# LOW RESOURCE AUDIO-TO-LYRICS ALIGNMENT FROM POLYPHONIC MUSIC RECORDINGS

*Emir Demirel[1], Sven Ahlbäck[2], Simon Dixon[1]*

[1]Centre for Digital Music, Queen Mary University of London, UK
[2]Doremir Music Research AB, SE

## ABSTRACT

Lyrics alignment in long music recordings can be memory exhaustive when performed in a single pass. In this study, we present a novel method that performs audio-to-lyrics alignment with a low memory consumption footprint regardless of the duration of the music recording. The proposed system first spots the anchoring words within the audio signal. With respect to these anchors, the recording is then segmented and a second-pass alignment is performed to obtain the word timings. We show that our audio-to-lyrics alignment system performs competitively with the state-of-the-art, while requiring much less computational resources. In addition, we utilise our lyrics alignment system to segment the music recordings into sentence-level chunks. Notably on the segmented recordings, we report the lyrics transcription scores on a number of benchmark test sets. Finally, our experiments highlight the importance of the source separation step for good performance on the transcription and alignment tasks. For reproducibility, we publicly share our code with the research community.

***Index Terms*—** audio-to-lyrics alignment, music information retrieval, automatic lyrics transcription, long audio alignment, automatic speech recognition

## 1. INTRODUCTION

Audio-to-lyrics alignment has a variety of applications within the music technology industry. Some of these applications include lyrics prompting for karaoke applications, captioning, and music score and video editing. Moreover, lyrics alignment systems can be leveraged to generate new training data for several tasks within MIR research, such as lyrics transcription, singing voice detection, source separation, music transcription and cover song identification.

The task of aligning song lyrics with their corresponding music recordings is among the most challenging tasks in music information retrieval (MIR) research due to three major factors: the multimodality of the information to be processed – namely *music* and *speech*, the presence of the musical accompaniment in the acoustic scene and the length of the music recording to be aligned. For processing linguistically relevant information, previous studies have taken the approach of adapting automatic speech recognition (ASR) paradigms to singing voice signals [1, 2, 3, 4]. Regarding the musical accompaniment, researchers have aligned lyrics on either source separated vocal tracks [4] or utilized acoustic models trained on polyphonic recordings [2, 3, 5].

ED received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 765068.

For relevant alignment tasks within MIR research, previous studies have presented efficient ways to handle the duration issue in alignment by using methods based on dynamic time warping (DTW) [6, 7]. The results in the MIREX 2017[1] public evaluation of audio-to-lyrics alignment systems showed that Viterbi-based alignment outperforms DTW [8]. Most lyrics alignment research since then has utilized a similar approach due to its performance and efficiency, however even Viterbi decoding may become intractable when processing long audio recordings. Within this context, performing robust and efficient lyrics alignment in long music recordings still remains as a bottleneck and preventing factor for audio-to-lyrics alignment technology in industrial applications involving large-scale web services or mobile deployment. In this paper, we aim to contribute to this field of research by proposing a low resource solution that is robust and efficient in terms of the audio length and the musical accompaniment to singing. Leveraging its duration-invariant capability, we further show that this approach could be exploited to generate sentence-level lyrics annotations for extending existing lyrics transcription training sets.

This paper continues as follows: in Section 2, we provide a brief overview of the state-of-the-art in audio-to-lyrics alignment. We explain the overall details of the proposed biased lyrics search pipeline in Section 3. Then, we evaluate the utilization capability of the overall system through lyrics alignment and transcription experiments. Also in this section, we conduct the first experiments evaluating different source separation algorithms in the lyrics alignment and transcription context. Finally, we report results that are competitive with the state-of-the-art in lyrics alignment and best recognition scores for lyrics transcription on a public benchmark evaluation dataset.

## 2. RELATED WORK

Early audio-to-lyrics alignment systems in research use Hidden Markov Model (HMM) based acoustic models which are utilized to extract frame-level phoneme (or character) posterior probabilities. Then a forward-pass decoding algorithm is applied on these posteriograms, obtaining phoneme alignments. Then using a language model (LM), phoneme posteriograms can be converted to word posteriograms to retrieve word-level alignments [1, 3, 4]. One recent successful system [2] showed a considerable performance boost compared to previous research using an end-to-end approach trained on a large corpus, where alphabetic characters are used as sub-word units of speech. Vaglio et al. [5] also employed an end-to-end approach for lyrics alignment applied in a multilingual context, but using phonemes as an additional intermediate representation,

[1]Can be accessed from `https://www.music-ir.org/mirex/wiki/2017:Automatic_Lyrics-to-Audio_Alignment_Results`

and obtained competitive results. In addition, the authors have used a public dataset [9] that is much smaller than the training set used in [2]. Gupta et al. [3] reported state-of-the-art results using an acoustic model trained on polyphonic music using genre-specific phonemes. According to the authors, their system applies forced alignment with a large beam size as their system attempts to process the entire music recording at once. Although achieving impressive results, the application of forced alignment in a single pass can be memory exhaustive.

