3. PROPOSED METHOD

This section describes the proposed ALT method based on a cascading multi-task architecture.

3.1 Multi-task Learning Approach

Our method takes as input the mel-spectrogram of a music recording and that of the singing voice extracted from the recording with a DNN-based music separation method called Open-Unmix [18]. Since the singing voice separation is known to affect ALT, the original recording is thus used as well as the separated singing voice. Let $\mathbf{X} \in \mathbb{R}^{C \times F \times T}$ be the set of the input mel-spectrograms, where C is the number of channels ($C = 2$ in this paper), F is the number of frequency bins, and T is the number of frames.

We use as a basic building block a convolutional recurrent neural network (CRNN) consisting of a convolutional neural network (CNN) working as an encoder and a recurrent neural network (RNN) working as a decoder. The encoder extracts latent features from the input spectrograms and then the decoder estimates an output sequence while considering the sequential dependency of the latent features. The CRNN consists of residual CNN blocks with skip connections and RNN blocks (Fig. 2). Each CNN has a rectified linear unit (ReLU) and implemented with instance normalization. In contrast, each RNN block is a bidirectional LSTM (BiLSTM) [19] with layer normalization.

As shown in Fig. 1, our method uses two CRNNs. One CRNN is used for jointly estimating the posterior probabilities of the semitone-level pitches and those of the onset presence at the frame level. Given the estimated pitch and onset probabilities, the other CRNN is used for transcribing the lyrics from the spectrograms. Specifically, the estimated pitch and onset probabilities are fed together with the latent features extracted from the CNN into the RNN. Both CRNNs are trained jointly such the sum of the framewise pitch and onset estimation losses and the CTC-based lyrics transcription loss is minimized.

3.1.1 Pitch and Onset Estimation

The goal of pitch and onset estimation (supplementary task) is to estimate the framewise pitch and onset probabilities, denoted by $p(\text{pitch}|\mathbf{X}) \in \mathbb{R}^{K \times T}$ and $p(\text{onset}|\mathbf{X}) \in \mathbb{R}^{1 \times T}$, respectively, from the input data $\mathbf{X} \in \mathbb{R}^{C \times F \times T}$, where K is the number of the semitone-level pitches corresponding to MIDI note numbers plus no-pitch ($K = 128 + 1$).

Specifically, $\mathbf{X}$ is first fed to the series of multiple residual CNN blocks as follows:

$$\mathbf{Z}_{\text{PO}} = \text{CNN}(\mathbf{X}), \tag{1}$$

where $\mathbf{Z}_{\text{PO}} \in \mathbb{R}^{C' \times F \times T}$ is the output of the CNN and C' is the number of channels. Note that the zero padding is performed so that the output of each residual CNN block retains the shape of $\mathbf{X}$ except for the number of channels. Then $\mathbf{Z}_{\text{PO}}$ is reshaped into $\mathbf{Z}'_{\text{PO}} \in \mathbb{R}^{C'F \times T}$ and fed to the RNN as follows:

$$\mathbf{Y}_{\text{PO}} = \text{RNN}(\mathbf{Z}'_{\text{PO}}), \tag{2}$$

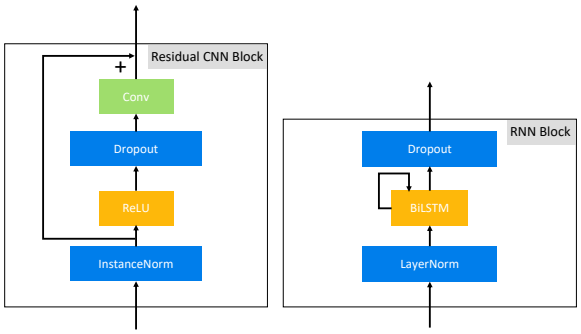

Figure 2. The residual CNN block and the RNN block.

