**CÁC CÔNG NGHỆ ĐỘT PHÁ VÀ XU HƯỚNG TƯƠNG LAI TRONG MÔ HÌNH NHẬN DẠNG GIỌNG NÓI TỰ ĐỘNG (ASR SOTA 2022-2025)**

**I. Tổng quan về Hiện trạng Nhận dạng Giọng nói Tự động (ASR SOTA Landscape)**

**I.1. Sự chuyển dịch từ Mô hình Lai sang End-to-End và Foundation Models**

Ngành công nghiệp Nhận dạng Giọng nói Tự động (ASR) đã trải qua một sự chuyển đổi căn bản trong những năm gần đây. Các hệ thống lai truyền thống, vốn dựa vào việc phân tách rõ ràng giữa mô hình âm học (Acoustic Models - AM), mô hình phát âm (Pronunciation Models) và mô hình ngôn ngữ (Language Models - LM), đã dần được thay thế bởi các kiến trúc End-to-End (E2E).1 Sự chuyển dịch này không chỉ đơn giản hóa chuỗi xử lý (pipeline) mà còn cải thiện hiệu suất tổng thể bằng cách tối ưu hóa toàn bộ hệ thống dưới một hàm mất mát duy nhất.

Cùng với sự trưởng thành của các kỹ thuật E2E, khái niệm Mô hình Nền tảng (Foundation Models - FM) đã định hình lại ASR.2 Các mô hình này được huấn luyện trên dữ liệu âm thanh và văn bản quy mô lớn (massive scale) và có khả năng thích ứng cao với nhiều tác vụ khác nhau—không chỉ là chuyển lời nói thành văn bản (ASR), mà còn là dịch thuật giọng nói, tóm tắt, và nhận diện ngôn ngữ.3 Sự phát triển này là hệ quả trực tiếp của sự trưởng thành của kỹ thuật Học Biểu diễn Tự Giám sát (Self-Supervised Learning - SSL). SSL đã loại bỏ rào cản về dữ liệu gán nhãn quy mô lớn, cho phép các nhà nghiên cứu mở rộng quy mô kiến trúc và phạm vi ngôn ngữ một cách chưa từng có. Do đó, các công nghệ như Wav2Vec 2.0 và HuBERT trở thành nền tảng cho các mô hình ASR quy mô lớn hiện đại.

**I.2. Định hình Kiến trúc E2E Chủ đạo**

Các phương pháp E2E chủ đạo trong ASR bao gồm Connectionist Temporal Classification (CTC), RNN-Transducer (RNN-T), và mô hình Attention-based Encoder-Decoder (AED), thường được hiện thực hóa bằng kiến trúc Transformer.1 Mỗi phương pháp này đều có ưu điểm riêng, nhưng tất cả đều nhằm mục đích ánh xạ trực tiếp từ đặc trưng âm thanh đầu vào sang chuỗi văn bản đầu ra.

Trong bối cảnh ASR Streaming (thời gian thực), nơi độ trễ là yếu tố then chốt, kiến trúc lai (Hybrid) đã nổi lên như một giải pháp tối ưu. Kiến trúc Hybrid CTC/RNN-Transducer sử dụng một Encoder chung được chia sẻ giữa các bộ giải mã (Decoder).4 Việc này không chỉ giúp tăng cường độ chính xác nhận dạng mà còn tăng tốc độ hội tụ trong quá trình huấn luyện mô hình. Đặc biệt, đối với các ứng dụng yêu cầu độ trễ cực thấp, việc kết hợp các Decoder khác nhau trên cùng một Encoder cho phép hệ thống tận dụng lợi thế của cả hai phương pháp giải mã: tốc độ của CTC và khả năng mô hình hóa ngôn ngữ của RNN-T.4

**II. Các Đột phá trong Học Biểu diễn Tự Giám sát (Self-Supervised Learning - SSL)**

SSL đóng vai trò là xương sống cho sự thành công của các ASR Foundation Models, cung cấp các biểu diễn âm thanh chất lượng cao ($Speech Representations$) từ các tập dữ liệu không gán nhãn khổng lồ.

**II.1. Wav2Vec 2.0: Cơ chế Tiên phong cho ASR Hiện đại**

Wav2Vec 2.0, được giới thiệu bởi Facebook AI, đã thiết lập tiêu chuẩn cho SSL trong xử lý giọng nói.5 Kiến trúc này kết hợp một Mạng Tích chập (Convolutional Network) để trích xuất các đặc trưng âm thanh thô (raw features), theo sau là một bộ Transformer Encoder mạnh mẽ để học các biểu diễn ngữ cảnh dài.6

Một trong những đóng góp kỹ thuật quan trọng nhất là việc sử dụng kỹ thuật **Lượng tử hóa (Quantization)** và **Gumbel-Softmax**. Wav2Vec 2.0 chuyển đổi các đặc trưng âm thanh tiềm ẩn (latent features) thành các "mã từ" (codewords) rời rạc thông qua một ma trận lượng tử hóa để tạo ra các *logits* (điểm số). Kỹ thuật Gumbel-Softmax cho phép lấy mẫu một mã từ từ mỗi codebook một cách khả vi (fully differentiable), một bước cần thiết để tối ưu hóa mô hình bằng gradient descent.7 Kỹ thuật này tương tự như lấy *argmax* nhưng có thêm yếu tố ngẫu nhiên nhỏ được kiểm soát bởi tham số nhiệt độ, giúp hỗ trợ quá trình huấn luyện và tăng cường việc sử dụng các mã từ.7

