Learning Audio-Lyric Alignment via Emotional Sentiment

Aimon Benfield-Chand
447
aimonbc@uw.edu

Javon Hickmon
517
javonh@uw.edu

Donovan Clay
447
donoclay@uw.edu

Project Information:

Project Type	Open-ended Project
Project Title	Learning Audio-Lyric Alignment via Emotional Sentiment

You don't need to fill out this part for the midway report, but you should fill this out for your final report.

Specify the individual contributions.

- **Aimon Benfield-Chand:** Planning, data preprocessing, evaluation, poster, and report.
- **Javon Hickmon:** Planning, fine-tuning model, poster, and report.
- **Donovan Clay:** Planning, data preprocessing, poster, and report.

Abstract

In this research, we investigate the feasibility of using emotional sentiment as means of aligning an audio-lyric joint embedding model. Our approach builds on trends in music information retrieval (MIR) by fine-tuning CLAP, a multimodal model with dual tower audio and text encoders, on "audio + lyric" pairs and "audio + sentiment-description" pairs, comparing the two approaches to the pre-trained CLAP model on cross-modal audio-to-lyric retrieval. For fine-tuning, we use the DALI dataset, which comprises over 5000 songs with synchronized audio, lyrics, and notes. The potential applications of the resulting model are broad, including emotion-based song search and recommendation, cross-modal lyric-audio retrieval, and emotion-conditioned music generation.

1 Introduction

1.1 Emotional Sentiment

Music and emotion are two concepts that are frequently coupled. Musicians frequently create from a place of emotion, and aim to present their art in a way that allows the audience to feel the same emotion. Leo Tolstoy perfectly encapsulated this concept when he wrote, "Music is the shorthand of emotion." Songs whose music and lyrics match in emotion tend to provoke stronger and more unified emotional reactions in listeners, whereas songs whose music and lyrics seem incongruous may evoke weaker or more meta responses (9). As a result of the interconnected nature of music and emotion, we aim to understand if this relationship is consistent and can be computationally modelled. We hypothesise that creating a model to learn the joint embedding space between lyrics and audio based on sentiment, will result in embeddings that show the tightly coupled nature of music and emotion, and will assist in downstream tasks.

1.2 Motivation

Our model not only poses to offer insight into the musicology of pop song composition, but also has applications to many downstream retrieval and generation applications in MIR, MER, and music generation. With the ability to predict emotional sentiment from pure audio, our model could enable a sentiment-to-song search engine that can retrieve or recommend songs based on a desired sentiment or mood. Similarly, our model might enhance music generation, allowing composers to condition the automatic generation of lyrics on the extracted sentiment of pre-composed song audio. Such an application might involve generating a set of lyric contenders and then using our model to select that which best aligns to the song's audio.

2 Related Works

2.1 MER

Music Information Retrieval (MIR) is a growing field of research that has seen great progress in recent years through the application of deep learning techniques to the domain of music. The task of recognizing the emotional sentiment of a song, coined Music Emotion Recognition (MER), has seen a particular increase in interest due to the emergence of deep neural networks. Early studies by Parisi and Delbouys, for instance, compared the performance of several neural architectures (LSTM, GRU, ConvNet, BERT, ELMo) on MER when given full audio as opposed to vocal-only audio (2; 9). Others incorporated even more input data sources into a fusion approach by training Transformer encoders separately on auditory and lyrical input for emotion prediction. For instance, (7) outperformed previous multi-modal emotion classification methods using an LSTM audio encoder and a BERT lyric encoder, while (15) achieved further strides in MER by employing multi-head attention

Transformers on lyric, audio, and visual motion-capture input. Given the apparent promise of fusion audio-language models trained on MER, we explore the infusion of emotion into more general fusion models as a means of aligning audio and lyrics.