where $\mathbf{Y}_{\text{PO}} \in \mathbb{R}^{C'' \times T}$ is the output of the RNN and C'' is the number of channels. Finally, $p(\text{pitch}|\mathbf{X})$ and $p(\text{onset}|\mathbf{X})$ are computed by feeding $\mathbf{Y}_{\text{PO}}$ to a fully-connected (FC) layer and the softmax and sigmoid functions, respectively, as follows:

$$[\mathbf{Y}_{\text{pitch}}, \mathbf{Y}_{\text{onset}}] = \text{FC}(\mathbf{Y}_{\text{PO}}), \tag{3}$$

$$p(\text{pitch}|\mathbf{X}) = \text{softmax}(\mathbf{Y}_{\text{pitch}}), \tag{4}$$

$$p(\text{onset}|\mathbf{X}) = \text{sigmoid}(\mathbf{Y}_{\text{onset}}), \tag{5}$$

where $\mathbf{Y}_{\text{pitch}} \in \mathbb{R}^{K \times T}$ and $\mathbf{Y}_{\text{onset}} \in \mathbb{R}^{1 \times T}$ are the intermediate outputs from the FC layer.

3.1.2 Lyrics Transcription

The goal of lyrics transcription is to estimate the framewise character probabilities, denoted by $p(\text{character}|\mathbf{X}) \in \mathbb{R}^{V \times L}$ from the input data $\mathbf{X} \in \mathbb{R}^{C \times F \times T}$, where V is the number of characters (dictionary size) including a special blank label used for CTC ($V = 33 + 1$ in this paper).

Specifically, in the same way as pitch and onset estimation, $\mathbf{X}$ is first fed to the series of multiple residual CNN blocks as follows:

$$\mathbf{Z}_{\text{L}} = \text{CNN}(\mathbf{X}), \tag{6}$$

where $\mathbf{Z}_{\text{L}} \in \mathbb{R}^{C' \times F \times T}$ is the output of the CNN. Then $\mathbf{Z}_{\text{L}}$ is reshaped into $\mathbf{Z}'_{\text{L}} \in \mathbb{R}^{C'F \times T}$, stacked with the estimated pitch probabilities $p(\text{pitch}|\mathbf{X}) \in \mathbb{R}^{K \times T}$ and onset probabilities $p(\text{onset}|\mathbf{X}) \in \mathbb{R}^{1 \times T}$, and fed to the RNN as follows:

$$\mathbf{Y}_{\text{L}} = \text{RNN}([\mathbf{Z}'_{\text{L}}, p(\text{pitch}|\mathbf{X}), p(\text{onset}|\mathbf{X})]), \tag{7}$$

where $\mathbf{Y}_{\text{L}} \in \mathbb{R}^{C'' \times L}$ is the output of the RNN and C'' is the number of channels. For computational simplicity, $\mathbf{Z}_{\text{L}}$ as well as $p(\text{pitch}|\mathbf{X})$ and $p(\text{onset}|\mathbf{X})$ are downsampled to $T/2$ frames with a 2-dimensional max-pooling layer. Finally, $p(\text{character}|\mathbf{X})$ is computed by feeding $\mathbf{Y}_{\text{L}}$ to a fully-connected (FC) layer and the softmax function as follows:

$$\mathbf{Y}_{\text{character}} = \text{FC}(\mathbf{Y}_{\text{L}}), \tag{8}$$

$$p(\text{character}|\mathbf{X}) = \text{softmax}(\mathbf{Y}_{\text{character}}), \tag{9}$$

where $\mathbf{Y}_{\text{character}} \in \mathbb{R}^{V \times T}$ is the intermediate output from the FC layer.

3.1.3 Joint Training with Domain Adaptation

The CRNN used for pitch and onset estimation and that for lyrics transcription are mutually dependent and thus trained jointly such that the sum of the framewise pitch and onset estimation losses (cross-entropies) and the lyrics transcription loss (CTC loss) is minimized. The CTC loss is computed from the framewise estimate $p(\text{character}|\mathbf{X})$ by efficiently accumulating the costs of all possible character sequences that can be reduced to the ground-truth character sequence by removing the blank labels. The pitch and onset probabilities $p(\text{pitch}|\mathbf{X})$ and $p(\text{onset}|\mathbf{X})$ and the underlying boundary information are considered to make $p(\text{character}|\mathbf{X})$ consistent with the ground-truth character sequence.

To mitigate the insufficiency of music data with lyrics annotations (DALI dataset [20]), we take the domain adaptation approach based on transfer learning [21]. Specifically, the CRNN used for lyrics transcription is trained using sufficient speech data (LibriSpeech corpus [22]) and then fine-tuned using both speech and music data.