Mục tiêu huấn luyện chính của Wav2Vec 2.0 là **Contrastive Loss (tổn thất đối nghịch)**.6 Mô hình được huấn luyện để phân biệt đặc trưng âm thanh đã che (masked speech features) với các đặc trưng nhiễu (negatives) trong ngữ cảnh. Đáng chú ý, tổn thất này chỉ được tính cho bước thời gian trung tâm của mask, tập trung vào việc dự đoán chính xác đơn vị tiềm ẩn tại vị trí đó.6

**II.2. HuBERT (Hidden Unit BERT): Đơn giản hóa Mục tiêu Huấn luyện SSL**

HuBERT ra đời sau Wav2Vec 2.0 và tìm cách đơn giản hóa quá trình huấn luyện SSL mà vẫn duy trì hiệu suất cao.8 Ý tưởng cốt lõi của HuBERT là khám phá các **Đơn vị Ẩn Rời rạc (Discrete Hidden Units)** để chuyển đổi dữ liệu giọng nói thành một cấu trúc có tính chất "giống ngôn ngữ hơn." Các đơn vị ẩn này có thể được so sánh với các token hoặc từ trong một câu văn bản.8

HuBERT đạt được điều này thông qua một quy trình **Clustering (phân cụm)** riêng biệt. Lấy cảm hứng từ phương pháp DeepCluster trong thị giác máy tính, HuBERT áp dụng clustering trên các đoạn âm thanh ngắn (25 mili giây) để tạo ra các *pseudo-labels*.8 Mô hình sau đó được huấn luyện để dự đoán các đơn vị ẩn này. Khi được tinh chỉnh (fine-tuned) cho nhận dạng giọng nói tự động, HuBERT đã chứng minh khả năng đạt hiệu suất ngang bằng hoặc cải thiện so với Wav2Vec 2.0.8

Sự khác biệt kỹ thuật chính so với Wav2Vec 2.0 nằm ở mục tiêu huấn luyện và cách tạo mục tiêu. HuBERT sử dụng **Cross-Entropy Loss (tổn thất Entropy Chéo)**, giống như trong mô hình BERT ban đầu. Hàm mất mát này đơn giản và ổn định hơn so với tổ hợp phức tạp Contrastive Loss và Diversity Loss của Wav2Vec 2.0, giúp quá trình huấn luyện trở nên dễ dàng và đáng tin cậy hơn.8 Sự chuyển dịch này đại diện cho một xu hướng kỹ thuật quan trọng: duy trì hoặc nâng cao chất lượng biểu diễn âm thanh trong khi tối ưu hóa sự phức tạp và độ ổn định của hàm mất mát.

**II.3. MMS: Đột phá về Quy mô Đa Ngôn ngữ**

Dự án Massively Multilingual Speech (MMS) đã chứng minh sức mạnh của SSL khi áp dụng ở quy mô cực lớn. Mục tiêu của MMS là mở rộng đáng kể phạm vi phủ sóng ngôn ngữ của công nghệ giọng nói từ khoảng một trăm ngôn ngữ lên 10 đến 40 lần, tùy thuộc vào tác vụ cụ thể.9

Thành phần chính để đạt được thành công này là việc xây dựng một tập dữ liệu mới dựa trên các bài đọc văn bản tôn giáo công khai và áp dụng hiệu quả kỹ thuật học tự giám sát.9 Kết quả là MMS đã xây dựng các mô hình wav2vec 2.0 pre-trained bao gồm 1,406 ngôn ngữ, và một mô hình nhận dạng giọng nói đa ngôn ngữ duy nhất cho 1,107 ngôn ngữ.10

Tác động định lượng của MMS là rất lớn. Các thử nghiệm cho thấy mô hình nhận dạng giọng nói đa ngôn ngữ của MMS có thể giảm hơn một nửa Tỷ lệ Lỗi Từ (WER) của mô hình Whisper trên 54 ngôn ngữ thuộc bộ tiêu chuẩn FLEURS.9 Kết quả này cung cấp một chiến lược rõ ràng cho các nhà phát triển ASR đa ngôn ngữ: việc ưu tiên các mô hình SSL được huấn luyện trên phạm vi ngôn ngữ cực lớn (như MMS) có thể mang lại hiệu suất vượt trội trên các ngôn ngữ tài nguyên thấp (low-resource languages) so với các mô hình tổng quát được giám sát (supervised generalist models).

**III. Phân tích Chuyên sâu về Mô hình Generalist ASR: OpenAI Whisper**

Whisper, được OpenAI giới thiệu vào cuối năm 2022, đã trở thành một Mô hình Nền tảng ASR (ASR Foundation Model) nổi tiếng nhờ khả năng tổng quát hóa và tính linh hoạt vượt trội.