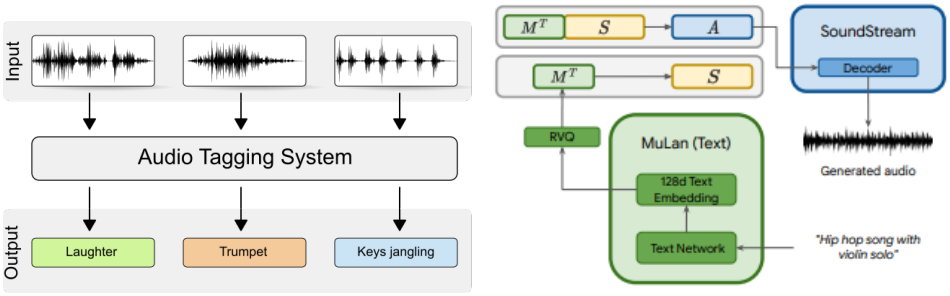


Figure 1: Diagrams of an example audio-tagging pipeline (left) and MusicLM's inference-time pipeline (right).

2.2 Audio-Lyric Alignment

Following in the example of CLIP (10), several recent studies, such as MuLan (6), Wav2CLIP (13), and CLAP (4), have proposed variations of CLIP's contrastive pre-training regime for learning joint text-audio representations. They do this by training dual audio and text embedding towers on audio-text pairs, containing spectrogram audio representations and text annotations of the audio. This approach has seen recent adoption in both MIR and generative pipelines. For instance, MuLan was integrated into Meta's MusicLM pipeline for its ability to convert text descriptions of a desired song generation into audio embeddings to be used as input condition for its audio decoder (1). Within MIR, these models have also found success for audio tagging, which involves automatically generating text annotations/descriptions of audio 1. Audio-tagging is particularly usefulf or identifying the musical sources present in a given audio file or classifying its musical genre. We posit, however, that these joint audio-text embeddings are less useful for tasks that require a more abstract understanding of audio's musical features and the experience they elicit in a human listener. Text descriptions of audio, in particular, lack the musical information present in song lyrics, which are essential in communicating the emotion of a song. To this end, we modify the existing CLAP pipeline to learn a correspondence from song audio to song lyrics, rather than to audio descriptions. This approach is more specialized to the medium of popular songs, which generally include continuous audio and sung/rapped lyrics, but less versatile for general audio samples that do not contain lyrics.

2.3 Model Specification

Since our pipeline fine-tunes the pre-trained CLAP model, we provide a detailed overview of its architecture below.

2.3.1 CLAP

The CLAP (Contrastive Learning of Audio and Text Embeddings for Speaker Discrimination) model is a deep learning architecture designed for learning joint audio and text representations in a contrastive learning framework. Its primary objective is speaker discrimination, which involves distinguishing between different speakers based on their audio recordings. Since this objective is different from our intended downstream task, we make a few key modifications to our training process.

Audio Encoder: CLAP uses the pretrained Audio Spectrogram Transformer (AST) proposed by (**?**). The ViT model processes input spectrograms as 16x16 patches and projects them to 1D patch embedding vectors of size 768 as output. As is common, AST prepends a '[CLS]'