3.2 Decoding

At run-time, we aim to estimate a series of note events with semitone-level pitches (MIDI note numbers) and onset times (frames) and transcribe the lyrics (Fig. 3). Specifically, the pitches with the maximum probabilities are taken from the estimated pitch probabilities $p(\text{pitch}|\mathbf{X})$. The onset times are determined with a peak picking strategy [23] with a window of 50 [ms] and a threshold of 0.4. A frame is counted as the onset time if the onset probability at this frame is maximal within 25 [ms] around this frame, where frames whose probabilities are less than 0.4 are excluded. The lyrics (best character sequence) are determined using a CTC decoder based on beam search [24] with a beam size of 25 frames and a 5-gram language model trained on the LibriSpeech corpus with a vocabulary of 200K words.

4. EVALUATION

This section reports a comparative experiment conducted for evaluating the effectiveness of the multi-task learning with pitch and onset estimation in ALT.

Details about data selection and training conditions can be found in the GitHub repository [1] of this paper.

4.1 Experimental Conditions

We explain the data used for evaluation, network configurations, compared methods, and evaluation measures.

4.1.1 Data

We used the DALI dataset [20] consisting of 5358 pieces of popular music in the English language along with fine-grained lyrics and pitch annotations. This dataset was made by collecting karaoke subtitle data and then searching for the corresponding audio data on the Internet, where an automatic alignment method was used for obtaining aligned

[1] https://github.com/TengyuDeng/lyrics-transcription-with-pitch-onset/

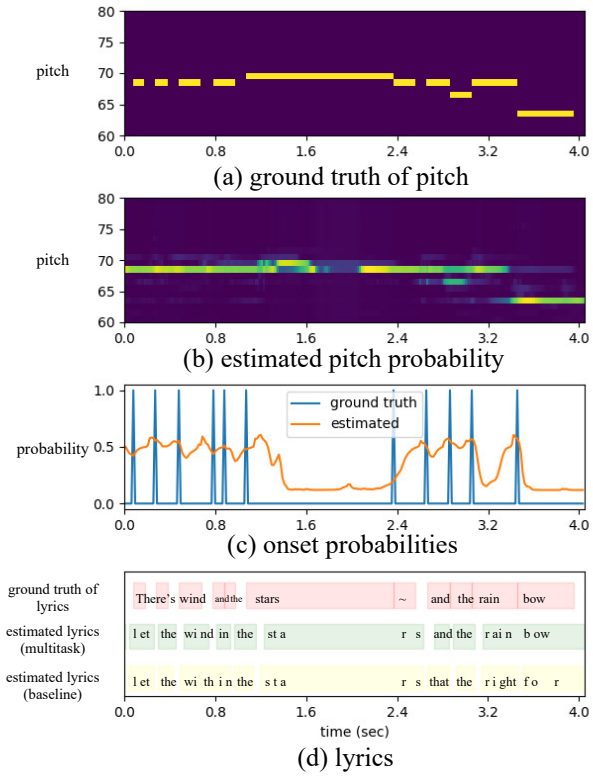

Figure 3. Example of estimations on a segment of a popular song from DALI. In (d), an estimated character is placed where the CTC decoder responds with the peak probability.

annotations. This dataset thus suffers from severe annotation errors [25]. First, the original karaoke data contained many problematic annotations such as global pitch shifts, wrong spellings, and onset time errors. Second, the automatic global alignment method often failed.

We thus filtered out music data with obviously wrong lyrics annotations and/or global pitch shift errors. The data selection procedure was based on the pitch estimation model and RWC Popular Music Database [26, 27]. This dataset contains 80 Japanese songs and 20 English songs, all of which are original popular songs and are provided with careful manual annotations. For simplicity, we used the framewise error rate given by

$$1 - \frac{\#\{\text{Frames Estimated Correctly}\}}{\#\{\text{Total Frames}\}}. \qquad (10)$$

We chose the 80 Japanese tracks of the RWC database and split them into a training set of 64 tracks and a test set of 16 tracks. The CRNN-based pitch estimation was trained on the training set, and attained 26.05% error rate on the test set, which was considered as a standard level.

