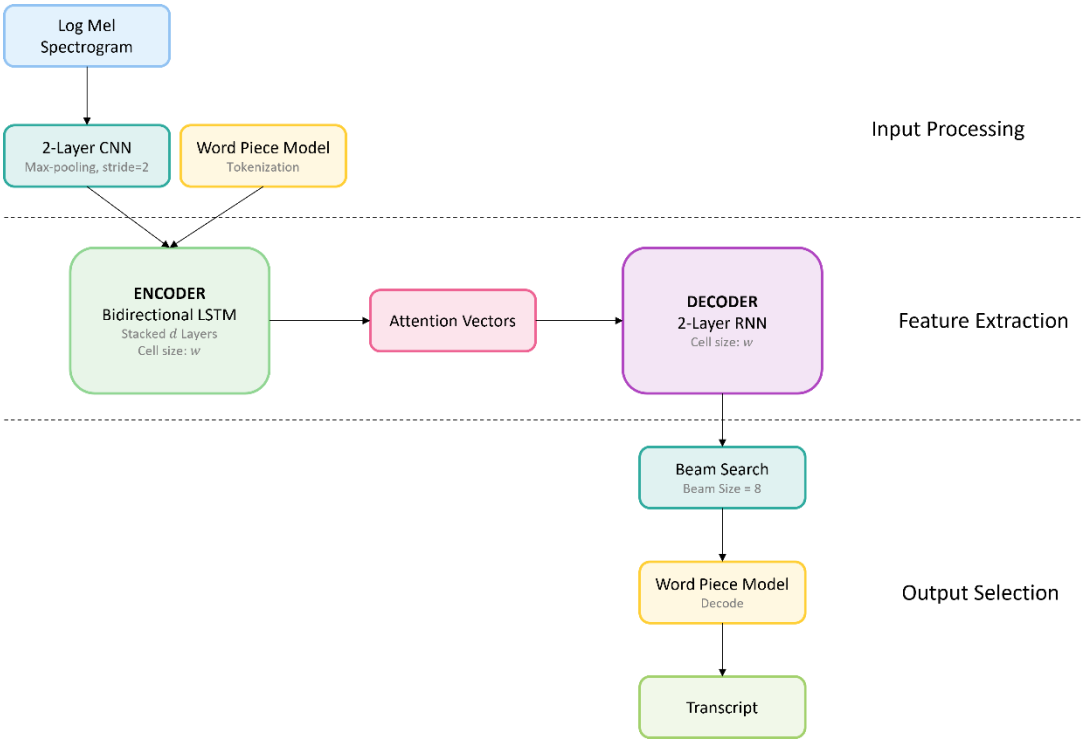
3.3.2 Mô hình

Nhóm tác giả bài báo sử dụng các mạng Listen, Attend and Spell (LAS) cho các tác vụ nhận dạng tiếng nói tự động (ASR). Các mô hình này, với đặc điểm là đầu-cuối, rất đơn giản để huấn luyện và có thêm lợi ích là đã có các bộ benchmark được công bố rộng rãi mà chúng tôi có thể xây dựng kết quả dựa trên đó. Trong phần này, chúng tôi sẽ xem xét lại các mạng LAS và giới thiệu một số ký hiệu để biểu diễn chúng. Chúng tôi cũng giới thiệu các lịch học mà chúng tôi sử dụng để huấn luyện các mạng, vì chúng hóa ra là yếu tố quan trọng trong việc quyết định hiệu suất. Cuối cùng, chúng

tôi xem xét phương pháp kết hợp mô hình nông, mà chúng tôi sử dụng để tích hợp các mô hình ngôn ngữ nhằm tăng thêm hiệu suất.

3.3.2.1. Kiến trúc LSA

Như đã nói, bài báo sử dụng mạng Listen, Attend and Spell (LAS) **[6]** cho các tác vụ ASR end-to-end được nghiên cứu trong **[7]**, với ký hiệu là LAS-d-w. Đầu vào là log mel spectrogram được đưa qua một mạng Neural Convolutional 2 lớp (CNN) với max-pooling và kích thước stride là 2. Đầu ra của CNN được đưa qua một encoder gồm d các LSTM hai chiều xếp chồng có kích thước ô là w để tạo ra một chuỗi attention vector. Các attention vector này sau đó được đưa vào một bộ giải mã RNN 2 lớp có kích thước ô là w, cho ra các token của transcript. Văn bản được tokenize bằng phương pháp Word Piece Model (WPM) **[8]**,với kích thước từ vựng là 16k cho LibriSpeech và 1k cho Switchboard. WPM cho LibriSpeech 960h được xây dựng bằng cách sử dụng tập transcript trainning. Với Switchboard 300h, transcript từ tập huấn luyện Fisher sẽ được kết hợp để xây dựng từ điển. Khi mô hình LAS (Listen, Attend and Spell) tạo ra kết quả, nó không tạo ra một câu hoàn chỉnh ngay lập tức, mà dự đoán từng token (từng chữ cái hoặc subword) theo từng bước. Tuy nhiên, có nhiều cách để chọn chuỗi ký tự đầu ra sao cho hợp lý nhất. Ở đây, tác giả sử dụng phương pháp Beam Search với beam size 8. So **[8]**, thì 'large model' của họ là mô hình LSA-4-1024 trong bài báo này.



Hình 2. *Kiến trúc mạng LSA được sử dụng trong SpecAugment*

3.3.2.2. Learning rate schedules

Learning rate schedules là một yếu tố quan trọng ảnh hưởng đến hiệu suất của các mô hình nhận dạng tiếng nói (ASR), đặc biệt khi có sử dụng augmentation. Trong nghiên cứu này, tác giả đưa ra các lịch trình huấn luyện với hai mục tiêu:

- Kiểm chứng rằng một lịch trình dài hơn giúp cải thiện hiệu suất cuối cùng của mô hình - đặc biệt khi có sử dụng SpecAugment.
- Tối đa hóa hiệu suất bằng cách sử dụng các lịch trình huấn luyện dài.

Mô hình sử dụng một learning rate schedule gồm 3 giai đoạn:

- **Ramp-up:** Tăng learning rate từ 0 lên giá trị tối đa trong khoảng step $(\mathbf{0}, \boldsymbol{s_r})$.
- **Hold:** Giữ learning rate ở giá trị tối đa một khoảng bước $(\boldsymbol{s_r}, \boldsymbol{s_i})$.
- **Exponential decay:** trong khoảng $(\boldsymbol{s_i}, \boldsymbol{s_f})$ giảm dần learning rate theo cấp số mũ cho đến khi chỉ còn $\frac{1}{100}$ của giá trị tối đa, sau đó giữ nguyên .

Ngoài ra, có hai yếu tố thời gian khác. Đầu tiên là weight nose (nhiễu trọng số) thêm nhiễu với độ lệch chuẩn là 0.075 tại thời điểm $\boldsymbol{s_{noise}}$ và giữ nó cố định trong suốt thời gian trainning. Weight noise này được áp dụng trong khoảng bước s_r đến s_i, tức là trong giai đoạn mà learning rate nằm ở mức cao (high plateu). Thứ hai, label smoothing được áp dụng đồng đều với mức độ bất định là 0.1, tức là nhãn đúng được gán xác suất 0.9, còn xác suất của các nhãn còn lại được chia đều cho các nhãn còn lại. Tuy nhiên label smoothing có thể làm bất ổn quá trình huấn luyện nếu learning rate quá nhỏ, vì vậy nhóm tác giả đôi khi chỉ áp dụng nó ở đầu quá trình huấn luyện và tắt khi bắt đầu giảm learning rate.