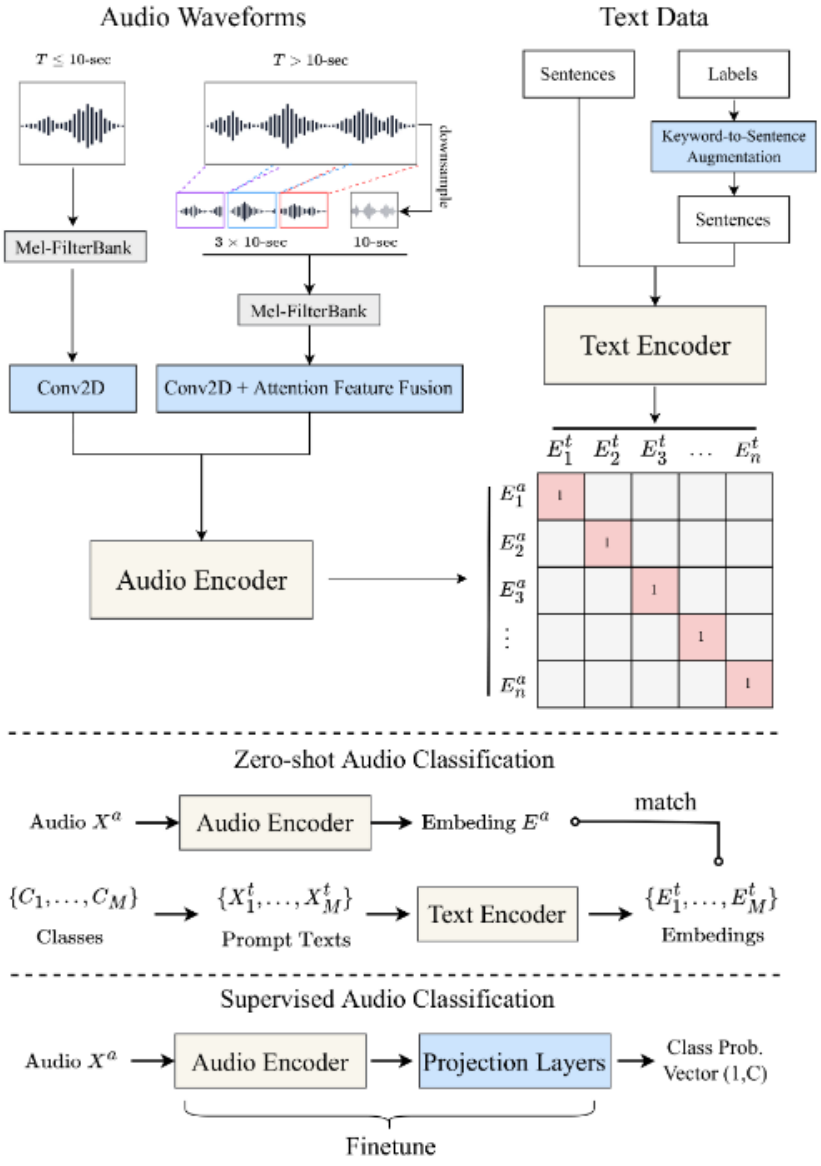


Figure 2: CLAP learning pipeline.

token to the input sequence. We use the final learned embedding of this classification token as our audio embedding.

Text Encoder The CLAP model uses a RoBERTa model to get text features. Both the text and audio features are then projected to a latent space with identical dimension. The dot product between the projected audio and text features is then used as a similar score.

3 Method

3.1 Models

As few studies have attempted to generate joint embeddings of song audio and lyrics, we develop and compare several variations of the pre-trained CLAP model on the task of audio-to-lyric retrieval (discussed in Section 6).

These model variations are:

- **CLAP-emo**: CLAP fine-tuned with emotional sentiment
- **CLAP-lyr**: CLAP fine-tuned with lyrics (no emotional sentiment)
- **CLAP-pre**: CLAP pre-trained (no fine-tuning)

3.1.1 CLAP Fine-tuned with Sentiment

For CLAP-emo, we align song audio to song lyrics using emotional sentiment as an alignment proxy. To achieve this, we first generate sentence descriptions of the emotional sentiment of each section of lyrics, which are then fed into the model along with the corresponding spectrogram audio representations. We perform contrastive fine-tuning on these audio + sentiment-description pairs to produce the final CLAP-emo model. We expect CLAP-emo to achieve superior performance on MER tasks due to its preliminary leveraging of an LLM for sentiment analysis and its explicit inclusion of emotional sentiment in the alignment process.

1. Generate a text description of the emotional sentiment of each song segment's lyric.
2. Embed these text descriptions using a frozen pre-trained sentence-embedding model to obtain lyric-sentiment embeddings.
3. Embed the audio spectrogram of each song segments using a ViT audio encoder.
4. Fine-tune the audio encoder according to the contrastive loss between the audio embeddings and the lyric-sentiment embeddings.

The reason we generate sentence descriptions, rather than mere classifications, of the emotional sentiment of each lyric in CLAP-emo is to enable an open-vocabulary. An open-vocabulary of lyrical sentiment theoretically enables our audio encoder to learn more precise embeddings, which also allow for more versatility and expressiveness in downstream cross-modal retrieval tasks. For instance, our open-vocab V1 model might be able to take in a text description of a desired emotional sentiment and retrieve the audio that best reflects that sentiment.