The model trained on the training set from the RWC database was then exploited to test on the whole DALI dataset. The annotation of each song was shifted by -12, -11, ..., 0, ..., 11, and 12 semitones, and the predicted pitches of the pre-trained model were compared to all 25 pitch-shifted versions of annotations. The pitch shift value with the minimal framewise error rate was considered as the global pitch shift value of the annotation for a song, and

the minimal framewise error rate was recorded. Finally, the DALI dataset was filtered by the recorded framewise error rates. Specifically, songs with framewise error rates larger than a certain threshold were considered to be with problematic annotations.

Of all 5358 songs in the DALI dataset, 3272 songs could be accessed in our region and were in the English language. We selected 2515 songs with the data selection procedure, with a threshold of 50%. They were then split into a train set of 2263 songs, a validation set of 125 songs, and a test set of 126 songs. The total durations were 148.82 hours, 8.16 hours, and 8.47 hours, respectively.

For the transfer learning procedure, we used the LibriSpeech corpus [22] that contains 1000 hours of reading English speech sampled at 16 kHz. The CRNN described in Section 3.1.2 was trained on the train-clean-360 and train-other-500 sets of the LibriSpeech corpus, where all-zero matrices were used as $p(\text{pitch}|\mathbf{X})$ and $p(\text{onset}|\mathbf{X})$ in Eq. (7). This model was then fine-tuned with the training set of the DALI dataset.

4.1.2 Configurations

As described in Section 3.1, the audio signals, sampled at a rate of 48 kHz, were first separated into singing-voice-only signals with model umxhq from open-unmix [18]. Then for computational simplicity, the separated signals and the original mixed signals were resampled to 16 kHz, before they were converted to mel-spectrogram features, respectively. The audio signals were converted to mel-spectrogram with a window size of 32 ms and a hop length of 16 ms, and the resulted mel-spectrogram contained 80 mel-scaled features. We clipped the mel-spectrograms into pieces of 1000 frames, with a duration of about 16 s. Therefore, following the annotations in 3.1.1, we had $C = 2, F = 80, T = 1000$.

In the pitch and onset estimation network, 6 residual convolution blocks were stacked. The kernel sizes were (5,5), (5,5), (3,3), (3,3), (3,3), (1,1), and the numbers of output channels were 64, 32, 32, 32, 32, 1, respectively. After that, only 1 RNN block was applied to obtain the pitch and onsest estimation results. The dropout probability was set to 0 in each layer. In other words, we didn't adopt dropout in the pitch and onset estimation network.

In the lyrics transcription network, 6 residual convolution blocks were stacked. The kernel sizes were (5,5), (5,5), (3,3), (3,3), (3,3), (3,3), and the numbers of output channels were 64, 32, 32, 32, 32, 16, respectively. After that, 3 RNN blocks were stacked, and the number of hidden units in each LSTM was 512. The dropout probability was set to 0.2 in each layer.

When training the model, the Adam optimizer was used, and the parameters were $\beta_1 = 0.9, \beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The model was trained with a learning rate of 5×10^{-4}, and a warming up strategy was used. The learning rate linearly increased from 0 to 5×10^{-4} for the first 100 batches. We used an early stop strategy to manage the learning process. As mentioned in Section 4.1.1, the dataset was split into a training, a validation, and a test set. After training for an epoch, the model was tested on the validation set, and the learning process was terminated when the test statistics for the validation set stopped improving for 10 epochs.

Method	DALI-test	Jamendo
Ours (baseline)	69.22	77.3
Ours (oracle)	**64.41**	/
Ours (multi-task)	68.29	**76.2**
[6]	/	77.8
[7]	/	87.9

Table 1. Comparison of WER (%) with different methods.

4.1.3 Compared Methods

In order to test the performance of our system, we compared a couple of different settings. We also compared the proposed system with previous related works.

For lyrics transcription, in addition to the multi-task architecture, we also trained a model using zero dummy pitch and onset probabilities as a baseline model. Besides, a model was also trained using the ground truth pitches and onset times as oracle information. In order to compare with related works, the multi-task architecture was also tested on the jamendo dataset [6], and the results were compared with the end-to-end models in [6] and [7].

For pitch and onset estimation, we trained the model in Section 3.1.1 without the lyrics transcription part as the baseline model, and the results were compared with that in the multi-task scenario. We also evaluated the test data on VOCANO [28], a note-level vocal melody estimation toolkit available in public.

