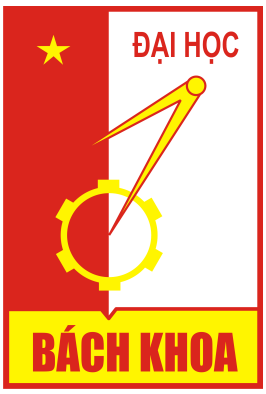
ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

—————o0o—————-



Học sâu và ứng dụng

Lyric Alignment

Giảng viên hướng dẫn: **PGS.TS. Nguyễn Thị Kim Anh**

TS. Trần Việt Trung

Sinh viên: **Hữu Tường Tú - 20194395**

Nguyễn Phúc Tân - 20194163

Hồ Anh - 20190037

Nguyễn Hải Dương - 20190044

Lời nói đầu

Lyrics Alignment là một lĩnh vực nghiên cứu trong lĩnh vực rộng hơn của quá trình xử lý âm thanh và âm nhạc. Đã có rất nhiều nghiên cứu nhằm mục đích phát triển các thuật toán và kỹ thuật để sắp xếp chính xác lời bài hát với âm thanh. Công việc ban đầu trong lĩnh vực này tập trung vào các phương thức căn chỉnh thủ công, trong đó các trình chú thích nghe âm thanh và được gán thủ công theo thời gian cho từng từ trong lời bài hát. Tuy nhiên, căn chỉnh thủ công là tốn thời gian và dễ bị lỗi, và kết quả là, đã có một sự thay đổi theo các phương pháp căn chỉnh lời bài hát tự động trong những năm gần đây. Các kỹ thuật liên kết lời bài hát tự động có thể được phân loại thành hai loại: dựa trên mẫu và dựa trên âm thanh. Các phương pháp dựa trên mẫu sử dụng một mẫu hoặc mẫu được xác định trước để căn chỉnh lời bài hát với âm thanh. Các phương pháp này hoạt động tốt cho một số loại bài hát nhất định, nhưng có thể bị giới hạn bởi sự phức tạp của mẫu và sự biến đổi của âm nhạc. Mặt khác, các phương pháp dựa trên âm thanh, phân tích các tín hiệu âm thanh trực tiếp để xác định thời gian của từng từ trong lời bài hát. Các phương pháp này có thể được chia thành hai loại: dựa trên nhịp và dựa trên âm tiết. Các phương pháp dựa trên beat phù hợp với lời bài hát với nhịp điệu trong âm nhạc, trong khi các phương pháp dựa trên âm tiết phù hợp với lời bài hát với các âm tiết trong giọng hát.
Căn chỉnh lời bài hát có thể được sử dụng trong nhiều ứng dụng khác nhau, bao gồm hệ thống karaoke, phần mềm giáo dục âm nhạc và trò chơi video âm nhạc. Nó cũng có thể được sử dụng để tạo chú thích đóng cho các video âm nhạc hoặc biểu diễn trực tiếp, làm cho nội dung rõ ràng dễ tiếp cận hơn.

Link source code:
`https://github.com/huutuongtu/Deep_Learning_Project`

Mục lục

Chương 1

Giới thiệu chung

1.1 Bài toán

Nhiều người trong chúng ta thích hát theo những bài hát theo cách của các ca sĩ yêu thích trong album, (phong cách karaoke). Để thực hiện nó, chúng tôi có thể cần phải loại bỏ giọng hát của (các) ca sĩ khỏi các bài hát, sau đó cung cấp lời bài hát phù hợp kịp thời với âm thanh đệm. Có nhiều công cụ khác nhau để loại bỏ giọng hát, nhưng rất khó để căn chỉnh lời bài hát với bài hát.
Để giải quyết vấn đề này, chúng ta sẽ xây dựng một mô hình để điều chỉnh lời bài hát với âm thanh âm nhạc

1.2 Yêu cầu

- **Đầu vào**: Một phân đoạn âm nhạc (bao gồm cả giọng hát) và lời bài hát của nó.
- **Đầu ra**: Thời gian bắt đầu và thời gian kết thúc của mỗi từ trong lời bài hát

Chương 2

Hướng nghiên cứu liên quan

- wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations

  Wav2vec 2.0 là một mô hình ASR SOTA có khả năng chuyển từ audio đầu vào thành text nhận dạng. Một model wav2vec2.0 có thể cho ra một mô hình xác suất theo timestamp và label. Điều này làm cho nó trở thành một lựa chọn phù hợp cho tác vụ lyrics alignment, vì xác suất từ mô hình wav2vec2 sẽ giúp căn chỉnh chúng với lời bài hát tương ứng lyrics ở dạng văn bản. Ngoài ra, Wav2vec 2.0 có khả năng xử lý dữ liệu giọng nói ồn ào, nhiều nhiễu và đa dạng, những dữ liệu thường có trong các audio âm nhạc và có thể học cách phiên âm lời bài hát ngay cả khi có nhạc nền và phong cách hát.

- Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks

  Connectionist Temporal Classification (CTC) đã được giới thiệu để huấn luyện mạng RNN để dán nhãn trực tiếp các chuỗi không phân đoạn. CTC là hàm loss đã được sử dụng trong nhận dạng giọng nói end-to-end với mô hình từ audio thành text.

- CTC-Segmentation of Large Corpora for German End-to-end Speech Recognition

  Forced Alignment là một kỹ thuật để lấy bản sao chính tả của một tệp âm thanh và tạo ra một phiên bản căn chỉnh theo thời gian. Ở bài toán này, sau khi chúng ta có mô hình xác suất các label trên timestamp, ta sẽ cho qua CTC forced alignment để khớp thời gian với lyrics.

- Attention is all you need

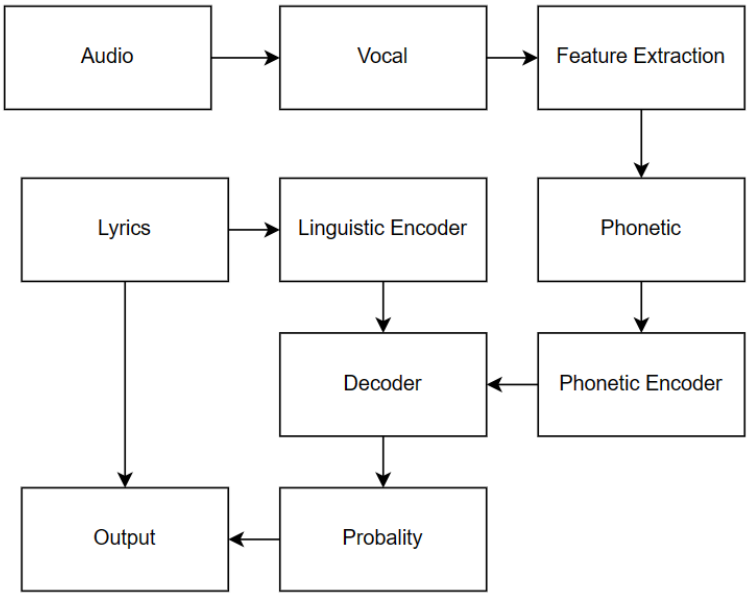
Attention là một cơ chế giúp mô hình có thể tập trung vào các phần quan trọng trên dữ liệu, ở đây với bài toán của chúng ta, lyrics được cung cấp, vậy ta có thể thêm lyrics attention vào model acoustic có thể giúp mô hình tăng khả năng nhận dạng xác suất hơn.