A similar challenge within automatic speech recognition (ASR) research is the *lightly supervised alignment* task [10] where the goal is to obtain timings of human-generated captions in long TV broadcasts that would be displayed to TV viewers as subtitles. Moreno et al. [11] present a system for long audio alignment which searches for the input word locations through a recursive application of speech recognition on a gradually restricted search space and language model. The method is then further improved in terms of robustness and efficiency in the search space using the factor transducer approach [12, 13]. The factor transducer introduces an important drawback within the lyrics alignment task as it exerts weak timing constraints during decoding, i.e. the output word alignments are not constrained to be in order. This would rise as a significant issue especially during successive patterns of similar word sequences, which occur frequently in song lyrics [14].

Another major challenge during lyrics alignment (and also transcription) is the influence of accompanying non-vocal musical sounds. One way to minimize this during the transcription and alignment is by isolating the vocal track using a vocal source separation system. There has been a number of powerful open-source music source separation toolkits made available for research recently [15, 16, 17], yet the output vocal track is not guaranteed to be free of sonic artifacts introduced during separation. In turn, these artifacts might affect the word intelligibility and thus the accuracy during the automatic lyrics transcription and alignment (ALTA) tasks. The effects of different source separation algorithms on lyrics transcription and alignment has not been studied extensively; we provide a comparison in this paper.

## 3. SYSTEM DETAILS

Our overall lyrics alignment pipeline is illustrated in Figure 1. We initially extract vocals from the original polyphonic mix and retrieve vocal segments using energy-based voice activity detection. We search for lyrics within these segments by applying automatic transcription using a decoding graph constructed via a biased language model (LM). The matching portions of the transcribed and original lyrics and their location in the audio signal are obtained through the text and forced alignment techniques respectively. The music recording is then segmented with respect to these matching portions and a final-pass alignment is applied to obtain the timings of all the words in the original lyrics. In order to be able to align and recognize out-of-vocabulary (OOV) words during decoding, we extend the pronunciation dictionary for the words in the input lyrics.

### 3.1. Extending the Pronunciation Dictionary

For achieving a robust lyrics alignment system, out-of-vocabulary (OOV) words have to be taken into account. Lyrics may contain linguistically ambiguous sequences of words such as 'la', 'na' and 'ooh', as well as words with repeated syllables or phonemes like 'do re mi fa so oh oh oh', out-of-language (OOL) words or special names. Thus, we extend the pronunciation dictionary with respect to the input lyrics, and construct a pronunciation transducer prior to decoding. We trained a grapheme-to-phoneme conversion (*g2p*) model [18], using the CMU English pronunciation dictionary[2], to generate new pronunciations for each OOV word in the input lyrics.

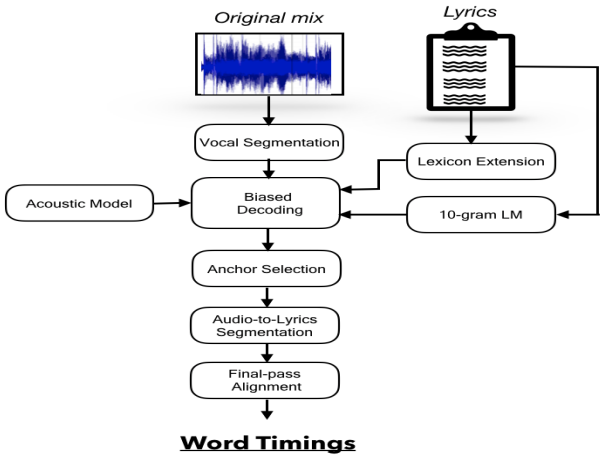


**Fig. 1**: Pipeline of our lyrics transcription and alignment system

### 3.2. Vocal Segmentation

Initially, we separate the vocal track from the original mix and determine the voice activity regions (VAR) based on the log-energy (the zeroth component of MFCC features). We merge consecutive VARs if the silence between them is less than $\tau_{silence}$ seconds, although we do not merge segments that are already more than $\tau_{max}$ seconds long. The values for $\tau$ are determined empirically. Note that if $\tau_{silence}$ is too short, there occurs the risk of over-segmenting the audio which could potentially reduce the word recognition rate. We have set $\tau_{silence} = 0.8$ and $\tau_{max} = 6$, based on our empirical observations.