4.1.4 Evaluation Measures

The lyrics transcription was evaluated using word error rate (WER). The pitch and onset estimation was evaluated using the method first proposed in [29]. We considered the Correct Onset (COn) and the Correct Onset, Pitch (COnP) measures.

4.2 Experimental Results

Before fine-tuning on lyrics data, the lyrics transcription model was trained on LibriSpeech ASR corpus. The model reached a WER of 6.56% on the test-clean dataset and 20.86% on the test-other dataset.

WERs on the DALI-test dataset and the jamendo dataset are shown in Table. 1. On the DALI-test dataset, where the ground truth of pitch and onset information was available, the model reached the best performance when provided with this ground truth information. This shows that correct pitch and onset information can guide the system to find the correct alignment, so that the performance can be increased. In the multi-task architecture, the lyrics transcription model was trained with joint pitch and onset estimation. Although not as good as the oracle-given situation, the performance still gained some improvement. On the jamendo dataset, our multi-task architecture achieved

	COn			COnP		
	precision (%)	recall (%)	F value (%)	precision (%)	recall (%)	F value (%)
Ours(baseline)	53.21	**30.99**	**38.77**	36.92	**21.49**	**26.90**
Ours(multi-task)	**59.84**	28.69	38.41	**40.49**	19.57	26.14
VOCANO [28]	18.78	20.45	19.07	7.46	7.71	7.40

Table 2. Comparison of pitch and onset estimation results.

a similar improvement compared with our baseline model and beat main previous end-to-end ALT systems.

Table. 2 shows the pitch and onset evaluation results. Compared to the baseline model, the pitch and onset estimation jointly trained with the lyrics transcription model remained the same performance. However, for both the COn and COnP statistics, the multi-task scenario had a higher precision but a lower recall value than the baseline model. This shows that being jointly trained with the lyrics transcription model, especially the onset estimation was guided to be in favor of more confident onset positions. This lead to higher precision and lower recall values. It is notable that our models, both the baseline and the multi-task scenario, also gained better results than the results obtained when the VOCANO system was applied to the same test dataset.

5. CONCLUSION

This paper has presented a neural ALT method based on a multi-task learning architecture that estimates the pitch and onset information jointly and then transcribes the lyrics at the character level in a pitch- and onset-conditioned manner. The experiment using the DALI dataset showed that joint pitch and onset estimation can improve the performance of lyrics transcription. Although no significant overall improvement was attained in pitch and onset estimation, higher precision but lower recall rates were observed in the multi-task learning scenario. Our future work includes more comprehensive evaluation by gathering reliable data with accurate aligned lyrics and pitch annotations and using a data augmentation technique.

6. ACKNOWLEDGEMENT

This work is supported in part by JST PRESTO No. JPMJPR20CB and JSPS KAKENHI Nos. 19H04137, 20K21813, 21K02846, 21K12187, 22H03661.

7. REFERENCES

[1] E. Demirel, S. Ahlbäck, and S. Dixon, "Automatic lyrics transcription using dilated convolutional neural networks with self-attention," in *Proc. of International Joint Conference on Neural Networks*, 2020, pp. 1–8.

[2] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proc. of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. of Advances in Neural Information Processing Systems*, 2017.

[4] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang, S. Watanabe, T. Yoshimura, and W. Zhang, "A comparative study on transformer vs rnn in speech applications," in *Proc. of IEEE Automatic Speech Recognition and Understanding Workshop*, 2019, pp. 449–456.

[5] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[6] D. Stoller, S. Durand, and S. Ewert, "End-to-end lyrics alignment for polyphonic music using an audio-to-character recognition model," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 181–185.

[7] C. Gupta, E. Yılmaz, and H. Li, "Automatic lyrics alignment and transcription in polyphonic music: Does background music help?" in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 496–500.

[8] S. Basak, S. Agarwal, S. Ganapathy, and N. Takahashi, "End-to-end lyrics recognition with voice to singing style transfer," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 266–270.

[9] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, D. FitzGerald, and B. Pardo, "An overview of lead and accompaniment separation in music," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 8, pp. 1307–1335, 2018.