Chương 3

Đề xuất hướng tiếp cận

3.1 Kiến trúc tổng quan

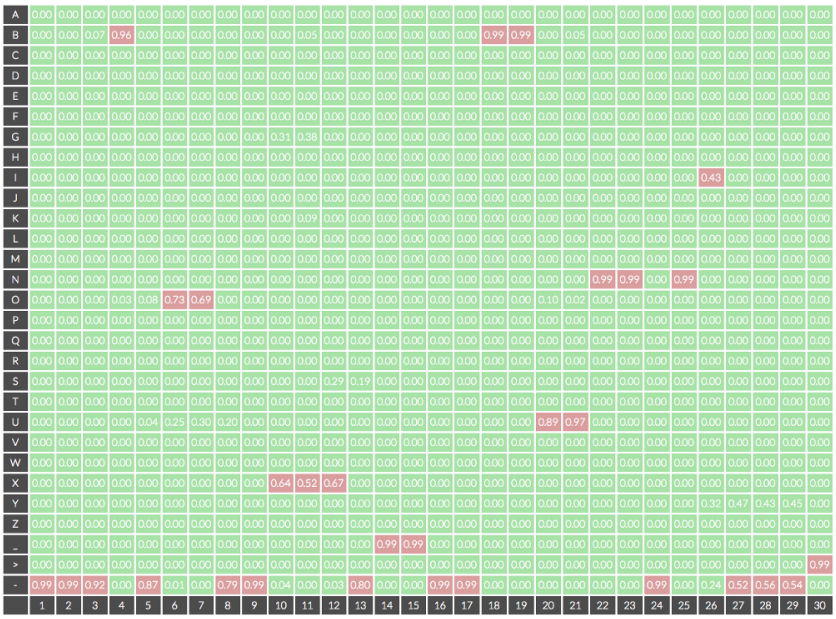
Kiến trúc tổng quan model



Đầu tiên, ta sẽ gỡ âm thanh nhạc ra khỏi audio, sau đó ta được giọng hát chay của ca sĩ. Tiếp theo, ta sử dụng mô hình pretrained wav2vec2.0 để trích xuất đặc trưng của mô hình, ta sẽ có một đặc trưng gọi là Phonetic.
Phonetic sẽ được đi qua một block Phonetic Encoder được Query và Lyrics sẽ được đi qua block Linguistic Encoder thu được Key và Value, ta sử dụng attention để có một cơ chế chú ý lyrics vào mô hình thu được vector ngữ

cảnh. Sau đó ta concat vector ngữ cảnh với Query rồi cho qua một hàm Linear thu được một mô hình xác suất của label theo thời gian.
Cuối cùng, ta cho mô hình xác suất qua Forced Alignment để căn chỉnh thời gian với lyrics.

Output probability



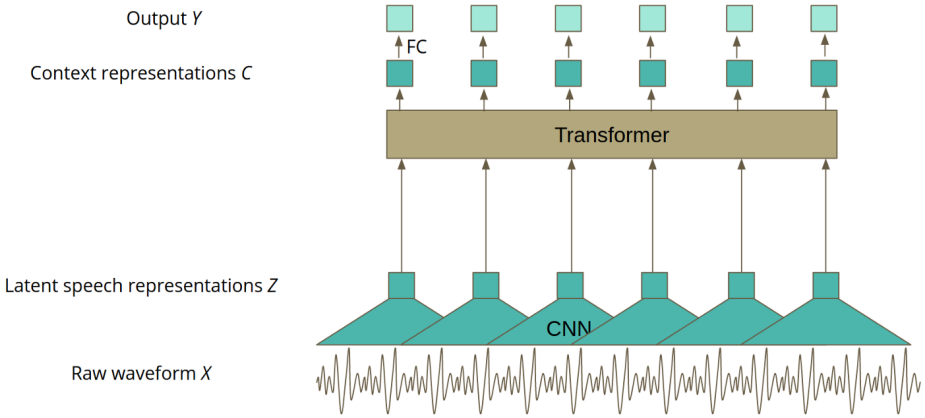
3.2 Feature Extraction

Input được chia làm các frame có độ dài 0.02s. Sau đó được đưa qua pre-trained model Wav2vec2.0 đã được đóng băng trích xuất được đặc trưng.

Input: Được chia thành các time-stamp. Kích thước: [time/0.02] x 1

Output Các feature vectors kích thước [time/0.02] x 768:

Kiến trúc của Wav2vec2.0



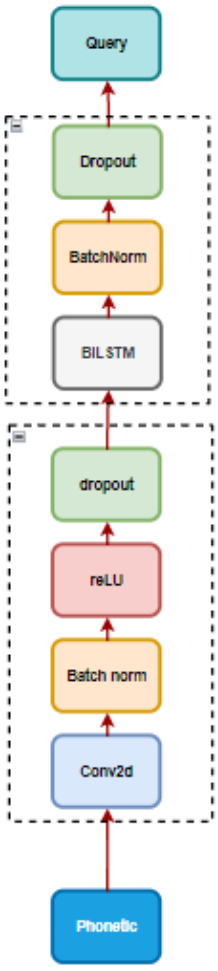
Raw waveform sẽ được đưa qua một mạng CNN để thu được Latent speech representation Z. Sau đó được đi qua N block Transformer thu được context representations C. Cuối cùng được đưa qua Linear để thu được một mô hình xác suất

3.3 Encoder

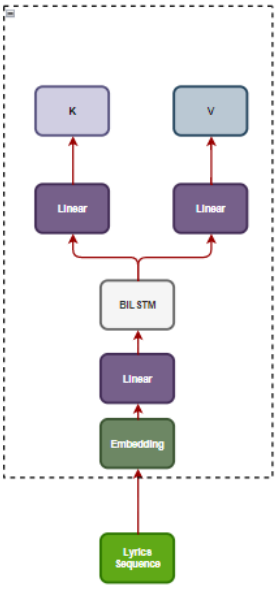
Phonetic đầu tiên sẽ đưa qua một mạng CNN tiếp theo đi qua một mạng RNN có cấu trúc như hình 3.1 từ đó ra được output Query. Lyrics sẽ được embedding 1 hot sau đó được đưa qua một Linear rồi đưa qua BiLSTM, cuối cùng được cho qua 2 Linear theo 2 nhánh được output Key và Value.