Có 2 kiểu learning rate schedule cơ bản:

- B(asic): $(s_r, s_{noise}, s_i, s_f)$ = (0.5k, 10k, 20k, 80k)
- D(ouble): $(s_r, s_{noise}, s_i, s_f)$ = (1k, 20k, 40k, 160k)

Nhóm tác giả đã đề xuất thêm một learning rate schedule dài hơn bởi vì nhận thấy có thể cải thiện được hiệu suất:

- L(ong): $(s_r, s_{noise}, s_i, s_f)$ = (1k, 20k, 140k, 320k)

3.3.2.3. Kết hợp Shallow Fussion với Mô hình Ngôn ngữ

Mặc dù đã đạt được hiệu quả cao khi dùng augmentation, tác giả nhận thấy vẫn có thể cải thiện thêm bằng cách dùng mô hình ngôn ngữ (Language Model). Trong quá trình decode hay inference, thay vì chỉ dùng xác suất từ mô hình nhận dạng giọng nói (ASR), ta cộng thêm xác suất từ mô hình ngôn ngữ (LM) theo công thức:

$$y^* = \text{argmax}_y(\log P(y|x) + \lambda \log P_{LM}(y))$$

Ý tưởng của việc này là dự đoán đầu ra không chỉ dựa vào âm thanh mà còn tận dụng kiến thức ngôn ngữ tự nhiên và vì chỉ thực hiện trong lúc test và không can thiệp vào quá trình học trọng số của mô hình ASR nên nó mới được gọi là shallow (nông) fussion. Ngoài ra, bài báo cũng dùng thêm coverage penalty với hệ số c để tránh mô hình tạo chuỗi quá ngắn hay bỏ sót thông tin. Dù không xuất hiện trực tiếp trong công thức nhưng phần này sẽ được cộng thêm vào điểm số tổng trong quá trình beam search decoding.

3.3.3 Các thực nghiệm trong bài báo

Trong phần này sẽ là miêu tả của các thực nghiệm trong bài báo trên hai bộ data LibriSpeech và Switchboard với SpecAugment. Kết quả thu về đạt được vượt trội so với các hệ thống hybrid được thiết kế công phu.

3.3.3.1. LibriSpeech 960h

Đối với LibriSpeech, nhóm tác giả sử dụng cấu hình như trong **[7]**, sử dụng đặc trưng phổ tần gồm 80 bộ lọc mel (filter banks) cùng với delta (đạo hàm bậc 1) và delta-delta acceleration (đạo hàm bậc 2) (tổng cộng 240 chiều đặc trưng), và một mô hình word-piece gồm 16.000 từ **[8]**.

Cả ba mô hình mạng LAS-4-1024, LAS-6-1024 và LAS-6-1280 và được huấn luyện trên LibriSpeech 960h với sự kết hợp của các chiến lược augement (None, LB, LD) trong **3.3.1** và các schedule khác nhau (B, D) trong **3.3.2.2**. Các thử nghiệm với các mô hình trên đều không áp dụng label smoothing và được thực hiện với learning rate ban đầu là 0.001, batch size 512, sử dụng 32 TPU của Google Cloud trong 7 ngày. Ngoài các chính sách tăng cường và lịch trình học, các siêu tham số khác được giữ cố định và không áp dụng thêm kỹ thuật nào.

Kết quả trên tập test được trình bày trong ***Bảng 2***.Kết quả cho thấy việc áp dụng SpecAugment giúp cải thiện rõ rệt hiệu suất của mô hình, đặc biệt với các mạng lớn hơn và thời gian huấn luyện dài hơn. Hiệu quả của lịch trình học dài càng rõ ràng khi sử dụng các chính sách tăng cường mạnh.

Network	Sch	Pol	NoLM		With LM	
			clean	other	clean	other
LAS-4-1024 [7]	B	-	4.7	13.4	3.6	10.3
	B	LB	3.7	10.0	3.4	8.3
	B	LD	3.6	9.2	2.8	7.5
LAS-4-1024	D	-	4.4	13.3	3.5	10.4
	D	LB	3.4	9.2	2.7	7.3
	D	LD	3.4	8.3	2.8	6.8
	D	-	4.5	13.1	3.6	10.3
LAS-6-1024	D	LB	3.4	8.6	2.6	6.7
	D	LD	3.2	8.0	2.6	6.5
	D	-	4.3	12.9	3.5	10.5
LAS-6-1280	D	LB	3.4	8.7	2.8	7.1
	D	LD	3.2	7.7	2.7	6.5

***Bảng 2**. LibriSpeech test WER(%) được đánh giá trên nhiều mạng, schedules và chiến lược khác nhau.*

Mạng LAS-6-1280 (lớn nhất trong các kiến trúc) được huấn luyện thêm với chính sách LD và lịch trình L (khoảng 24 ngày). Label smoothing được áp dụng khi số step nhỏ hơn 140k. Hiệu suất trên tập test được đánh giá tại checkpoint có kết quả tốt nhất trên tập dev. Mô hình này đạt được kết quả tốt nhất mà không cần sử dụng mô hình ngôn ngữ.

Method	No LM		With LM	
	clean	other	clean	other
HMM				
Panayotovetal.,(2015) [9]			5.51	13.97
Poveyetal.,(2016) [10]			4.28	
Hanetal.,(2017) [11]			3.51	8.58
Yangetal. (2018) [12]			2.97	7.50
CTC/ASG				
Collobertetal.,(2016) [13]	7.2			
Liptchinskyetal.,(2017) [14]	6.7	20.8	4.8	14.5
Zhouetal.,(2018) [15]			5.42	14.70
Zeghidouretal.,(2018) [16]			3.44	11.24
Lietal.,(2019) [17]	3.86	11.95	2.95	8.79
LAS				
Zeyeretal.,(2018) [18]	4.87	15.39	3.82	12.76
Zeyeretal.,(2018) [19]	4.70	15.20		
Irieetal.,(2019) [20]	4.7	13.4	3.6	10.3
Sabouretal.,(2019) [21]	4.5	13.3		
Paper				
LAS	4.1	12.5	3.2	9.8
LAS+ SpecAugment	2.8	6.8	2.5	5.8

***Bảng 3**. LibriSpeech test WER(%)*

3.3.3.2. Switchboard 300h

Với tập dữ liệu Switchboard 300h, bài báo sử dụng recipe "s5c" của Kaldi [40] để xử lý dữ liệu, nhưng điều chỉnh lại recipe để sử dụng đặc trưng âm thanh gồm 80 chiều filter bank kèm delta và delta-delta, dùng mô hình WPM với từ vựng gồm 1 nghìn từ [26] để mã hóa đầu ra, được xây dựng dựa trên sự kết hợp của từ vựng Switchboard và Fisher.