[10] C. Raffel, B. McFee, E. J. Humphrey, J. Salamon, O. Nieto, D. Liang, and D. P. W. Ellis, "mir_eval: A transparent implementation of common MIR metrics," in *Proc. of the 15th International Society for Music Information Retrieval Conference*, Taipei, Taiwan, 2014, pp. 367–372.

[11] J. Pons, R. Gong, and X. Serra, "Score-informed syllable segmentation for a cappella singing voice with convolutional neural networks," in *Proc. of the 18th*

International Society for Music Information Retrieval Conference, Suzhou, China, 2017, pp. 383–389.

[12] J. Huang, E. Benetos, and S. Ewert, "Improving lyrics alignment through joint pitch detection," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022.

[13] K. S. Rao, P. P. Das *et al.*, "Melody extraction from polyphonic music by deep learning approaches: A review," *arXiv preprint arXiv:2202.01078*, 2022.

[14] J. Salamon, E. Gomez, D. P. W. Ellis, and G. Richard, "Melody extraction from polyphonic music signals: Approaches, applications, and challenges," *IEEE Signal Processing Magazine*, vol. 31, no. 2, pp. 118–134, 2014.

[15] T.-H. Hsieh, L. Su, and Y.-H. Yang, "A streamlined encoder/decoder architecture for melody extraction," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 156–160.

[16] S. Yu, X. Sun, Y. Yu, and W. Li, "Frequency-temporal attention network for singing melody extraction," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 251–255.

[17] R. Nishikimi, E. Nakamura, M. Goto, and K. Yoshii, "Audio-to-score singing transcription based on a crnn-hsmm hybrid model," *APSIPA Transactions on Signal and Information Processing*, vol. 10, 2021.

[18] F.-R. Stöter, S. Uhlich, A. Liutkus, and Y. Mitsufuji, "Open-unmix – a reference implementation for music source separation," *Journal of Open Source Software*, 2019.

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

[20] G. Meseguer-Brocal, A. Cohen-Hadria, and G. Peeters, "Dali: A large dataset of synchronized audio, lyrics and notes, automatically created using teacher-student machine learning paradigm," in *Proc. of the 19th International Society for Music Information Retrieval Conference*, Paris, France, 2018, pp. 431–437.

[21] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[22] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 5206–5210.

[23] S. Böck, F. Krebs, and M. Schedl, "Evaluating the online capabilities of onset detection methods," in *Proc. of the 13th International Society for Music Information Retrieval Conference*, Porto, Portugal, 2012, pp. 49–54.

[24] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[25] G. M. Brocal, R. Bittner, S. Durand, and B. Brost, "Data cleansing with contrastive learning for vocal note event annotations," in *Proc. of the 21st International Society for Music Information Retrieval Conference*, Montreal, Canada, 2020, pp. 255–262.

[26] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "Rwc music database: Popular, classical and jazz music databases," in *Proc. of the 3rd International Conference on Music Information Retrieval*, Paris, France, 2002.

[27] M. Goto, "Aist annotation for the rwc music database," in *Proc. of the 7th International Conference on Music Information Retrieval*, Victoria, Canada, 2006, pp. 359–360.

[28] J.-Y. Hsu and L. Su, "Vocano: A note transcription framework for singing voice in polyphonic music," in *Proc. of the 22nd International Society for Music Information Retrieval Conference*, Online, 2021, pp. 293–300.

[29] E. Molina, A. M. Barbancho, L. J. Tardón, and I. Barbancho, "Evaluation framework for automatic singing transcription," in *Proc. of the 15th International Society for Music Information Retrieval Conference*, Taipei, Taiwan, 2014, pp. 567–572.