3.1.2 CLAP Fine-tuned with Lyrics

To isolate the effectiveness of emotional sentiment as an alignment proxy, we perform an ablation. To this end, we train CLAP-lyr by aligning audio to song lyrics directly, omitting any explicit representation of emotional sentiment. Similar to CLAP-emo, we performing contrastive fine-tuning, but now use these audio + lyric pairs, instead of audio + sentiment-description pairs. Compared to the CLAP-emo model, we expect this ablated-model to learn more precise representations of song lyrics themselves, but perhaps less robust and abstract representations of musical emotion, which might be detrimental on MER tasks. Whether

lyrical precision or emotional robustness produces preferable joint audio-lyric embeddings for cross-modal retrieval is the major inquiry of our research.

1. Embed each song segment's lyric using a frozen pre-trained sentence-embedding model to obtain lyric embeddings.
2. Embed the audio spectrogram of each song segments using a ViT audio encoder.
3. Fine-tune the audio encoder according to the contrastive loss between the audio embeddings and the lyric embeddings.

3.1.3 CLAP Pre-trained

For CLAP-pre, we simply evaluate the pre-trained CLAP model without performing any additional fine-tuning. We expect this baseline to achieve weaker performance on MER and cross-modal retrieval tasks, since it has not been pre-trained specifically on audio + lyric pairs, but on audio + text-description pairs.

3.2 Contrastive Fine-Tuning

3.2.1 Positive and Negative Pairs

Unlike traditional contrastive learning regimes for cross-modal alignment, we only unfreeze and fine-tune our pretrained audio encoder model, not our lyric sentiment encoder, in order to align the audio embeddings to the frozen lyric sentiment embeddings. Our method presumes that lyrical sentiment is a viable proxy for the lyric-audio alignment of songs, making it desirable to preserve the original lyrical sentiment embedding as a gold label.

In our method, positive pairs consist of the embeddings of audio spectrograms and lyrics from the same song segment, while negative pairs consist of the audio-lyric embeddings from different song segments.

Time permitting, we also plan to experiment with labeling a small number of audio-lyric embeddings from different song segment's as positive pairs according to their cosine similarity.

3.2.2 Contrastive Loss

We calculate contrastive loss according to the following formula

$$L(Y, D) = (1 - Y) \cdot \frac{1}{2}(D)^2 + Y \cdot \frac{1}{2} \max(0, m - D)^2$$

where

- $L(Y, D)$ represents the contrastive loss
- Y is a binary label (0 for positive pairs, 1 for negative pairs)
- D is the distance between the embeddings, calculated as cosine similarity
- m is the margin of tolerance for the distance between negative pairs

4 Data

4.1 CLAP

CLAP is trained on the LAION's LAION-Audio-630K, which contains 633,526 audio-text pairs. Due to copyright reasons, however, LAION has not released the exact contents of the pre-training dataset.

4.2 DALI

We train and evaluate our models on separate splits of DALI, a large dataset of synchronised audio, lyrics, and notes, containing 5358 songs of real music, 4018 of which were English. Since we're doing a proof of concept and we know Bert performs better on English, we filtered out non-English songs. For each song, the dataset provides a YouTube link to the audio/video, time-aligned lyrics, and the time-aligned notes of the vocal melody. Lyrics are synchronized hierarchically at four levels of granularity: notes, words, lines and paragraphs. Metadata concerning song genre, language, and musician are also included.

4.3 Pre-processing

To preprocess the data, we follow the following steps.

1. Group lines of lyrics into "lyric segments", each corresponding to $\sim$ 10 seconds of audio
2. Use Chat-GPT-3.5 to generate a sentence text description of each lyric segment's sentiment.
3. Download audio MP4 files from YouTube
4. Convert audio to WAV format
5. Generate log MEL-spectrograms for each segment.

5 Code

Our code is available at https://github.com/aimonbc24/Audio-Lyric-Alignment-via-Emotional-Sentiment.

6 Experiments

Audio-to-Lyric Retrieval:

Objective: Evaluate and compare the CLAP-emo, CLAP-lyr, and CLAP-pre models on a lyric retrieval task.