LAS-4-1024 được huấn luyện với các chính sách augmentation (None, SM, SS) và sử dụng lịch học schedule B. Như trước, learning rate là 0.001, batch size tổng là 512, và huấn luyện trên 32 TPU của Google. Các thí nghiệm này được thực hiện có và không dùng kỹ thuật label smoothing. Vì không có tập validation, mô hình được đánh giá tại điểm cuối cùng của schedule, tức là sau 100k bước với schedule B. Nhóm tác giả nhận thấy đường cong huấn luyện chững lại sau khi lịch giảm tốc độ học kết thúc (bước s_f), và hiệu năng của mạng không thay đổi nhiều. Hiệu quả của các chính sách augmentation khác nhau, với và không dùng label smoothing, trên tập Switchboard 300h được trình bày trong ***Bảng 4***. Dễ dàng thấy rằng làm mượt nhãn và augmentation có hiệu ứng cộng hưởng trong trường hợp này (CH là tập test của bộ CallHome).

Policy	LS	SWBD	CH
-	✗	12.1	22.6
	○	11.2	21.6
SM	✗	9.5	18.8
	○	9.1	16.1
SS	✗	9.7	18.2
	○	8.6	16.3

Bảng 4. *Switchboard 300h WER (%) được đánh giá với mô hình LAS-4-1024 huấn luyện theo schedule B, với các chính sách augmentation và LS khác nhau. Không sử dụng mô hình ngôn ngữ.*

Tương tự như với tập LibriSpeech 960h, LAS-6-1280 huấn luyện tập Switchboard 300h với lịch học L (thời gian huấn luyện khoảng 24 ngày) để đánh giá hiệu năng tốt nhất. Trong trường hợp này, nhóm tác giả nhận thấy rằng bật label smoothing trong suốt quá trình huấn luyện giúp cải thiện đáng kể kết quả. Kết quả được báo cáo tại điểm cuối của quá trình huấn luyện, sau 340k bước, trong bối cảnh kết quả được trình bày trong ***Bảng 5***. Shallow fussion kết hợp với mô hình ngôn ngữ huấn luyện từ Fisher-Switchboard cũng được áp dụng, với các tham số kết hợp được xác định từ tập RT-03 (bao gồm nhiều đoạn hội thoại từ Fisher, Switcherboard, CallHome). Không giống như trường hợp của LibriSpeech, các tham số kết hợp này không chuyển giao tốt giữa các mạng được huấn luyện khác nhau - ba mục trong Bảng 5 sử dụng các tham số kết hợp (λ, c) trong **3.3.2.3** lần lượt là (0.3, 0.05), (0.2, 0.0125), và (0.1, 0.025).

Method	No LM		With LM	
	SWBD	CH	SWBD	CH
HMM				
Vesel´ yetal.,(2013) [22]			12.9	24.5
Poveyetal.,(2016) [10]			9.6	19.3
Hadianetal.,(2018) [23]			9.3	18.9
Zeyeretal.,(2018) [18]			8.3	17.3
CTC/ASG				
Zweigetal.,(2017) [24]	24.7	37.1	14.0	25.3
Audhkhasietal.,(2018) [25]	20.8	30.4		
Audhkhasietal.,(2018) [26]	14.6	23.6		
LAS				
Luetal.,(2016) [27]	26.8	48.2	25.8	46.0
Toshniwaletal.,(2017) [28]	23.1	40.8		
Zeyeretal.,(2018) [18]	13.1	26.1	11.8	25.7
Wengetal.,(2018) [29]	12.2	23.3		
Zeyeretal.,(2018) [19]	11.9	23.7	11.0	23.1
Paper				
LAS	11.2	21.6	10.9	19.4
LAS + SpecAugment (SM)	**7.2**	14.6	**6.8**	14.1
LAS + SpecAugment (SS)	7.3	**14.4**	7.1	**14.0**

Bảng 5. *Switchboard 300h WER (%)*

3.3.4 Phân tích và kết luận của bài báo

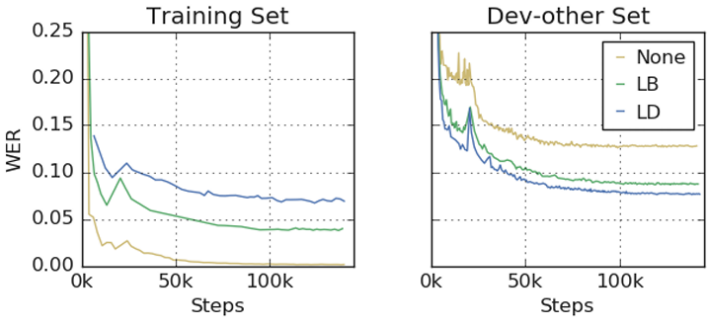
- ***Time warping có đóng góp nhưng không phải là nhân tố chính để nâng cao hiệu suất.***
  Trong ***Bảng 6***, thể hiện ba cách augement lần lượt bị tắt. Có thể thấy rằng ảnh hưởng của time warping là có nhưng không đáng kể.

W	F	mF	T	p	mT	$test-other$	$test$
80	27	1	100	1.0	1	10.0	3.7
0	27	1	100	1.0	1	10.1	3.8
80	0	-	100	1.0	1	11.0	4.0
80	27	1	0	-	-	10.9	4.1

Bảng 6. *Test set WER(%) không dùng LM với LAS-4-1024và schedule B*

- ***Label Smoothing gây bất ổn trong quá trình huấn luyện.***
  Nhận thấy rằng tỷ lệ các lần huấn luyện bị bất ổn tăng lên đối với tập LibriSpeech khi áp dụng label smoothing cùng với kỹ thuật tăng cường dữ liệu. Hiện tượng này trở nên rõ ràng hơn khi learning rate bắt đầu giảm. Do đó, nhóm tác giả đưa ra một lịch trình áp dụng label smoothing cho quá trình huấn luyện trên LibriSpeech, trong đó nhãn chỉ được làm trơn ở giai đoạn đầu của learning rate schedule.
- ***Augmentation chuyển một vấn đề overfitting trở thành underfitting.***

  Như có thể thấy từ các đường huấn luyện của mạng trong ***Hình 3***, các mạng trong quá trình huấn luyện không chỉ underfit (không khớp tốt) với hàm mất mát và WER trên tập dữ liệu tăng cường, mà còn underfit ngay cả trên tập huấn luyện gốc khi được huấn luyện với dữ liệu đã tăng cường. Điều này hoàn toàn trái ngược với tình huống thông thường, nơi mà các mạng có xu hướng overfit (khớp quá mức) với dữ liệu huấn luyện. Đây chính là lợi ích lớn nhất của việc huấn luyện với kỹ thuật tăng cường dữ liệu.



Hình 3. *LAS-6-1280 trên bộ LibriSpeech với schedule D.*

- **Các phương pháp phổ biến để xử lý underfitting mang lại cải thiện đáng kể.**
  Bài báo đã đạt được những cải thiện đáng kể về hiệu năng nhờ các phương pháp tiêu chuẩn nhằm giảm hiện tượng underfitting - như sử dụng mạng lớn hơn và huấn luyện lâu hơn. Hiệu suất hiện tại được báo cáo là kết quả của quá trình lặp đi lặp lại, bao gồm việc áp dụng chính sách tăng cường dữ liệu mạnh tay, sau đó mở rộng kiến trúc mạng (rộng hơn, sâu hơn) và huấn luyện với lịch trình dài hơn để giải quyết vấn đề underfitting.