**III.1. Kiến trúc Đa nhiệm (Multitask Architecture)**

Whisper sử dụng kiến trúc Sequence-to-Sequence Transformer, được huấn luyện trên một tập dữ liệu âm thanh đa dạng và quy mô lớn.11 Khác biệt cốt lõi của Whisper là khả năng đa nhiệm (multitasking). Nó được thiết kế không chỉ để nhận dạng giọng nói mà còn để thực hiện dịch giọng nói, nhận dạng ngôn ngữ nói (spoken language identification), và phát hiện hoạt động giọng nói (voice activity detection).11

Cơ chế này được hiện thực hóa thông qua việc biểu diễn các tác vụ khác nhau dưới dạng một chuỗi token mà bộ giải mã (decoder) phải dự đoán.11 Whisper sử dụng một tập hợp các *special tokens* hoạt động như các bộ chỉ định tác vụ (task specifiers) hoặc các mục tiêu phân loại. Ví dụ, một token có thể chỉ định rằng mô hình nên dịch giọng nói sang tiếng Anh thay vì chỉ phiên âm.11 Điều này cho phép một mô hình duy nhất thay thế nhiều giai đoạn của một chuỗi xử lý giọng nói truyền thống.

**III.2. Ưu điểm Vượt trội và Khả năng Ứng dụng**

Nhờ được huấn luyện trên một tập dữ liệu cực kỳ đa dạng, Whisper thể hiện khả năng **Tổng quát hóa (Generalization)** mạnh mẽ. Nó có thể nhận diện chính xác các giọng nói địa phương hoặc các biến thể ngữ âm trong cùng một ngôn ngữ (ví dụ: Anh-Mỹ, Anh-Anh).14 Điều này mang lại hiệu quả cao trong việc xử lý các tình huống giọng nói thực tế, không được chuẩn hóa.

Whisper không chỉ là một công cụ ASR thông thường mà còn sở hữu tính linh hoạt cao, cho phép ứng dụng trong nhiều lĩnh vực: từ phiên âm các cuộc họp, phỏng vấn, podcast đến hỗ trợ người khiếm thính và dịch thuật trực tiếp.14

**III.3. So sánh Định lượng: Whisper vs. SSL Chuyên biệt**

Mặc dù Whisper là một mô hình tổng quát mạnh mẽ, các nghiên cứu chuyên sâu đã chỉ ra rằng nó không phải là giải pháp tối ưu cho mọi tác vụ. Sự lựa chọn giữa Whisper và các mô hình SSL chuyên biệt như Wav2Vec 2.0 hoặc HuBERT phụ thuộc vào yêu cầu cụ thể của miền ứng dụng.

**Ưu thế của SSL:** Phân tích định lượng trên các bộ dữ liệu chuyên biệt cho thấy mô hình Wav2Vec2, khi được tinh chỉnh hiệu quả, đã vượt trội hơn mô hình Whisper-small trên hầu hết các chỉ số đánh giá, đạt độ khớp cao hơn và Tỷ lệ Lỗi Từ (WER) cũng như Tỷ lệ Lỗi Ký tự (CER) thấp hơn trên bộ dữ liệu Badini.15 Điều này nhấn mạnh rằng các mô hình SSL, khi được tối ưu hóa cho một ngôn ngữ hoặc miền cụ thể, vẫn có thể vượt qua tính tổng quát của Whisper.

**Điểm yếu Cụ thể:** So sánh giữa các mô hình cũng tiết lộ những điểm yếu riêng biệt. Wav2Vec-BERT có xu hướng mắc lỗi trong việc phân biệt phụ âm mũi, trong khi Whisper gặp khó khăn hơn với các cặp âm vị sát nhau, chẳng hạn như âm xát (fricative) và cặp âm bật hơi/không bật hơi (aspirated/unaspirated).16 Whisper cũng được ghi nhận là đôi khi mắc lỗi khi chuyển đổi số thành từ (number-to-word conversion) trong một số ngôn ngữ. Những điểm yếu khác nhau này cho thấy việc lựa chọn mô hình cần dựa trên phân tích âm vị học của ngôn ngữ mục tiêu và đặc điểm dữ liệu ứng dụng.16

Sự đối lập giữa độ chính xác chuyên biệt cao của Wav2Vec 2.0/HuBERT và tính tổng quát tuyệt vời của Whisper đòi hỏi sự cân nhắc chiến lược.