Methodology:

- *Task Description:* The task involves classifying an audio query by matching it with the correct lyrical segment from an assortment of 10 lyrics, testing the models' ability to align audio embeddings with lyric embeddings.
- *Data:* Utilize subsets of the DALI test split, where each audio segment is paired with multiple lyrical segments, including one correct match and several distractors.
- *Model Preparation:* Employ the model.
- *Evaluation Process:*
  - Given an audio-lyric pair $a^{(i)}, l^{(i)}$, embed the audio and lyric segments using the models to produce an audio query embedding $\mathbf{q}^{(i)}_{\text{audio}}$ and a lyric query embedding $\mathbf{q}^{(i)}_{\text{lyric}}$.
  - Embed the lyrics $\{l^{(1)}, l^{(2)}, ..., l^{(n)}\}$ according to the model embedding strategies to produce lyric key embeddings $\{\mathbf{k}^{(1)}, \mathbf{k}^{(2)}, ..., \mathbf{k}^{(n)}\}$.
  - Compute the cosine similarity between the audio query embedding $\mathbf{q}^{(i)}_{\text{audio}}$ and each lyric key embedding $\mathbf{k}^{(j)}$. Take the softmax of these similarity logits to produce an audio-lyric probability distribution $P_\theta(\mathbf{k}|\mathbf{q}_{\text{audio}})$.

- Compute the cosine similarity between the lyric query embedding $\mathbf{q}_{\text{lyric}}^{(i)}$ and every lyric embedding $\mathbf{k}^{(j)}$. Take the softmax of these similarity logits to produce a lyric-lyric probability distribution $P_\theta(\mathbf{k}|\mathbf{q}_{\text{lyric}})$.
- Compute the KL divergence $D_{KL}(P_\theta(\mathbf{k}|\mathbf{q}_{\text{audio}}) \parallel P_\theta(\mathbf{k}|\mathbf{q}_{\text{lyric}}))$

- *Evaluation Metrics:*
  - Kullback-Leibler (KL) Divergence: used to quantify the discrepancy between the he audio-lyric and lyric-lyric probability distributions, treating the audio-lyric probability distribution as the prediction and the lyric-lyric probability distribution as the ground-truth.
  - Top-k Classification Accuracy: measures the accuracy of the correct class being in the top-k predicted most probable classes. For a larger k, the accuracy should be higher, because the evaluation is more forgiving to the model.

Expected Outcome: The aim is to demonstrate the models' effectiveness in cross-modal retrieval, highlighting their capability to correlate information across audio and text modalities.

7 Results

7.1 Fine-Tuning Results

During fine-tuning, we freeze the text encoder and focus on learning the weights for the audio encoder. We do this primarily for data efficiency and Using our custom dataset, each input is sent to the audio encoder is of the shape. During training, we use the Adam optimizer with $\beta 1 = 0.99$, $\beta 2 = 0.9$ and cosine learning rate decay at a basic learning rate of 104. We train the model using a batch size of 128 on two NVIDIA RTX-6000 gpus with our custom dataset. This dataset has 32228 samples on training dataset, 4029 samples on the validation set, and 4028 samples on the test set. We train the model for 5 epochs.

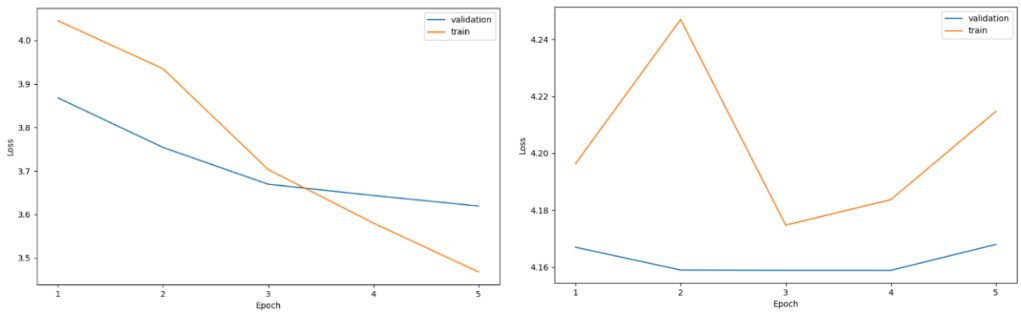


Figure 3: Loss plot after training for 5 epochs with CLAP-pre (left) and CLAP-emo (right).

We also create t-SNE plots for each of our models to further understand the embedding spaces that were formed.