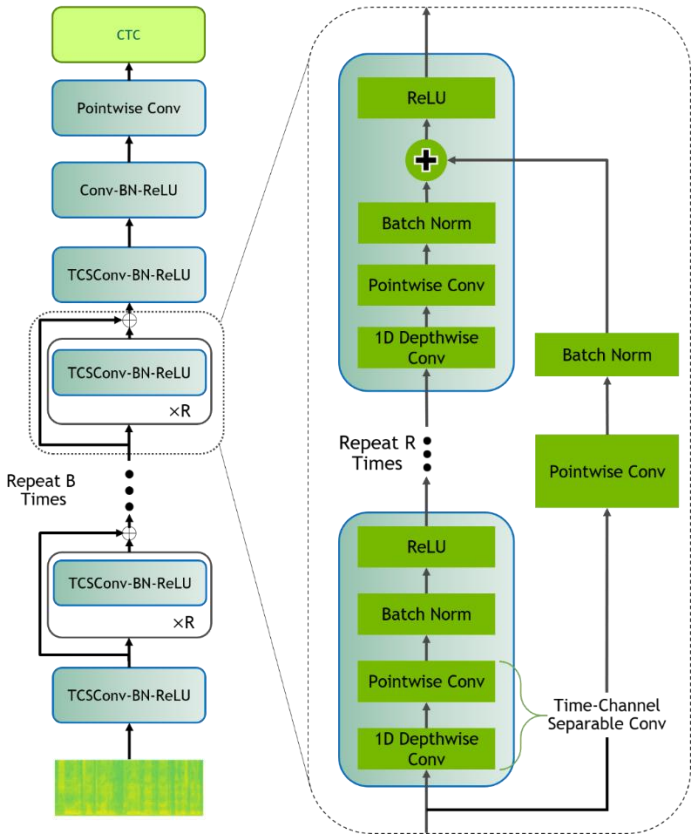
3.4 Quá trình thực nhiệm lại:

Trong báo cáo này, mô hình QuartzNet 15x5, một kiến trúc mạng tích chập một chiều (1D Convolutional Neural Network), được lựa chọn cho bài toán nhận dạng tiếng nói tự động (ASR). Mô hình này thuộc bộ công cụ NVIDIA NeMo và có dạng end-to-end, giúp đơn giản hóa quá trình huấn luyện và đạt hiệu quả cao khi kết hợp với các kỹ thuật tăng cường dữ liệu như SpecAugment.

Khác với mô hình LAS (Listen, Attend and Spell) được đề cập trong bài báo gốc nhưng không công bố mã nguồn cụ thể, QuartzNet 15x5 được ưu tiên lựa chọn nhờ có sẵn mô hình pretrained, dễ dàng tái tạo kết quả và đã được huấn luyện trên các tập dữ liệu chuẩn như LibriSpeech (có thể dựa vào các checkpoint **[30]** để thực hiện finetune cho Tiếng Việt).

3.4.1. Kiến trúc

QuartzNet 15x5 **[31]** là một mô hình nhận dạng tiếng nói end-to-end dựa hoàn toàn trên mạng tích chập một chiều (1D convolutional neural network). Mô hình được thiết kế nhằm tối ưu hóa hiệu quả tính toán và độ chính xác thông qua việc sử dụng các khối TCSConv-BN-ReLU (Time-Channel Separable Convolution), trong đó phép tích chập theo chiều thời gian và chiều kênh được tách biệt để giảm số lượng tham số.



Hình 4. *Kiến trúc QuartzNet15x5*

Đầu vào của hệ thống là biểu đồ phổ (log mel spectrogram), được đưa qua một chuỗi các tầng convolution, bao gồm:

- Một tầng **pointwise convolution** nhằm điều chỉnh số lượng kênh,
- Các khối **TCSConv-BN-ReLU**, mỗi khối bao gồm:
  - 1D depthwise convolution,
  - Pointwise convolution,
  - Batch normalization,
  - Hàm kích hoạt ReLU,
  - Một kết nối tắt (residual connection) giữa đầu vào và đầu ra của khối.

Cấu hình 15x5 cho biết mô hình bao gồm 15 khối cơ bản được lặp lại 5 lần.

Tầng đầu ra được nối với một tầng Connectionist Temporal Classification (CTC), cho phép ánh xạ chuỗi đầu vào theo thời gian thành chuỗi ký tự đầu ra mà không cần căn chỉnh (alignment) giữa âm thanh và nhãn

3.4.2. Kết hợp với SpecAugment

Để tăng độ chính xác và khả năng tổng quát hóa, kỹ thuật SpecAugment được áp dụng trong quá trình huấn luyện. SpecAugment thực hiện các phép biến đổi như masking theo trục thời gian và tần số trên spectrogram, giúp mô hình chống overfitting và học được các đặc trưng ổn định hơn từ dữ liệu âm thanh. Tuy nhiên nhóm không thử nghiệm sử dụng time warping vì nó là phương pháp tốn chi phí nhất nhưng không cho hiệu qua rõ rệt. Trong SpectrogramAugmentation module của NeMo ASR Configuration Files, và được cấu hình như sau:

```
model:
  ...
  spec_augment:
    _target_: nemo.collections.asr.modules.SpectrogramAugmentation
    # SpecAugment parameters
    freq_masks: 1
    freq_width: 27
    time_masks: 1
    time_width: 50
```

Đây không phải là tham số tối ưu chỉ dùng để có thể xem được cách mà SpecAugment ảnh hưởng đến hiệu suất của mô hình NeMo như thế nào trên các bộ data sử dụng.

3.4.3. Dữ liệu

Nhóm đã thực hiện trên hai bộ dữ liệu: LibreSpeech (Tiếng Anh) và VIVOS (Tiếng Việt) như sự hướng dẫn của thầy. Tuy nhiên với tài nguyên hạn chế vì phải train bằng GPU P100 của Kaggle (bị hạn chế chỉ train liên tục chỉ trong 12h) nên nhóm chỉ sử dụng bộ clean và 100 giờ của LibreSpeech còn bộ VIVOS vì tập train chỉ có 15 giờ (dễ để thể hiện hiệu quả của SpecAugment).

	Training	Testing
Speakers	251	40
Utterances	28539	2620
Duration	100	5.4

Bảng 7. *Thống kê của LibreSpeech clean-100*

	Training	Testing
Speakers	46	19
Utterances	11660	760
Duration	14:55	00:45

Bảng 8. *Thống kê của VIVOS*

Trong đó:

- Speakers: số lượng người nói
- Utterances: số lượng transcripts (dữ liệu)
- Duration: Tổng thời gian của file âm thanh tương ứng theo tập

3.4.4. Kết quả và phân tích

3.4.4.1. LibreSpeech-100

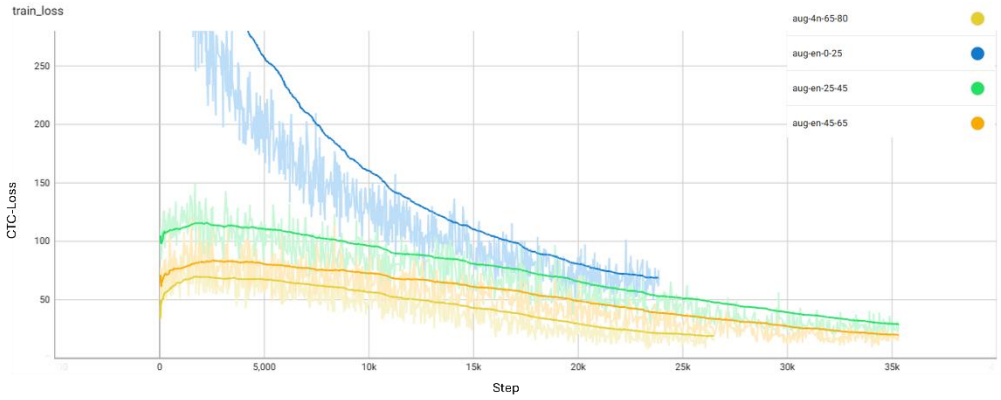
Do giới hạn về tài nguyên, nhóm chỉ có thể huấn luyện mô hình từ đầu (from scratch) trên bộ dữ liệu LibriSpeech-100 bằng cách chia nhỏ quá trình huấn luyện thành 4 lần train cho mỗi phương pháp: có và không sử dụng SpecAugment (tổng cộng 80 epoch).