3.4 Decoder

Query, Key và Value từ output đầu ra của Encoder sẽ được Attention vào với nhau thu được vector ngữ cảnh C. Sau đó ta concat C với Query rồi đưa qua Linear, cuối cùng sẽ được tính bằng softmax để đưa ra được một mô hình xác suất.

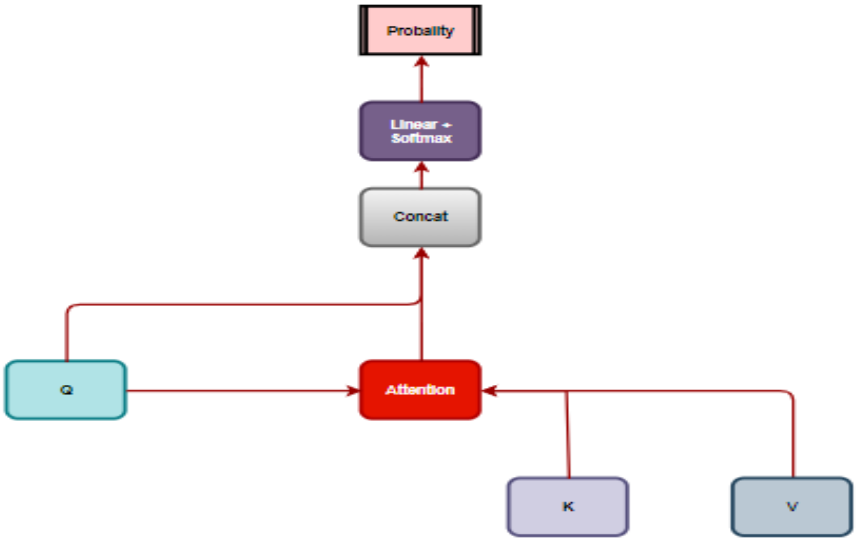


Hình 3.1: Phonetic Encoder



Hình 3.2: Linguistic Encoder

,,



3.5 CTC Loss

Hàm loss chúng ta sử dụng ở đây là CTC Loss:

- Thực thi trên từng time-step của ma trận probability đầu ra.
- S là tập huấn luyện bao gồm các cặp dữ liệu (x, y) - (x là dữ liệu đầu vào, y là groundtruth).
- B là hàm loại bỏ các kí tự trùng nhau và ký tự trống. Ví dụ:

$$B(\text{"}-t-oo\text{"}) = \text{"}to\text{"}; B(-ttt-oooo-o-) = \text{"}too\text{"}$$

- π là path (mỗi path là một cách chọn ký tự ở một timestamp rồi kết hợp chúng lại với nhau)
- Alignment của một nhãn L là tập các path π với độ dài time-step thỏa mãn $B(\pi) = L$
- Xác suất của nhãn y đối với dữ liệu đầu vào x:

$$p(y \mid x) = \Sigma_{\pi \in B^{-1}(y)} p(\pi \mid x)$$

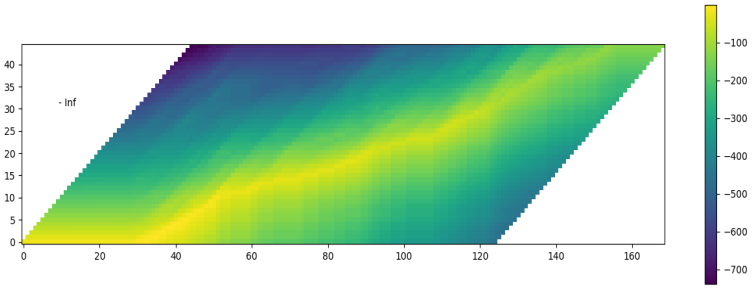
- Loss function:

$$Loss(S) = -\sum_{(x,y) \in S} \ln(p(y \mid x))$$

3.6 CTC Forced Alignment

Sau khi đã có một mô hình xác suất, ta sử dụng một thuật toán quy hoạch động để căn chỉnh thời gian với từ.