Table I: So sánh Kiến trúc ASR Hiện đại

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kiến trúc** | **Loại/Kiến trúc** | **Mục tiêu Huấn luyện Chính** | **Khả năng Đa ngôn ngữ** | **Lợi thế Cốt lõi** |
| **Wav2Vec 2.0** | Encoder Transformer | Contrastive Loss + Quantization 6 | Thấp (cần MMS để mở rộng) | Biểu diễn âm thanh SOTA, Độ chính xác cao khi Fine-tuned 15 |
| **HuBERT** | Encoder Transformer | Clustering + Cross-Entropy Loss 8 | Thấp (cần MMS để mở rộng) | Độ ổn định huấn luyện cao, Hiệu suất tương đương Wav2Vec 2.0 |
| **MMS (Wav2Vec 2.0 base)** | Encoder Transformer | SSL trên 1400+ ngôn ngữ 10 | Cực kỳ Cao | Giảm WER đáng kể trên ngôn ngữ tài nguyên thấp 9 |
| **Whisper** | Seq-to-Seq Transformer | Supervised Multi-tasking 11 | Cao (Generalization) | Tính linh hoạt, xử lý đa tác vụ, nhận diện đa giọng điệu 14 |
| **Conformer** | Conv-Augmented Transformer | Hybrid CTC/RNN-T (E2E) 4 | Tùy thuộc Fine-tuning | Tối ưu hóa cho Streaming/Latency, Cân bằng ngữ cảnh cục bộ/toàn cục 17 |

**IV. Conformer: Kiến trúc Tối ưu cho Hiệu suất và Streaming ASR**

Conformer, kiến trúc kết hợp giữa Convolutional Neural Network (CNN) và Transformer, đã trở thành tiêu chuẩn vàng cho ASR End-to-End, đặc biệt trong các môi trường yêu cầu hiệu suất cao và độ trễ thấp.

**IV.1. Cấu trúc "Macaron-Net Sandwich"**

Thiết kế kiến trúc của Conformer block được lấy cảm hứng từ Macaron-Net, tạo ra một cấu trúc kẹp (sandwich) các thành phần chính.17 Cụ thể, khối Conformer bao gồm hai mô-đun Feed Forward (FFN) kẹp mô-đun Multi-Headed Self-Attention (MHSA) và mô-đun Convolution ở giữa.

Kiến trúc này sử dụng **kết nối dư nửa bước (half-step residual weights)** trong các mô-đun FFN. Đối với đầu vào $x\_i$ của khối Conformer thứ $i$, mô-đun FFN đầu tiên áp dụng công thức $\tilde{x\_i} = x\_i + \frac{1}{2} FFN(x\_i)$. Mô-đun FFN thứ hai, đặt ở cuối, cũng sử dụng kết nối nửa bước và được theo sau bởi một lớp LayerNorm cuối cùng.17 Các nghiên cứu cho thấy việc có hai lớp FFN kiểu Macaron-Net kẹp các mô-đun chú ý và tích chập mang lại sự cải thiện đáng kể về hiệu suất nhận dạng giọng nói so với việc chỉ sử dụng một mô-đun FFN duy nhất.

Ngoài ra, Conformer tích hợp **Mã hóa Vị trí Tương đối (Relative Positional Encoding - RPE)** vào mô-đun MHSA, một kỹ thuật được thừa hưởng từ Transformer-XL.17 RPE là rất quan trọng vì nó cho phép mô-đun tự chú ý tổng quát hóa tốt hơn trên các độ dài đầu vào khác nhau, khiến bộ mã hóa (encoder) trở nên mạnh mẽ hơn đối với sự thay đổi của độ dài câu nói.17

**IV.2. Tầm quan trọng của Convolution Module**

Sự ra đời của Convolution Module là chìa khóa để giải quyết một hạn chế cố hữu của kiến trúc Transformer thuần túy. Trong khi Transformer xuất sắc trong việc mô hình hóa ngữ cảnh toàn cục (long-range global context), chúng lại kém hiệu quả trong việc trích xuất các mẫu đặc trưng cục bộ (fine-grained local feature patterns) trong chuỗi âm thanh.17

Convolution Module được thiết kế để hiệu quả nắm bắt các tương quan cục bộ dựa trên độ lệch tương đối.17 Trong khối Conformer, mô-đun này được xếp chồng **sau** mô-đun MHSA. Cấu trúc của Convolution Module bắt đầu bằng một cơ chế cổng (gating mechanism) bao gồm một phép tích chập pointwise và một đơn vị tuyến tính có cổng (GLU). Tiếp theo là một lớp tích chập depthwise 1-D duy nhất. Việc triển khai Batchnorm ngay sau phép tích chập hỗ trợ hiệu quả quá trình huấn luyện các mô hình sâu.17

Sự kết hợp chiến lược này—Transformer cho ngữ cảnh dài và Convolution cho ngữ cảnh ngắn—tạo ra một kiến trúc cân bằng, dẫn đến sự vượt trội về mặt lý thuyết và thực nghiệm cho ASR.

**IV.3. Tối ưu hóa cho ASR Streaming (FastConformer)**

Conformer là nền tảng cho nhiều mô hình ASR streaming (ASR thời gian thực) hiệu suất cao, chẳng hạn như Parakeet, dựa trên kiến trúc FastConformer.18 FastConformer là một phiên bản được tối ưu hóa, được thiết kế đặc biệt để giảm độ trễ (latency) trong môi trường phát trực tuyến.4

Việc tối ưu hóa độ trễ được thực hiện thông qua các kỹ thuật sau:

1. **Ràng buộc Ngữ cảnh (Context Constraints):** Giới hạn cả ngữ cảnh look-ahead (tương lai) và ngữ cảnh quá khứ trong Encoder.4 Điều này là cần thiết vì trong chế độ streaming, không có sẵn toàn bộ dữ liệu âm thanh tương lai.
2. **Cơ chế Activation Caching:** Một cơ chế bộ nhớ đệm kích hoạt được giới thiệu để cho phép Encoder phi tự hồi quy (non-autoregressive encoder) hoạt động tự hồi quy (autoregressively) trong quá trình suy luận. Điều này giải quyết sự khác biệt về độ chính xác giữa thời gian huấn luyện và thời gian suy luận, một vấn đề phổ biến đối với nhiều mô hình streaming.4

Ngoài ra, việc sử dụng kiến trúc **Hybrid CTC/RNN-Transducer** là một kỹ thuật tiêu chuẩn trong ASR streaming. Việc sử dụng Encoder chung với cả Decoder CTC và RNN-T giúp tăng cường độ chính xác nhận dạng và tiết kiệm tính toán. Kiến trúc lai này không chỉ tăng tốc độ hội tụ của Decoder CTC mà còn cải thiện độ chính xác của các mô hình streaming so với các mô hình Decoder đơn.4 Mặc dù việc ràng buộc ngữ cảnh tương lai là cần thiết để giảm độ trễ, các nhà phát triển cần lưu ý đến sự đánh đổi (trade-off) tiềm ẩn, vì việc hạn chế ngữ cảnh toàn cục có thể dẫn đến giảm nhẹ độ chính xác trong các tình huống mơ hồ về ngữ nghĩa ở cuối câu nói.

**V. Xu hướng Mới: Tích hợp LLM, Giảm Độ trễ và Tăng cường Ngữ cảnh**

Lĩnh vực ASR đang nhanh chóng hội tụ với Mô hình Ngôn ngữ Lớn (LLM), tạo ra các đột phá về ngữ nghĩa nhưng đồng thời đặt ra những thách thức mới về hiệu suất.

**V.1. LLM trong ASR: Tăng cường Ngữ nghĩa và Thách thức Latency**

LLM đã được áp dụng vào ASR bằng cách kết hợp bộ mã hóa âm thanh (audio encoder) để trích xuất các biểu diễn giọng nói (speech embeddings).19 Lợi ích chính là sự gia tăng đáng kể về độ chính xác nhận dạng và khả năng hỗ trợ phạm vi ngôn ngữ, phương ngữ và giọng điệu rộng hơn nhờ khả năng mô hình hóa ngôn ngữ cấp cao của LLM.19

Tuy nhiên, việc tích hợp LLM (như Speech-Llama 7B hay Seed-ASR >10B tham số) tạo ra một thách thức lớn về quy mô. Kích thước mô hình tăng lên đáng kể so với các bộ mã hóa âm thanh truyền thống (thường dưới 1B tham số).19 Điều này dẫn đến một nút cổ chai về hiệu suất: bản chất giải mã tự hồi quy (autoregressive decoding) của LLM khiến độ trễ giải mã trở thành trở ngại chính đối với hiệu suất, tiêu tốn nhiều thời gian suy luận hơn đáng kể so với bộ mã hóa âm thanh.19

**V.2. Giải mã Suy đoán (Speculative Decoding - SpecASR): Giải pháp cho Độ trễ**

Để chống lại nút cổ chai độ trễ do LLM gây ra, kỹ thuật Giải mã Suy đoán (Speculative Decoding) đã được áp dụng vào ASR (gọi là SpecASR).

**Nguyên tắc Cơ bản:** Kỹ thuật này sử dụng một mô hình ASR nhỏ hơn, nhanh hơn (Draft Model) để nhanh chóng tạo ra các token nháp theo cách tự hồi quy. Sau đó, mô hình Target (LLM lớn, chính xác hơn) được sử dụng để xác thực chuỗi nháp đó cùng một lúc. Các token nháp khớp với đầu ra của Target Model sẽ được chấp nhận, cải thiện hiệu suất của Target Model mà không làm giảm chất lượng nhận dạng.19

**SpecASR Đột phá:** Các cải tiến trong SpecASR, chẳng hạn như *dynamic tree-structure speculative decoding* và *draft sequence recycling*, đã được phát triển để tăng tỷ lệ chấp nhận xác minh và tính linh hoạt. Dữ liệu thử nghiệm cho thấy những kỹ thuật tiên tiến này có thể giảm đáng kể tổng thời gian giải mã (Total latency), ví dụ, từ khoảng 485.54ms xuống còn 367.79ms.19 Sự thành công của SpecASR trong ASR cho thấy rằng rào cản hiệu suất hiện tại không còn nằm ở việc trích xuất đặc trưng âm thanh mà là ở mô hình hóa ngôn ngữ cấp cao. Việc tối ưu hóa LLM (ví dụ: thông qua SpecASR) đã trở thành một kỹ năng cốt lõi cho các nhà phát triển ASR SOTA.