Để đảm bảo tính nhất quán giữa các lần huấn luyện và giữ được momentum của quá trình tối ưu, nhóm đã sử dụng chung một lịch học (learning rate schedule) với các tham số sau:

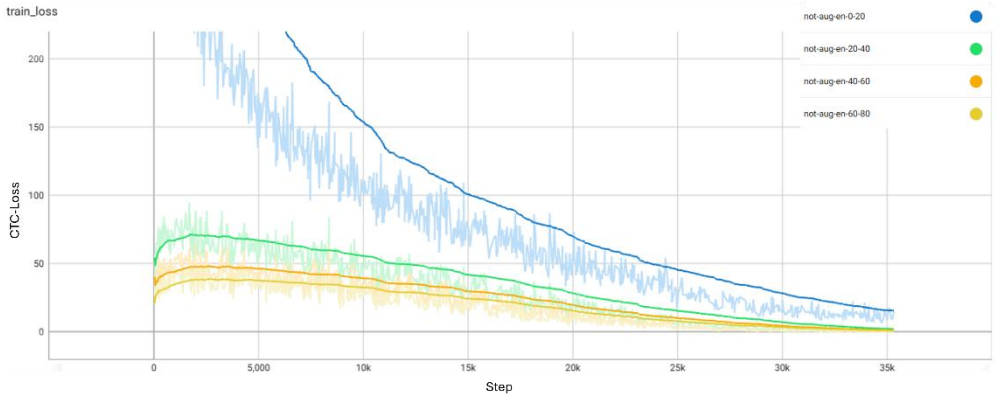
- Learning rate khởi đầu: 0.01
- Weight decay: 0.001
- Optimizer betas: [0.8, 0.5]

Việc sử dụng learning rate lớn ngay từ đầu có thể dẫn đến lãng phí một vài epoch đầu do mô hình chưa ổn định, tuy nhiên về tổng thể, chiến lược này cho thấy hiệu quả huấn luyện tốt hơn so với việc sử dụng learning rate nhỏ ngay từ đầu.

Việc train kéo dài với 35k step nên nhóm sẽ smoothing scalars 0.99 trong phần train loss để có thể dễ dàng nhìn và phân tích hơn. Do quá trình train các epoch từ 0-25 khi sử dụng SpecAugment các events bị gián đoạn do vượt quá thời gian train của Kaggle nên chỉ có giá trị của giai đoạn đầu.

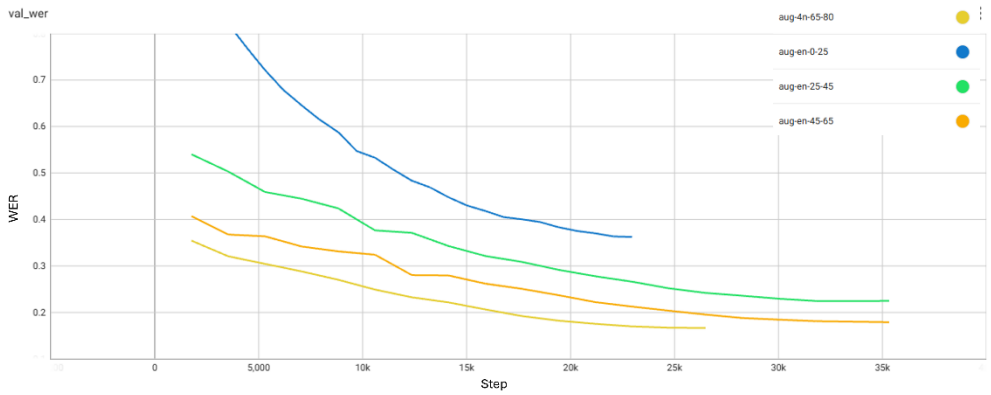


Hình 5. *Train loss trên LibreSpeech-100 khi sử dụng SpecAugment*

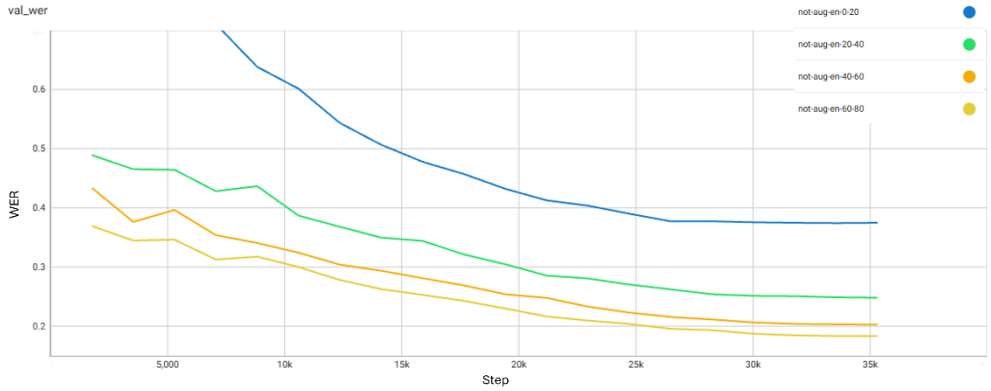


Hình 6. *Train loss trên LibreSpeech-100 khi không sử dụng Augment*

- Cả hai mô hình đều có xu hướng giảm CTC Loss theo thời gian, thể hiện rằng mô hình đang học và tối ưu dần.
- Mô hình không sử dụng SpecAugment có xu hướng giảm loss nhanh hơn trong giai đoạn đầu. Điều này có thể do dữ liệu không bị biến đổi, giúp mô hình học nhanh hơn từ các đặc trưng rõ ràng.
- Trong khi đó, mô hình có sử dụng SpecAugment giảm loss chậm hơn ở giai đoạn đầu. Tuy nhiên, sau khoảng 10k step, mô hình này thể hiện sự ổn định hơn, với loss giảm đều và ít dao động hơn so với mô hình không dùng SpecAugment.
- Điều này cho thấy SpecAugment mặc dù làm chậm quá trình học ban đầu do tăng độ phức tạp của dữ liệu, nhưng lại giúp mô hình học được các đặc trưng tổng quát hơn, ổn định hơn và có thể dẫn đến khả năng khái quát hóa tốt hơn trong giai đoạn về sau.