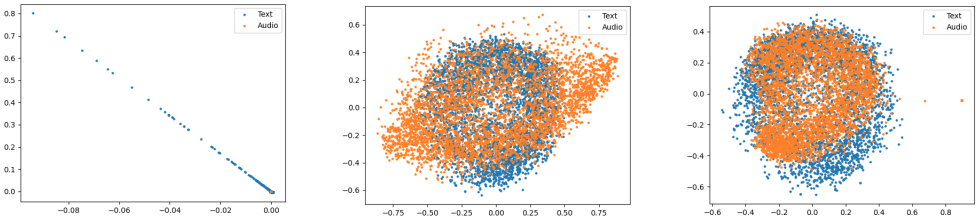


Figure 4: t-SNE plots for CLAP-emo embeddings (left), CLAP-lyr embeddings (middle), CLAP-pre embeddings (right).

7.2 Retrieval Experiment Results

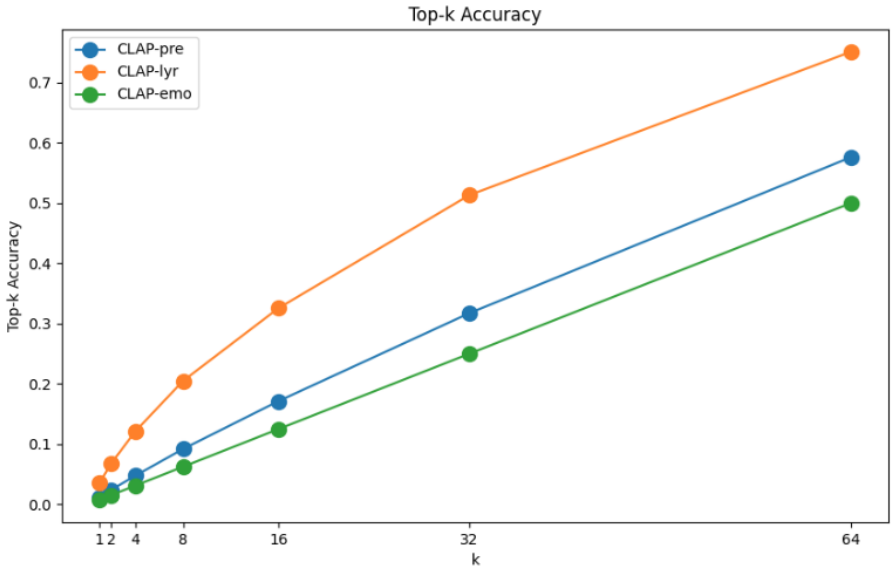


Table 1: Experimental Results Summary

Metric	KL-Div	Top-1%	Top-2%	Top-4%	Top-8%
CLAP-pre	3.400	1.18 %	2.44 %	4.76 %	9.20 %
CLAP-emo	0.087	0.78 %	1.56 %	3.13 %	6.25 %
CLAP-lyr	3.524	3.55 %	6.85 %	12.12 %	20.49 %

8 Discussion

The initial results were surprisingly negative. As is viewable from our top-k accuracy plot. Our model achieves worse performance on average when compared to CLAP-pre. We hypothesize that this is the result of a number of constraints in our training and evaluation processes.

Primarily, the lack of time and compute to train our model, process our data, and evalute the results was a major hinderance in achieving adequate performance. Despite this, the results are extremely telling when we look at the t-SNE plots for each of the models. The

CLAP-lyr embeddings performed the best overall, as is shown by the highly interspersed embedding space. Even the out-of-the box CLAP-pre has a fairly dispersed embedding space when compared to CLAP-emo.

One potential source of this bad embedding space could have been our custom training data. Each of our sentiment descriptions began with the same phrase, so the model likely learned how to predict based on that sequence. In the future we aim to look more closely at how we form our data, along with our training and architecture choices.

9 Conclusion

In this study, we investigated the use of emotional sentiment as a mechanism for aligning audio and lyric embeddings by fine-tuning the CLAP model, a multimodal approach with dual tower audio and text encoders. Our methodology involved the comparison of fine-tuning CLAP with "audio + lyric" and "audio + sentiment-description" pairs against its pre-trained state, focusing on the task of cross-modal audio-to-lyric retrieval. This was facilitated by employing the DALI dataset, which consists of over 5000 songs with synchronized audio, lyrics, and notes, to examine the potential for emotion-based song search, recommendation, and emotion-conditioned music generation.