Table II: Phân tích Hiệu quả Giảm Độ trễ Giải mã ASR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kỹ thuật** | **Mô hình/Kiến trúc** | **Cơ chế Giảm Độ trễ** | **Hiệu quả Định lượng (Ví dụ SpecASR)** | **Bối cảnh Ứng dụng** |
| **Speculative Decoding** | LLM-based ASR (Target Model lớn) | Draft Model nhanh hơn dự đoán, Target Model xác thực song song [20] | Total Latency giảm từ 485.54ms xuống 367.79ms (cải tiến ~24%) 19 | Tối ưu hóa LLM Decoder, ứng dụng AI đàm thoại |
| **Streaming Conformer** | FastConformer (Parakeet) | Ràng buộc Look-ahead Context, Activation Caching 4 | Độ trễ thấp hơn so với mô hình buffered truyền thống 4 | ASR thời gian thực, Tổng đài/Hội nghị trực tiếp |
| **Self-Hosted LLMs** | Generic LLMs/ASR | Loại bỏ API request delays, tối ưu hóa hạ tầng 21 | Giảm đáng kể thời gian phản hồi (phụ thuộc vào setup) | Ứng dụng khối lượng lớn, yêu cầu kiểm soát độ trễ tuyệt đối 21 |

**V.3. Tối ưu hóa Hệ thống Thời gian Thực (Real-Time Systems)**

Trong các hệ thống tương tác giọng nói (voice agents), việc giảm thiểu độ trễ là tối quan trọng.22 Độ trễ không chỉ bao gồm tốc độ tạo token của mô hình mà còn là thời gian khởi động dịch vụ (startup latency). Việc tránh các độ trễ khởi động (cold start) là thiết yếu để duy trì trải nghiệm người dùng liền mạch.22

Một mô hình xử lý hiệu quả là **Xử lý Phân đoạn (Staged Processing)**. Hệ thống sẽ phát trực tuyến âm thanh đến ASR khi người dùng đang nói, đệm văn bản. Ngay sau khi phát hiện kết thúc lời nói (end-of-speech), văn bản tích lũy sẽ được gửi ngay lập tức đến LLM để bắt đầu tạo phản hồi. Quá trình Text-to-Speech (TTS) sẽ bắt đầu ngay khi các token đầu tiên của phản hồi được xuất ra. Kỹ thuật này đảm bảo độ trễ phản hồi được đẩy xuống mức tối thiểu, tạo ra các cuộc hội thoại tự nhiên hơn.22 Đối với các ứng dụng khối lượng lớn, việc chuyển sang các mô hình tự lưu trữ (self-hosted LLMs) được tối ưu hóa, chạy trên hạ tầng GPU/TPU được tối ưu hóa, cũng giúp giảm đáng kể độ trễ bằng cách tránh độ trễ yêu cầu API.21

**V.4. Tăng cường Ngữ cảnh Động (Dynamic Contextual Biasing)**

Một thách thức dai dẳng đối với các mô hình ASR E2E, kể cả các Foundation Models, là độ chính xác đối với các cụm từ hiếm (rare phrases) không xuất hiện thường xuyên trong dữ liệu huấn luyện, chẳng hạn như tên riêng, thương hiệu, hoặc thuật ngữ kỹ thuật.23

Để giải quyết vấn đề này, các phương pháp **Contextual Biasing (thiên vị ngữ cảnh)** động đã được phát triển. Một phương pháp là giới thiệu một mạng dự đoán cụm từ ngữ cảnh (contextual phrase prediction network). Mạng này sử dụng *contextual embeddings* để huấn luyện mô hình và tính toán *bias loss*.23 Phương pháp này đã đạt được cải thiện Tỷ lệ Lỗi Từ (WER) tương đối 12.1% so với mô hình baseline trên kho dữ liệu LibriSpeech và giảm 40.5% WER đối với chính các cụm từ ngữ cảnh đó.23

Một kỹ thuật tiên tiến khác cho phép thay đổi trọng số thiên vị "on the fly" tại thời điểm suy luận là sử dụng **Weighted Finite State Transducer (WFST)**. Điều khoản thiên vị được lắp ráp thành một biểu diễn WFST và mô hình ASR kết hợp WFST này ở mỗi bước giải mã.24 Điều này đảm bảo rằng mô hình có thể thích ứng tức thì với danh sách các từ khóa hoặc thuật ngữ chuyên biệt được cung cấp, một yếu tố bắt buộc để đạt được độ chính xác tối ưu trong các miền dữ liệu chuyên biệt.

**VI. Kết luận và Khuyến nghị Chiến lược Phát triển Mô hình**

Phân tích các công nghệ ASR đột phá từ năm 2022 đến 2025 cho thấy một sự hội tụ rõ ràng: ASR hiện đại không chỉ tập trung vào việc chuyển đổi âm thanh thành văn bản, mà còn là tối ưu hóa biểu diễn âm thanh thông qua SSL và tích hợp khả năng mô hình hóa ngôn ngữ cấp cao của LLM.