CONTRASTIVE AUDIO-LANGUAGE LEARNING FOR MUSIC

Ilaria Manco[1,2] **Emmanouil Benetos**[1] **Elio Quinton**[2] **György Fazekas**[1]
[1] School of EECS, Queen Mary University of London, London, U.K
[2] Music & Audio Machine Learning Lab, Universal Music Group, London, U.K.
{i.manco, emmanouil.benetos, george.fazekas}@qmul.ac.uk, elio.quinton@umusic.com

ABSTRACT

As one of the most intuitive interfaces known to humans, natural language has the potential to mediate many tasks that involve human-computer interaction, especially in application-focused fields like Music Information Retrieval. In this work, we explore cross-modal learning in an attempt to bridge audio and language in the music domain. To this end, we propose MusCALL, a framework for Music Contrastive Audio-Language Learning. Our approach consists of a dual-encoder architecture that learns the alignment between pairs of music audio and descriptive sentences, producing multimodal embeddings that can be used for text-to-audio and audio-to-text retrieval out-of-the-box. Thanks to this property, MusCALL can be transferred to virtually any task that can be cast as text-based retrieval. Our experiments show that our method performs significantly better than the baselines at retrieving audio that matches a textual description and, conversely, text that matches an audio query. We also demonstrate that the multimodal alignment capability of our model can be successfully extended to the zero-shot transfer scenario for genre classification and auto-tagging on two public datasets.

1. INTRODUCTION

Developing effective methods for finding music is at the core of Music Information Retrieval (MIR). Over the years, many approaches have been proposed to browse, search and discover music through a variety of interfaces. Beyond simple search by metadata, existing music retrieval systems allow to express queries via lyrics [1,2], audio examples [3], videos [4] and humming [5], among others [6–8]. Although each of these query types has its merits, none of these systems supports another popular way of searching for music today: through free-form text. For example, we commonly look for songs by typing text into a search engine [9] or by asking online song naming communities to identify a piece of music we do not have bibliographic information about [10]. It becomes evident then that enabling MIR systems to interpret natural language queries can have far-reaching benefits. This idea is not entirely new to MIR, with prior work [11–13] suggesting similar research directions in the past. So far, however, multimodal systems that integrate natural language have not been widely adopted within the MIR community, possibly due to a lack of suitable datasets or to the practical limitations of NLP methods predating modern language models.

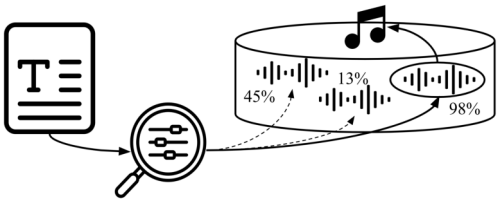

Figure 1: **Illustration of text-based music retrieval**, where audio items in a database are ranked according to their relevance to a free-form language query.

In light of recent breakthroughs in language modelling, we argue that audio-and-language learning has now the potential of closing the semantic gap in MIR [14], providing a bridge between computational representations of music signals and the high-level abstractions needed to use those representations in real-world scenarios. With this in mind, we propose MusCALL, a method for learning alignment between music-related audio and language data via multimodal contrastive learning. Our choice of a contrastive approach is inspired by the recent success of similar methods for joint visio-linguistic modelling (see Section 2.3). In designing MusCALL, we prioritise the ability to perform retrieval at scale and adopt a dual-encoder architecture, where modalities are processed independently. Compared to multimodal architectures with joint encoders and cross-modal attention mechanisms [15], this design allows to share embedding computations among pairs, resulting in a computationally more efficient model.

With this work, we aim to unify audio and language modelling, paving the way for MIR systems that can interpret language-based queries, as illustrated in Figure 1. Our primary contributions can be summarised as follows: (i) we explore multimodal contrastive learning in the context of music-related audio and language for the first time; (ii) we introduce a method for cross-modal retrieval of music, providing the first example of sentence-based music search; (iii) we perform an extensive set of experiments to systematically validate details of our approach and evaluate its performance on popular MIR tasks in a zero-shot setting. [1]

[1] Code and supplementary material are available at https://github.com/ilaria-manco/muscall

2. RELATED WORK

2.1 Natural Language Processing in MIR

Prior works in the MIR literature have explored leveraging natural language in the music domain from different angles. Early efforts focused on text as a modality in isolation, adopting NLP techniques to construct knowledge bases from music-related text corpora [16], build semantic graphs for artist similarity from biographies [17], or perform genre classification based on album reviews [18]. Recent efforts, more closely related to the present work, have instead started favouring multimodal approaches. These have explored deep learning with multimodal input data, typically audio combined with text such as reviews or lyrics, for applications as varied as music classification and recommendation [19], mood detection [20], music emotion recognition [21] and music captioning [22–24].