The results of our research indicate that while the integration of emotional sentiment into the alignment process offers a promising direction for enhancing the effectiveness of multimodal models in music information retrieval (MIR) tasks, the actual performance improvements and applicability require careful consideration of the model's limitations and the complexities inherent in emotional sentiment analysis. From the results of our cross-modal audio-to-lyric task, it appears that the CLAP-lyric model outperformed both the CLAP-emotion and CLAP-pretrained models. This is not particularly suprising, however, given that the CLAP-lyric model embedded the lyrics explicitly, whereas the CLAP-emotion model embedded only the emotional abstraction of the lyrics. As a result, the CLAP-emotion model embeddings were likely underfit to for the purpose of retrieving exact lyrics themselves. Nonetheless, we expect that learning the emotional abstraction of the lyrics would be beneficial for a task like MER.
To further test this behavior, we aim to evaluate all of the models on our custom sentiment dataset alongside the pure lyrics. It is important to note that even if our model had functioned better than the base model, the intention is not to use CLAP-emotion for classification tasks. This model is strictly useful when you want to match text to emotion.

10 Future Work

Due to time constraints, we were only able to evaluate our model on the one downstream task of cross-modal audio-to-lyric retrieval. While this task serves as a fine sanity check of how useful our models' embeddings are for music information retrieval, there are many other downstream tasks we would want to evaluate our models on in the future.

MER:
To verify the that our CLAP-emo model learned a useful abstraction of emotional sentiment, we would first like to fine-tune and evaluate our foundation models on music emotion retrieval. This would involve learning a linear classification head that uses the foundation models' audio encoders to embed and then classify audio queries across a fixed vocabulary of emotional sentiments.

Emotion-based Song Recommendation:
This is likely the simplest application of the CLAP-emo embeddings, and be operationalized by embedding emotion descriptions and querying a vector database for the k-nearest audio neighbors to be returned as song recommendations.

Music Generation:
Another promising application of audio-language models involves conditioning music generation on prior musical information. Given their joint audio-lyric embedding space,

our models have natural application to audio-conditioned lyric generation and lyric-conditioned audio generation. Each of these tasks are extend and depend on cross-modal retrieval; Whereas audio-conditioned lyric generation is the generative correlate to our evaluation task of audio-to-lyric retrieval, lyric-conditioned audio generation reverses this pipeline and could be achieved within the Meta's MusicLM pipeline. By replacing MuLan, which currently conditions MusicLM's audio encoder on text descriptions, with our own foundation models, we could fine-tune a pipeline to generate audio from lyric priors.

References

[1] Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, and Christian Frank. Musiclm: Generating music from text, 2023.

[2] Rémi Delbouys, Romain Hennequin, Francesco Piccoli, Jimena Royo-Letelier, and Manuel Moussallam. Music mood detection based on audio and lyrics with deep neural net, 2018.

[3] Simon Durand, Daniel Stoller, and Sebastian Ewert. Contrastive learning-based audio to lyrics alignment for multiple languages. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[4] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[5] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer, 2021.

[6] Qingqing Huang, Aren Jansen, Joonseok Lee, Ravi Ganti, Judith Yue Li, and Daniel P. W. Ellis. Mulan: A joint embedding of music audio and natural language, 2022.

[7] Gaojun Liu and Zhiyuan Tan. Research on multi-modal music emotion classification based on audio and lyirc. In *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, volume 1, pages 2331–2335, 2020.

[8] Gabriel Meseguer-Brocal, Alice Cohen-Hadria, and Geoffroy Peeters. Dali: A large dataset of synchronized audio, lyrics and notes, automatically created using teacher-student machine learning paradigm. 2018.

[9] Loreto Parisi, Simone Francia, Silvio Olivastri, and Maria Stella Tavella. Exploiting synchronized lyrics and vocal features for music emotion detection, 2019.

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[12] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020.

[13] Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. Wav2clip: Learning robust audio representations from clip. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4563–4567, 2022.