**VI.1. Lộ trình Lựa chọn Kiến trúc cho Dự án ASR**

Việc lựa chọn nền tảng kiến trúc phải được định hướng bởi mục tiêu ứng dụng cụ thể:

1. **Nếu Mục tiêu là Đa Ngôn ngữ và Ngôn ngữ Tài nguyên Thấp (Low-Resource):** Không nên chỉ dựa vào các mô hình tổng quát như Whisper. Thay vào đó, khuyến nghị sử dụng các mô hình Foundation được huấn luyện theo cơ chế SSL mở rộng quy mô, đặc biệt là **MMS dựa trên Wav2Vec 2.0 hoặc HuBERT**. MMS đã chứng minh khả năng giảm WER vượt trội trong các ngôn ngữ hiếm, do chiến lược huấn luyện tự giám sát trên hàng nghìn ngôn ngữ.9
2. **Nếu Mục tiêu là Tổng quát hóa Cao và Đa Tác vụ (Dịch/ID/ASR):** **Whisper** là lựa chọn tối ưu nhờ kiến trúc Seq-to-Seq Transformer đa nhiệm và khả năng xử lý các biến thể giọng nói địa phương một cách chính xác.11
3. **Nếu Mục tiêu là ASR Thời gian Thực (Real-Time/Streaming):** Bắt buộc phải sử dụng kiến trúc **Conformer/FastConformer** kết hợp Hybrid CTC/RNNT Decoding. Các kỹ thuật như Activation Caching và ràng buộc ngữ cảnh là cần thiết để đạt được hiệu suất độ trễ thấp tối ưu.4

**VI.2. Xu hướng Tương lai và Mô hình Nền tảng Thế hệ Kế tiếp**

Tương lai của ASR sẽ được định hình bởi ba xu hướng chính:

* **Foundation Models theo Hướng Dẫn (Instruction-Following ASR):** Nghiên cứu chỉ ra rằng các mô hình ASR có thể được huấn luyện để hiểu và thực hiện các hướng dẫn văn bản tự do cho các tác vụ ASR phức tạp, như thao tác bản ghi hoặc tóm tắt. Điều đáng chú ý là khả năng này có thể đạt được ngay cả khi không cần LLM pre-trained, nhấn mạnh tiềm năng của các mô hình ASR Foundation có khả năng lý luận nội tại.25
* **AI Đa phương thức (Multimodal AI):** Sự phát triển không ngừng của các mô hình có khả năng xử lý đồng thời văn bản, hình ảnh, âm thanh và video là xu hướng chủ đạo.2 Các mô hình ASR Foundation thế hệ kế tiếp sẽ kết hợp khả năng hiểu ngữ cảnh từ nhiều modal, dẫn đến nhận dạng và dịch thuật ngữ nghĩa chính xác hơn.
* **Tối ưu hóa Mô hình Nhỏ (SLMs):** Mặc dù LLM lớn thống trị về mặt hiệu suất tuyệt đối, nhu cầu phát triển các mô hình AI nhỏ gọn hơn, hiệu quả hơn (Small Language Models - SLMs) vẫn tăng cao do yêu cầu về tính bền vững năng lượng (Green AI) và nhu cầu triển khai tại biên (edge deployment).26

**VI.3. Khuyến nghị Kỹ thuật Cụ thể cho Việc Phát triển Mô hình**

Để đạt được hiệu suất ASR SOTA trong môi trường sản xuất, các nhà nghiên cứu nên tập trung vào các khuyến nghị kỹ thuật sau:

* **Chiến lược Latency cho LLM:** Nếu dự án yêu cầu tích hợp LLM (đặc biệt là các mô hình trên 1 tỷ tham số), việc triển khai **Speculative Decoding (SpecASR)** là kỹ thuật bắt buộc để giảm thiểu đáng kể độ trễ giải mã tự hồi quy.19
* **Tăng cường Domain-Specific:** Đối với các ứng dụng doanh nghiệp có yêu cầu độ chính xác cao đối với thuật ngữ chuyên ngành, cần ưu tiên các phương pháp **Contextual Biasing** sử dụng WFST 24 hoặc Contextual Embeddings.23 Kỹ thuật này đảm bảo rằng mô hình có thể thích ứng động với các từ hiếm không có trong tập dữ liệu huấn luyện chung.
* **Kiến trúc Encoder Chuẩn:** **FastConformer** nên được xem xét là kiến trúc Encoder mặc định cho các tác vụ yêu cầu hiệu suất cao và streaming, nhờ sự kết hợp tối ưu giữa Transformer và Convolution, vượt trội hơn so với kiến trúc Transformer thuần túy trong việc nắm bắt ngữ cảnh cục bộ.17