Hình 7. *Validation WER trên LibreSpeech-100 khi sử dụng SpecAugment*



Hình 8. *Validation WER trên LibreSpeech-100 khi không sử dụng Augment*

- Cả hai mô hình đều cho thấy xu hướng giảm WER (Word Error Rate) theo thời gian huấn luyện, chứng tỏ quá trình học diễn ra hiệu quả.
- Ở ảnh 1 (có sử dụng SpecAugment), các đường biểu diễn cho từng khoảng aug-en đều giảm đều, với độ dao động thấp. Tuy tốc độ giảm ban đầu không quá nhanh, nhưng càng về sau mô hình càng ổn định và cho kết quả tốt hơn.
- Trong khi đó, ảnh 2 (không sử dụng SpecAugment) có tốc độ giảm WER ban đầu nhanh hơn, đặc biệt rõ ở các khoảng not-aug-en-0-20. Tuy nhiên, sự dao động cao hơn thể hiện mô hình dễ bị ảnh hưởng bởi overfitting hoặc thiếu tính tổng quát.
- Đặc biệt, ở các bước sau 25k, các đường WER ở mô hình không dùng SpecAugment có xu hướng đi ngang sớm hơn, trong khi mô hình dùng SpecAugment vẫn còn cải thiện nhẹ, cho thấy hiệu quả dài hạn của phương pháp này.
- Điều này gợi ý rằng SpecAugment có thể làm chậm quá trình học ở giai đoạn đầu (do tăng tính khó của dữ liệu), nhưng lại giúp mô hình tổng quát hóa tốt hơn về sau, đặc biệt ở giai đoạn huấn luyện dài.

Nhận xét:

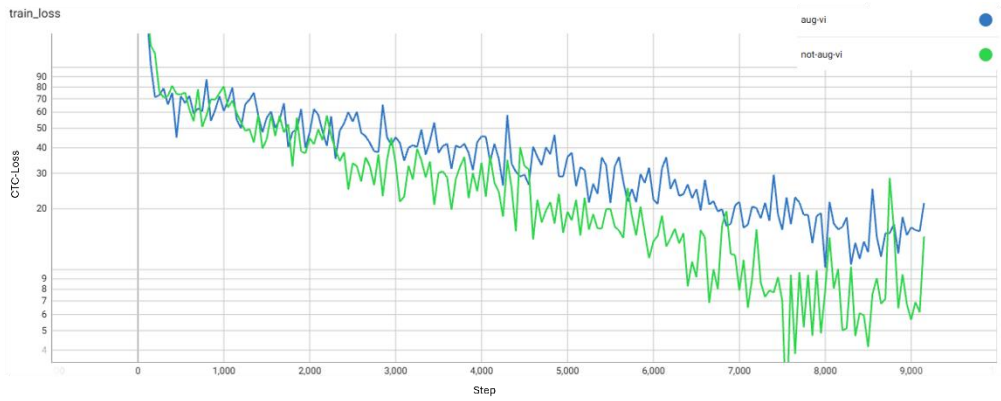
- Cả hai mô hình đều có xu hướng giảm CTC Loss và WER theo thời gian huấn luyện, cho thấy quá trình học hiệu quả.
- Mô hình không sử dụng SpecAugment học nhanh hơn ở giai đoạn đầu nhờ dữ liệu rõ ràng, nhưng loss dao động nhiều hơn, WER giảm nhanh rồi chững lại sớm, cho thấy khả năng overfitting.
- Mô hình sử dụng SpecAugment học chậm hơn lúc đầu do dữ liệu biến đổi nhiều hơn, nhưng giữ được sự ổn định trong quá trình huấn luyện, CTC Loss giảm đều và WER tiếp tục cải thiện ở giai đoạn sau.
- Điều này cho thấy SpecAugment đóng vai trò như một kỹ thuật regularization hiệu quả, giúp mô hình tổng quát hóa tốt hơn và giảm thiểu nguy cơ overfitting, đặc biệt khi huấn luyện trong thời gian dài.

Tiêu chí	Không Aug	Có SpecAugment
Train CTC Loss	Thấp hơn, dao động mạnh	Cao hơn, ổn định
Validation WER	Giảm nhanh ban đầu, chững lại sớm (~18.3%)	Giảm đều, xuống thấp hơn (~16.7%)
Overfitting	Có dấu hiệu xảy ra	Không có dấu hiệu rõ ràng
Khả năng tổng quát hóa	Thấp hơn	Cao hơn

Bảng 9. *So sánh hiệu năng trên LibreSpeech-100*

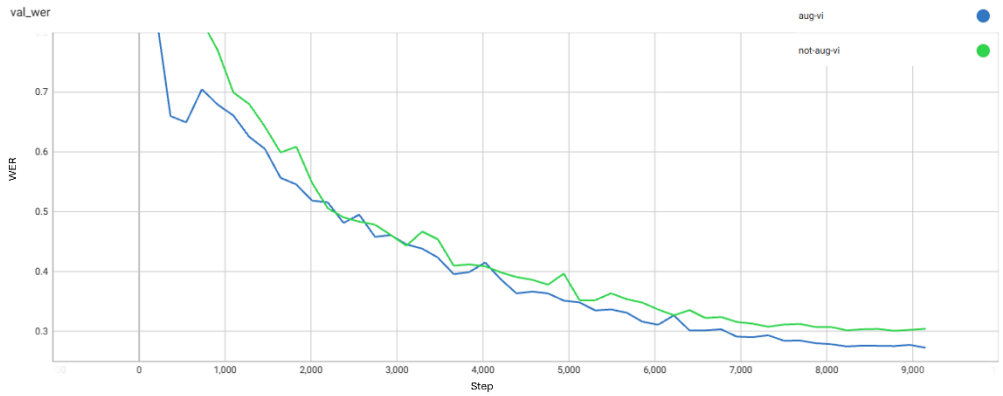
3.4.4.2. VIVOS Corpus

Bộ dữ liệu VIVOS không được train from scratch như trên bộ LibreSpeech-100 mà dùng pren model `stt_en_quartznet15x5` có sẵn của `nemo_asr.models.ASRModel` với 50 epoch.



Hình 9. *Train loss trên VIVOS*

- Cả hai mô hình đều có xu hướng giảm CTC Loss theo thời gian.
- Tuy nhiên, mô hình không dùng SpecAugment (xanh lá) có vẻ giảm loss nhanh hơn trong giai đoạn đầu.
- Mô hình có dùng SpecAugment (xanh dương) giảm loss chậm hơn nhưng ổn định hơn về sau, ít dao động hơn so với mô hình không dùng.
- Điều này cho thấy SpecAugment có thể làm chậm quá trình học ban đầu nhưng giúp mô hình học tổng quát tốt hơn về lâu dài.



Hình 10. *Validation WER trên VIVOS*

- Ở giai đoạn đầu (từ 0 đến khoảng 2.000 steps), cả hai mô hình đều có xu hướng giảm WER, tuy nhiên mô hình có SpecAugment (xanh dương) giảm nhanh và ổn định hơn.
- Từ 3.000 đến 9.000 steps, mô hình có SpecAugment tiếp tục cải thiện đều, trong khi mô hình không dùng SpecAugment bắt đầu chậm lại.
- Tại điểm cuối cùng (khoảng 9.000 steps), mô hình có SpecAugment đạt WER thấp hơn rõ rệt so với mô hình không dùng SpecAugment.