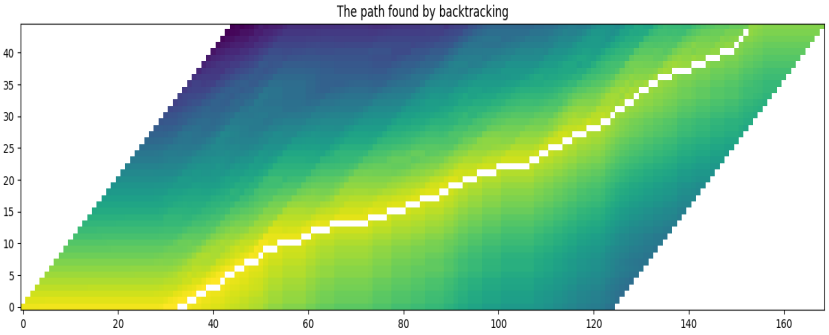
$$k_{t,j} = \begin{cases} \max(k_{t-1,j} \cdot p(blank|t),\ k_{t-1,j-1} \cdot p(c_j|t)) & \text{if } t > 0 \wedge j > 0 \\ 0 & \text{if } t = 0 \wedge j > 0 \\ 1 & \text{if } j = 0 \end{cases}$$

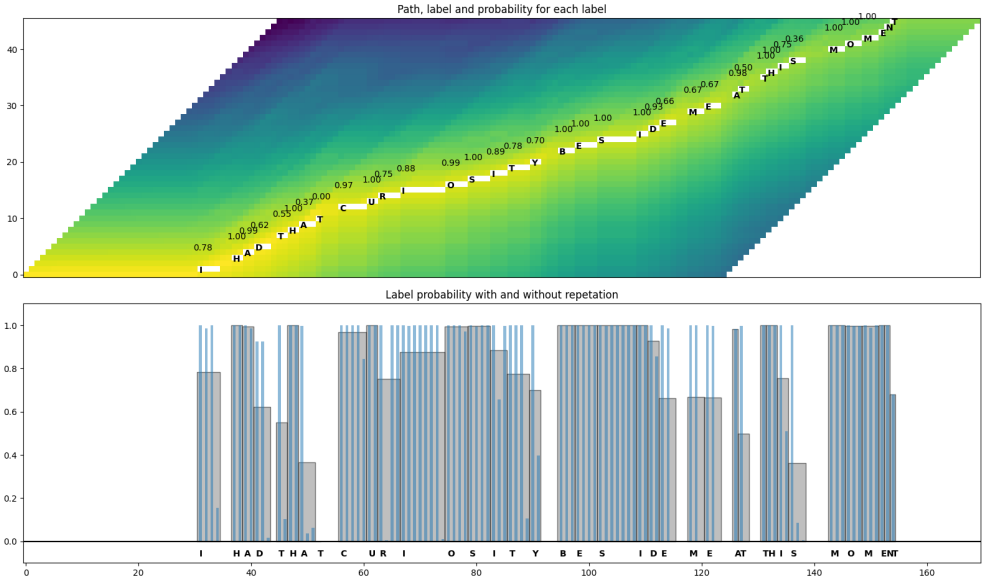


Ta định nghĩa một ma trận k là ma trận trellis như công thức trên. Sau khi có ma trận trellis, ta sử dụng backingtrack duyệt từ frame cuối lên, bắt đầu từ frame mà xác suất của từ cuối cùng của lyrics là cao nhất.
Nhãn khớp thời gian sẽ được áp dụng theo công thức:

$$a_t = \begin{cases} a_{t+1} & k_{t,a_{t+1}} p(blank|t+1) > k_{t,a_{t+1}-1} p(c_{a_{t+1}-1}|t+1) \\ a_{t+1} - 1 & else \end{cases}$$

Nghĩa là nếu thỏa mãn điều kiện thứ nhất thì frame trước của frame đang duyệt có nhãn giữ nguyên, nếu không thì nhãn của frame trước frame hiện tại sẽ dịch 1 vị trí về bên trái của lyrics


The path found by backtracking


Path, label and probability for each label
Label probability with and without repetation
I HAD THAT CURIOSITY BESIDE ME AT THIS MOMENT

Chương 4

Thực nghiệm

4.1 Dữ liệu

Dữ liệu training:

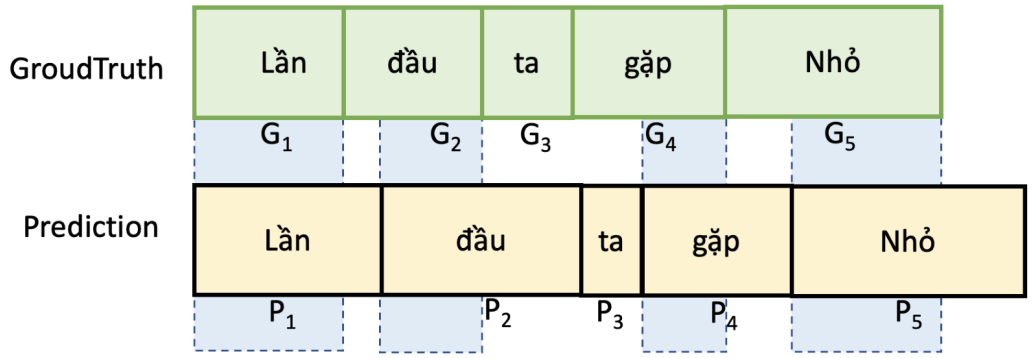
- 1057 phân đoạn âm nhạc từ 480 bài
- Mỗi phân đoạn được cung cấp một âm thanh được định dạng dưới dạng tệp WAV và tệp JSON bao gồm lời bài và khung thời gian được căn chỉnh từng từ như ví dụ trên.

Dữ liệu testing

- 264 đoạn nhạc từ 120 bài hát mà không mà không tệp căn chỉnh lời bài hát

4.2 Evaluation Metrics

- Để đánh giá kết quả của dự đoán bằng cách sử dụng Intersection over Union (IoU).
- Với IoU metric, chỉ số càng cao thì càng tốt.



Hình 4.1: Ví dụ của IoU cho prediction and ground truth

IoU của prediction và ground truth của audio segment s_i được tính bởi công thức sau:

$$IoU(s_i) = \frac{1}{m}\sum_{j=}^{m}\frac{G_j \bigcap P_j}{G_j \bigcup P_j}.100\% \tag{4.1}$$

trong đó: m là số lượng các token của s_i
IoU trên tất cả n phân đoạn âm thanh là trung bình của IoUs tương ứng của chúng.

$$Final\_IoU = \frac{1}{n}\sum_{i=1}^{n} IoU(s_i) \tag{4.2}$$

4.3 Các mô hình thực nghiệm

6 mô hình được implement và thực nghiệm để so sánh kết quả

- Wav2vec2.0 finetune layer linear cuối cùng
- Kết hợp wav2vec2.0 + vocal và finetune layer cuối cùng
- Sử dụng đặc trưng Phonetic trích xuất từ wav2vec2.0 và cho qua Phonetic Encoder
- Sử dụng đặc trưng Phonetic trích xuất từ wav2vec2.0 + Linguistic qua Phonetic Encoder + Linguistic Encoder
- Sử dụng đặc trưng Phonetic trích xuất từ wav2vec2.0 + Vocal qua Phonetic Encoder
- Sử dụng đặc trưng Phonetic trích xuất từ wav2vec2.0 + Vocal + Linguistic qua Phonetic Encoder + Linguistic Encoder

4.4 Kết quả thực nghiệm

Model	Result
Baseline (Wav2vec2.0)	45.87
Wav2vec2.0 + Vocal	46.21
Phonetic	50.22
Phonetic + Linguistic	50.32
Phonetic + Vocal	49.98
Phonetic + Linguistic + Vocal	50.26

Bảng kết quả thực nghiệm cho thấy khi ta áp dụng mô hình end-to-end trên đặc trưng phonetic attention với lyrics là tốt nhất với độ chính xác 50.32 phần trăm. Cải thiện 4.45 phần trăm so với mô hình pretrained Wav2vec2.0 finetune last layer Linear.

Chương 5

Kết luận

5.1 Nhận xét chung

Khi ta sử dụng mô hình đề xuất lên phonetic cải thiện hơn khoảng 4 phần trăm so với mô hình gốc.
Tuy nhiên khi thêm lyrics dường như kết quả mô hình không cải thiện. Có thể một do mô hình sau khi đưa ra căn chỉnh thời gian được hậu xử lý kéo dài thêm để nối các từ với nhau. Việc kéo dài thời gian có thể đã bao phủ được output của 4 model dưới.

5.2 Hướng cải tiến

Sử dụng các pretrained ASR khác để extract feature phonetic, sau đó ta có thể fusion các mô hình xác suất lại với nhau để tránh overfit