**Nguồn trích dẫn**

1. Automatic Speech Recognition with BERT and CTC Transformers: A review - arXiv, truy cập vào tháng 10 30, 2025, <https://arxiv.org/html/2410.09456v1>
2. Mô hình nền tảng là gì? Tổng quan về mô hình nền tảng - VinBigdata, truy cập vào tháng 10 30, 2025, <https://vinbigdata.com/kham-pha/mo-hinh-nen-tang-la-gi-tong-quan-ve-mo-hinh-nen-tang.html>
3. ASR Models là gì - Viện FMIT, truy cập vào tháng 10 30, 2025, <https://fmit.vn/en/glossary/asr-models-la-gi>
4. arXiv:2312.17279v3 [cs.CL] 2 May 2024, truy cập vào tháng 10 30, 2025, <https://arxiv.org/pdf/2312.17279>
5. wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations - arXiv, truy cập vào tháng 10 30, 2025, <https://arxiv.org/abs/2006.11477>
6. Wav2Vec 2.0: A Framework for Self-Supervised Learning of Speech Representations, truy cập vào tháng 10 30, 2025, <https://neurosys.com/blog/wav2vec-2-0-framework>
7. An Illustrated Tour of Wav2vec 2.0 | Jonathan Bgn, truy cập vào tháng 10 30, 2025, <https://jonathanbgn.com/2021/09/30/illustrated-wav2vec-2.html>
8. HuBERT: How to Apply BERT to Speech, Visually Explained | Jonathan Bgn, truy cập vào tháng 10 30, 2025, <https://jonathanbgn.com/2021/10/30/hubert-visually-explained.html>
9. MMS - Hugging Face, truy cập vào tháng 10 30, 2025, <https://huggingface.co/docs/transformers/en/model_doc/mms>
10. [2305.13516] Scaling Speech Technology to 1,000+ Languages - arXiv, truy cập vào tháng 10 30, 2025, <https://arxiv.org/abs/2305.13516>
11. openai/whisper: Robust Speech Recognition via Large-Scale Weak Supervision - GitHub, truy cập vào tháng 10 30, 2025, <https://github.com/openai/whisper>
12. [Voice-Speech to text] Khám phá Whisper và công cụ API chuyển đổi âm thanh thành văn bản Deepgram - Viblo, truy cập vào tháng 10 30, 2025, <https://viblo.asia/p/voice-speech-to-text-kham-pha-whisper-va-cong-cu-api-chuyen-doi-am-thanh-thanh-van-ban-deepgram-AZoJjd0yVY7>
13. Whisper - Hugging Face, truy cập vào tháng 10 30, 2025, <https://huggingface.co/docs/transformers/model_doc/whisper>
14. Khám phá Whisper AI, Công nghệ nhận dạng giọng nói của AI - FPT Shop, truy cập vào tháng 10 30, 2025, <https://fptshop.com.vn/tin-tuc/danh-gia/whisper-ai-170295>
15. Which one Performs Better? Wav2Vec or Whisper? Applying both in Badini Kurdish Speech to Text (BKSTT) - arXiv, truy cập vào tháng 10 30, 2025, <https://arxiv.org/html/2508.09957v1>
16. Adaptability of ASR Models on Low-Resource Language: A Comparative Study of Whisper and Wav2Vec-BERT on Bangla - arXiv, truy cập vào tháng 10 30, 2025, <https://arxiv.org/html/2507.01931v1>
17. Conformer: Convolution-augmented Transformer for Speech ..., truy cập vào tháng 10 30, 2025, <https://arxiv.org/abs/2005.08100>
18. Speech Recognition — NVIDIA Riva - Conformer-CTC, truy cập vào tháng 10 30, 2025, <https://docs.nvidia.com/deeplearning/riva/user-guide/docs/reference/models/asr.html>
19. SpecASR: Accelerating LLM-based Automatic Speech Recognition via Speculative Decoding - arXiv, truy cập vào tháng 10 30, 2025, <https://arxiv.org/html/2507.18181v1>
20. Get 3× Faster LLM Inference with Speculative Decoding Using the Right Draft Model, truy cập vào tháng 10 30, 2025, <https://www.bentoml.com/blog/3x-faster-llm-inference-with-speculative-decoding>
21. Optimizing LLM-Based Chatbots: How to Reduce Latency & Improve Response Accuracy, truy cập vào tháng 10 30, 2025, <https://www.webcluesinfotech.com/optimizing-llm-based-chatbots/>
22. How to optimise latency for voice agents - Nikhil R, truy cập vào tháng 10 30, 2025, <https://rnikhil.com/2025/05/18/how-to-reduce-latency-voice-agents>
23. Contextualized End-to-End Speech Recognition with Contextual Phrase Prediction Network - ISCA Archive, truy cập vào tháng 10 30, 2025, <https://www.isca-archive.org/interspeech_2023/huang23d_interspeech.pdf>
24. Improved Contextual Recognition in Automatic Speech Recognition Systems by Semantic Lattice Rescoring - arXiv, truy cập vào tháng 10 30, 2025, <https://arxiv.org/html/2310.09680v4>
25. Training Speech Recognition Models to Follow Instructions - OpenReview, truy cập vào tháng 10 30, 2025, <https://openreview.net/forum?id=wYLASkYFJU>
26. 5 xu hướng Gen AI trong năm 2024 - VinBigdata, truy cập vào tháng 10 30, 2025, <https://vinbigdata.com/kham-pha/5-xu-huong-gen-ai-trong-nam-2024.html>
27. How do Multimodal Foundation Models Encode Text and Speech? An Analysis of Cross-Lingual and Cross-Modal Representations - arXiv, truy cập vào tháng 10 30, 2025, <https://arxiv.org/html/2411.17666v1>
28. Xu hướng mô hình ngôn ngữ lớn và AI tạo sinh trong 2025: Dự báo và triển vọng, truy cập vào tháng 10 30, 2025, <https://vinbigdata.com/kham-pha/xu-huong-mo-hinh-ngon-ngu-lon-va-ai-tao-sinh-trong-2025-du-bao-va-trien-vong.html>