Nhận xét:

- Mô hình được huấn luyện với SpecAugment đạt WER tốt hơn trong toàn bộ quá trình huấn luyện, đặc biệt về giai đoạn sau, thể hiện khả năng tổng quát hóa tốt hơn.
- SpecAugment giúp mô hình tránh bị overfitting khi tiếp cận nhiều mẫu huấn luyện biến thể, từ đó giúp cải thiện hiệu suất trên tập validation.

Tiêu chí	Không Aug	Có SpecAugment
Train CTC Loss	Thấp hơn, dao động mạnh	Cao hơn, ổn định
Validation WER	Giảm chậm hơn (~30%)	Giảm đều, xuống thấp hơn (~27%)
Overfitting	Có dấu hiệu xảy ra	Không có dấu hiệu rõ ràng
Khả năng tổng quát hóa	Thấp hơn	Cao hơn

Bảng 10. *So sánh hiệu năng trên VIVOS*

3.4.4.3. Kết hợp với fusion

Thay vì dùng shallow fusion nhưng nhận thấy việc kết nối với mô hình QuartzNet khá là khó khăn (việc lấy được logits của từng token dự đoán theo vocab bên NeMo không được hỗ trợ) nên nhóm đã quyết định sử dụng deep fusion thay vì shallow fusion. Tức là bây giờ bây giờ kết quả sẽ bị ảnh hưởng ở decoder level không phải output level bằng cách thay đổi lớp decoder của mô hình.

Hình 11. *Thay đổi thành phần decoder (Bên phải: QuartzNet gốc, bên trái: kết hợp với RNN-T)*

Trong nghiên cứu này, nhóm đề xuất một kiến trúc hybrid kết hợp ưu điểm của QuartzNet encoder và RNN-T decoder. Thay vì sử dụng kiến trúc QuartzNet truyền thống với CTC decoder, chúng tôi thay thế phần decoder bằng RNN-T architecture bao gồm prediction network và joint network. Cụ thể, encoder QuartzNet được giữ nguyên với 15 layers sử dụng separable convolutions để trích xuất đặc trưng acoustic hiệu quả, trong khi decoder RNN-T với prediction network 1 layer (640 hidden units) đóng vai trò như một language model được tích hợp sâu vào kiến trúc. Joint network với 640 hidden units kết hợp thông tin từ encoder và decoder để tạo ra phân phối xác suất cuối cùng. Kiến trúc này cho phép end-to-end joint training thay vì shallow fusion, tận dụng khả năng xử lý song song của CNN encoder và khả năng modeling sequence dependencies của RNN-T decoder, đồng thời duy trì khả năng streaming inference của RNN-T.

Đối với bộ VIVOS, nhóm train thêm 10 epoch với decoder mới với mô hình sinh ra cuối cùng trong quá trình train trước đó (50 epoch). Nhóm cũng đã thử nghiệm Deep Fusion trên LibriSpeech-100, tuy nhiên do giới hạn thời gian và tài nguyên, chỉ huấn luyện được 5 epoch (trung bình 1.5 tiếng/epoch). Mô hình chưa hội tụ đầy đủ và kết quả đạt được chỉ gần tương đương với mô hình không dùng fusion. Nếu có thêm thời gian để huấn luyện dài hơn (ví dụ 10–20 epoch), hiệu suất rất có thể sẽ được cải thiện đáng kể. Tuy nhiên, nhóm chỉ có 1 tuần để hoàn thiện hệ thống, nên chưa thể đạt được kết quả tối ưu trên tập này. Ngoài ra, khi chạy vượt quá 5 epoch, hệ thống gặp lỗi xung đột bộ nhớ trong NeMo (lỗi nội tại) nên không thể xử lý tiếp. Nhóm cũng thử nghiệm freeze encoder và chỉ huấn luyện decoder, giúp giảm thời gian còn ~30 phút/epoch, nhưng mô hình lại không hội tụ ổn định, nên phương án này không được đưa vào kết quả chính thức cũng như phần demo.

Kết quả cũng nằm trong dự đoán của nhóm với sự cải thiện đáng kể về hiệu suất thể hiện trong . Có thể thấy việc áp dụng LM cho decoder có tác động đáng kể đến hiệu suất giống với tinh thần của bài báo.

	No Fussion		With Fussion	
	LibreSpeech-100	VIVOS	LibreSpeech-100	VIVOS
QuartzNet	0.1833	0.3006	-	0.2567
QuartzNet+SpecAug	0.1664	0.2723	-	0.2310

Bảng 11. *Bảng so sánh hiệu năng giữa các phương pháp*

Chương 4. KẾT LUẬN

Qua quá trình huấn luyện và đánh giá trên hai bộ dữ liệu LibriSpeech-100 và VIVOS, nhóm đã tiến hành so sánh hiệu quả của hai chiến lược huấn luyện: có và không sử dụng kỹ thuật SpecAugment, cũng như xem xét hiệu quả khi kết hợp với kỹ thuật Deep Fusion. Mặc dù việc huấn luyện từ đầu (from scratch) gặp nhiều giới hạn về tài nguyên, nhóm đã linh hoạt chia nhỏ quá trình huấn luyện và giữ được tính nhất quán bằng cách sử dụng chung lịch học với các tham số cố định. Điều này giúp đảm bảo rằng mọi khác biệt trong kết quả huấn luyện có thể được quy về ảnh hưởng của kỹ thuật SpecAugment một cách rõ ràng và khách quan.

Kết quả thực nghiệm trên cả hai bộ dữ liệu đều cho thấy một xu hướng nhất quán: **SpecAugment đóng vai trò như một kỹ thuật regularization hiệu quả**, giúp mô hình tránh overfitting và cải thiện khả năng tổng quát hóa, đặc biệt khi huấn luyện kéo dài. Cụ thể, trên bộ LibriSpeech-100, dù mô hình sử dụng SpecAugment có tốc độ học chậm hơn trong giai đoạn đầu, nhưng về sau lại thể hiện sự ổn định và cải thiện đều đặn về CTC Loss cũng như WER, đạt hiệu suất tốt hơn (WER ~16.6% so với ~18.3%). Trong khi đó, mô hình không sử dụng SpecAugment mặc dù học nhanh hơn ở đầu nhưng lại có xu hướng chững lại và dao động nhiều hơn, phản ánh nguy cơ overfitting.

Trên bộ VIVOS – một bộ dữ liệu tiếng Việt – khi sử dụng mô hình **QuartzNet15x5 pretrained**, kết quả cũng tương tự: mô hình sử dụng SpecAugment giảm WER ổn định hơn và đạt hiệu suất cao hơn về cuối (WER ~27% so với ~30%). Ngoài ra, các thử nghiệm áp dụng kỹ thuật **Fusion** (kết hợp với mô hình ngôn ngữ) đều cho kết quả tốt hơn rõ rệt so với khi không sử dụng, bất kể có áp dụng SpecAugment hay không.

Đáng lưu ý, khi làm việc với tiếng Việt – một ngôn ngữ có **hệ thống thanh điệu và dấu phụ phong phú** – mô hình gặp nhiều khó khăn hơn so với tiếng Anh do sự **đa dạng âm vị và sự phân biệt tinh tế giữa các từ thông qua dấu**. Điều này dẫn đến yêu cầu mô hình cần có khả năng phân biệt chi tiết và khái quát hóa cao hơn. Việc sử dụng **mô hình pretrained** đã giúp rút ngắn quá trình học và cải thiện độ chính xác đáng kể, tuy nhiên cũng đồng thời cho thấy rằng kỹ thuật tăng cường dữ liệu như SpecAugment là **cần thiết để chống lại hiện tượng overfitting và tăng độ bền vững của mô hình** trong điều kiện dữ liệu huấn luyện hạn chế.