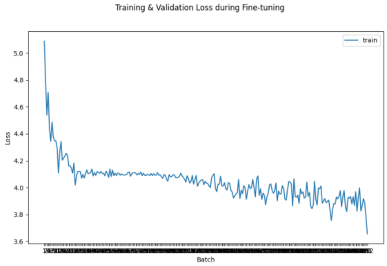
[14] Yi Yu, Suhua Tang, Francisco Raposo, and Lei Chen. Deep cross-modal correlation learning for audio and lyrics in music retrieval. *ACM Trans. Multimedia Comput. Commun. Appl.*, 15(1), feb 2019.

[15] Junfeng Zhang, Lining Xing, Zhen Tan, Hongsen Wang, and Kesheng Wang. Multi-head attention fusion networks for multi-modal speech emotion recognition. *Computers Industrial Engineering*, 168:108078, 2022.

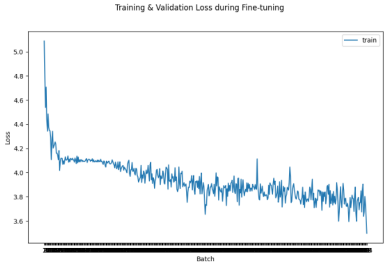
A Appendix

Training Results (base):

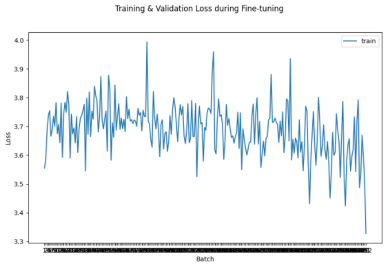
epoch 1:



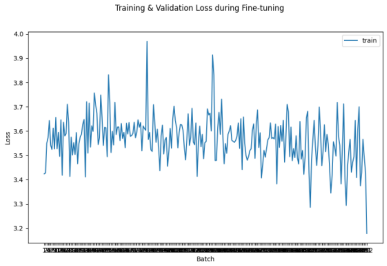
epoch 2:



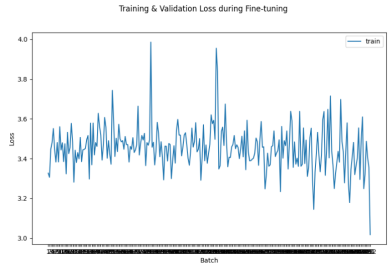
epoch 3:



epoch 4:

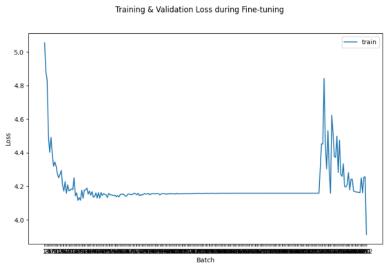


epoch 5:

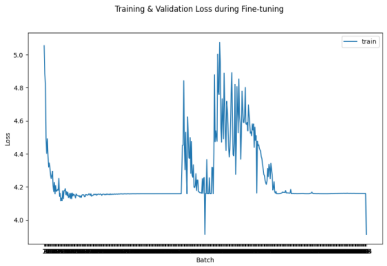


Training Results (sentiment):

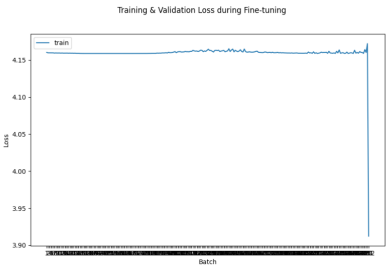
epoch 1:



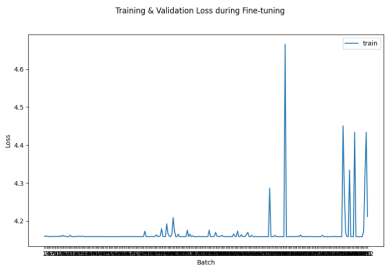
epoch 2:



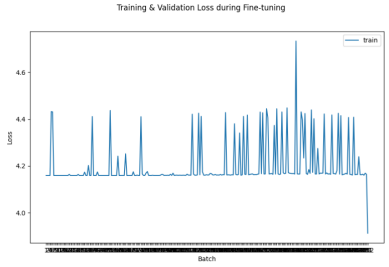
epoch 3:



epoch 4:



epoch 5:


Training & Validation Loss during Fine-tuning
train
4.6
4.4
4.2
4.0
Loss
Batch