2.2 Audio-Text Cross-modal Learning

Another related line of work is cross-modal learning that leverages audio and tags as text input. Works in this area have proposed contrastive learning-based approaches to enrich audio representations via cross-modal alignment, either for general-purpose audio classification [25, 26] or for music-focused tasks [27]. Differently from our work, these approaches require finetuning on the downstream tasks and none of them directly uses cross-modal representations. Others have explored pre-trained word embeddings in triplet networks to perform tag-based music retrieval via a multimodal embedding space [28, 29]. At a high level, these works share a similar approach to ours and all aim to learn multimodal audio representations by leveraging text. Unlike our work, however, none of them makes use of natural language, using tags as text input instead.

Finally, similarly to our work, [30] also addresses the problem of matching audio and long-form text for music retrieval, but offers a fundamentally different approach, which relies on bridging the audio and text modalities via a common emotion embedding space.

2.3 Learning from Language Supervision

In the previous sections, we have highlighted some research efforts towards bringing together audio and language, but, ultimately, multimodal learning still occupies a marginal role in MIR and has yet to fully enjoy the benefits of modern language models. In adjacent fields such as computer vision, jointly modelling vision and language has instead become a very active area of research, with several successful attempts at using multimodality to develop task-agnostic models that can easily adapt to novel tasks [31–34]. The key insight behind these models is that language captures many of the abstractions humans use to navigate the world and can therefore act as a rich supervisory signal for general-purpose learning, even in tasks that are not directly based on language [35]. Among these works, CLIP [33] is a particularly influential example, having demonstrated for the first time that supervision from natural language can induce highly generic visual representations at scale.

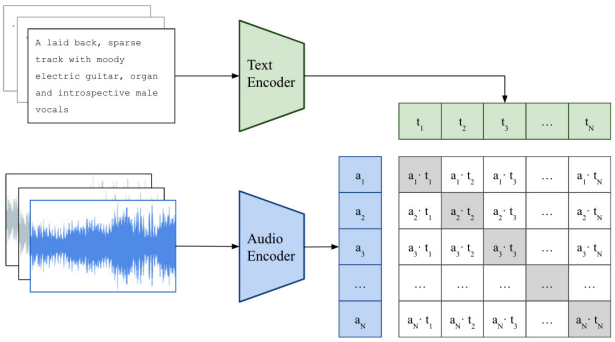

Figure 2: **Overview of MusCALL**. An audio encoder and a text encoder are trained via a contrastive loss to maximise the similarity between representations of N aligned (audio, text) pairs within a mini-batch. At test time, the similarity between embeddings in the learnt multimodal space is used to rank database items and perform cross-modal retrieval.

These breakthroughs suggest that natural language supervision has a large potential beyond the image domain. This has recently prompted the adoption of similar approaches in machine listening, where applications to both music [15] and non-music audio [36, 37] have started to emerge. A subset of works in this area have proposed to directly extend CLIP by incorporating audio as an additional modality [38–40]. Although similar in spirit to our work, these approaches exploit audio-visual correspondences in video, thus requiring large-scale visio-linguistic pre-training. Like these works, we also borrow from CLIP's contrastive dual-encoder approach, but we adapt this framework to the audio modality without any pre-training, while also addressing the specific set of challenges that arise from joint audio-linguistic modelling in the music domain.

3. MUSCALL

3.1 Multimodal Contrastive Learning (MCL)

Contrastive learning has emerged as a powerful paradigm in self-supervised representation learning for images [41] and audio [42–45]. At their core, all flavours of contrastive learning rely on constructing different views of the data and forcing representations of positive pairs of such views to have, on average, higher similarity than those of negative pairs. This encourages a model to learn representations that are *discriminative* with respect to changes to the content and *invariant* to nuisance transformations. In representation learning, this is a useful property, since it results in more robust and effective representations [46–48]. Most formulations of contrastive learning in unimodal scenarios use data augmentations to create multiple views of the inputs. When switching to the multimodal setting, however, different views naturally co-occur in the data, making this a natural testbed for contrastive learning [33, 49, 50].

3.2 Extending MCL to Music and Language

In this work, we explore multimodal contrastive learning (MCL) in the context of audio-linguistic data. More specifically, we consider pairs of music audio tracks and their