Từ những quan sát trên, nhóm rút ra kết luận rằng **việc sử dụng kỹ thuật biến đổi dữ liệu như SpecAugment là cần thiết và mang lại lợi ích rõ rệt trong các bài toán nhận dạng tiếng nói**, đặc biệt là khi huấn luyện kéo dài hoặc với dữ liệu không quá lớn. Mặc dù có thể làm chậm quá trình học ban đầu, SpecAugment lại đóng vai trò quan trọng trong việc xây dựng mô hình có khả năng **tổng quát hóa tốt hơn** – điều đặc biệt quan trọng khi mô hình được triển khai thực tế và phải xử lý dữ liệu chưa từng thấy.

Tóm lại, SpecAugment không chỉ đơn thuần là một kỹ thuật tăng cường dữ liệu mà còn là một **phương pháp regularization mạnh mẽ giúp ổn định và nâng cao chất lượng mô hình ASR**, đặc biệt hiệu quả trong cả hai thiết lập: huấn luyện từ đầu và fine-tune từ mô hình có sẵn. Trong tương lai, nhóm dự kiến mở rộng huấn luyện với **các lịch học dài hơn (learning rate schedule) cả trong encoder và decoder** và áp dụng thêm **các kỹ thuật augment khác được đề xuất trong các nghiên cứu gần đây** như [2] và [3] để tiếp tục cải thiện hiệu suất, đặc biệt đối với các ngôn ngữ có cấu trúc âm vị phức tạp như tiếng Việt.

TÀI LIỆU THAM KHẢO

[1] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, Quoc V. Le, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," Interspeech, 2019.

[2] Emiru Tsunoo, Kentaro Shibata, Chaitanya Narisetty, Yosuke Kashiwagi, Shinji Watanabe, "Data Augmentation Methods for End-to-end Speech Recognition on Distant-Talk Scenarios," Interspeech, 2021.

[3] Beiming Cao, Kristin Teplansky, Nordine Sebkhi, Arpan Bhavsar, Omer Inan, Robin Samlan, Ted Mau, Jun Wang, "Data Augmentation for End-to-end Silent Speech Recognition for Laryngectomees," Interspeech, 2022.

[4] G. Kov´acs, L. T´oth, D. Van Compernolle, and S. Ganapathy, "In creasing the robustness of cnn acoustic models using autoregres sive moving average spectrogram features and channel dropout," in *vol. 100*, 2017, p. pp. 44–50.

[5] L. T´oth, G. Kov´acs, and D. Van Compernolle, "A perceptually inspired data augmentation method for noise robust cnn acoustic models," SPECOM, 2018.

[6] William Chan, Navdeep Jaitly, Quoc V. Le, Oriol Vinyals, "Listen, Attend and Spell," arXiv, 2015.

[7] K. Irie, R. Prabhavalkar, A. Kannan, A. Bruguier, D. Rybach, and, "Model Unit Exploration for Sequence-to-Sequence," arXiv, 2019.

[8] M. Schuster and K. Nakajima, "Japanese and korean voice search," ICASSP, 2012.

[9] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," n ICASSP, 2015.

[10] D. Povey, V. Peddinti, D. Galvez, P. Ghahrmani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," Interspeech, 2016.

[11] K. J. Han, A. Chandrashekaran, J. Kim, and I. Lane, "The CA PIO 2017 Conversational Speech Recognition System," arXiv, 2017.

[12] X. Yang, J. Li, and X. Zhou, "A novel pyramidal-FSMN archi tecture with lattice-free MMI for speech recognition," arXiv, 2018.

[13] R. Collobert, C. Puhrsch, and G. Synnaeve, "Wav2Letter: an End to-End ConvNet-based Speech Recognition System," arXiv, 2016.

[14] V. Liptchinsky, G. Synnaeve, and R. Collobert, "Letter-Based Speech Recognition with Gated ConvNets," arXiv, 2017.

[15] Y. Zhou, C. Xiong, and R. Socher, "Improving End-to-End Speech Recognition with Policy Learning," ICASSP, 2018.

[16] N. Zeghidour, Q. Xu, V. Liptchinsky, N. Usunier, G. Synnaeve, and R. Collobert, "Fully Convolutional Speech Recognition," arXiv, 2018.

[17] J. Li, V. Lavrukhin, B. Ginsburg, R. Leary, O. Kuchaiev, J. M. Cohen, H. Nguyen, and R. T. Gadde, "Jasper: An End-to-End Convolutional Neural Acoustic Model," arXiv, 2019.

[18] A. Zeyer, K. Irie, R. Schluter, and H. Ney, "Improved training of end-to-end attention models for speech recognition," Interspeech, 2018.

[19] A. Zeyer, A. Merboldt, R. Schluter, and H. Ney, "A comprehen sive analysis on attention models," IRASL, 2018.

[20] K. Irie, R. Prabhavalkar, A. Kannan, A. Bruguier, D. Rybach, and P. Nguyen, "Model Unit Exploration for Sequence-to-Sequence Speech Recognition," arXiv, 2019.

[21] S. Sabour, W. Chan, and M. Norouzi, "Optimal Completion Dis tillation for Sequence Learning," ICLR, 2019.

[22] K. Vesely, A. Ghoshal, L. Burger, and D. Povey, "Sequence discriminative training of deep neural networks," Interspeech, 2013.

[23] H. Hadian, H. Sameti, D. Povey, and S. Khudanpur, "End-to-end speech recognition using lattice-free MMI," Interspeech, 2018.

[24] G. Zweig, C. Yu, J. Droppo, and A. Stolcke, "Advances in All Neural Speech Recognition," ICASSP, 2017.

[25] K. Audhkhasi, B. Ramabhadran, G. Saon, M. Picheny, and D. Na hamoo, "Direct Acoustics-to-Word Models for English Conversa tional Speech Recognition," Interspeech, 2018.

[26] K. Audhkhasi, B. Kingsbury, B. Ramabhadran, G. Saon, and M. Picheny, "Building competitive direct acoustics-to-word mod els for english conversational speech recognition," ICASSP, 2018.

[27] L. Lu, X. Zhang, and S. Renals, "On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition," ICASSP, 2016.

[28] S. Toshniwal, H. Tang, L. Lu, and K. Livescu, "Multitask Learn ing with Low-Level Auxiliary Tasks for Encoder-Decoder Based Speech Recognition," Interspeech, 2017.

[29] C. Weng, J. Cui, G. Wang, J. Wang, C. Yu, D. Su, and D. Yu, "Improving Attention Based Sequence-to-Sequence Models for End to-End English Conversational Speech Recognition," Interspeech, 2018.

[30] "NVIDIA NeMo Models Catalog,NVIDIA NGC, 2024," NVIDIA, [Online]. Available: https://catalog.ngc.nvidia.com/models?query=nemo&orderBy=scoreDESC

[31] "QuartzNet - NVIDIA NeMo Documentation," NVIDIA, [Online]. Available: https://docs.nvidia.com/nemo-framework/user-guide/24.07/nemotoolkit/asr/models.html