### 3.3. Biased Decoding

The goal of this stage is to detect the word positions in the lyrics and the audio that we will use for segmentation later on. In order to be able to detect the words in input lyrics with a higher recognition rate, we restrict the search space by building an n-gram language model (LM) on the input lyrics. First, we transcribe the contents within the VARs using the pretrained acoustic model in [19] and the biased LM. Song lyrics often contain repetitive word sequences for which the system might recognize a word in the wrong position or repetition in the lyrics, potentially causing accumulated errors in segmentation. For robustness against such cases, we exert strong constraints on the word order via building the LM with 20-grams. Moreover, prior to processing, we add a *<NOISE>* tag at the beginnings and endings of each lyrics line to further reduce the risk of alignment errors between long non-vocal segments.

Building the LM from only the input lyrics has two major advantages: First, it increases the chance of recognizing the words in input lyrics while minimizing the risk of wrong word predictions. Secondly, through constructing the LM on the fly, we do not need an external pretrained LM to perform lyrics alignment. This aspect of our system makes it applicable in a multilingual setting in the presence of a g2p model for target languages.

[2] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

### 3.4. Anchor Selection

Next, we apply text alignment between the transcriptions and the reference lyrics to determine the matching portions, i.e. *anchoring* words. To impose further constraints on word order, we consider $N_{anchor}$ number of successive matching words as the anchoring instances between the lyrics and the audio signal. On the corresponding VARs, we apply forced alignment using these anchoring words to get their positions on the audio signal. We refer these portions of the audio signal matched with its lyrics as *islands of confidence*. Using a large $N_{anchor}$ would carry the risk of detecting lesser anchoring words while a very small value could cause over-segmentation. In our experiments, we chose $N_{anchor} = 5$ (Fig. 2).

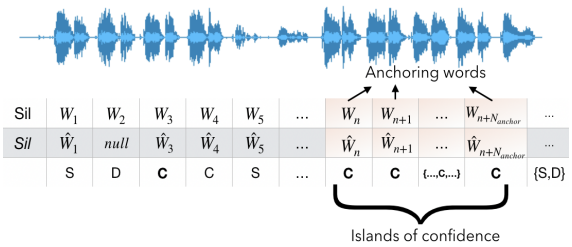


**Fig. 2**: Anchor selection. $W_n$ and $\hat{W}_n$ are the reference and predicted words respectively. D and S stand for word deletions and substitutions after text alignment. C are the labels for correctly recognized (matching) words.

### 3.5. Audio and Lyrics Segmentation

Once detected, the anchoring words are utilized to split the music recording into shorter segments. In order to further alleviate the risk of over-segmenting the recording, we allow a maximum of $N_{segment}$ anchoring words to be in each segment. We start segmenting monotonically from the beginning to the very end of the recording. Once $N_{segment}$ words are spotted, the audio is split with respect to the beginning of the first and the ending of the $N_{segment}$-th word. Empirically, we found this approach to function well for $N_{segment} > 10$. If the first word in the original lyrics is not spotted, the beginning of the first segment in the initial voice-activity-based segmentation is used. A similar approach is applied for the last word.

### 3.6. Final-Pass Alignment

Finally, we can apply forced alignment on shorter audio segments which are extracted in the previous step using a smaller beam size. In our experiments, we were able to obtain alignments without any memory issues using a beam size of 30 and a retry beam size of 300, which are much lower than the values reported in [4].

## 4. EXPERIMENTAL SETUP

To test the quality of the output audio-to-lyrics segmentation, we evaluate the precision of the word timings produced by the final pass alignment. Further, we evaluate the transcription performance on the initial voice-activity segments and the final segmentation, to gain an insight into whether these segments can be used for further training.

*Vocal Extraction :* The alignment and decoding are applied on the separated vocal tracks, in order to minimize the influence of accompanying musical sounds. Additionally, the acoustic model employed in decoding is trained on monophonic singing voice recordings [19]. The output of the vocal source separation has a direct effect on the performance of decoding, and hence the overall performance of lyrics alignment. There are two mainstream approaches in vocal separation: spectrogram-based and waveform-based models. While spectrogram-based approaches have been more widely used [20, 15], there have been recent successful waveform-based music source separation methods, motivated by capturing the phase information in the signal [16]. To test the effect of the source separation step, we compare a waveform-based model (Demucs) [17] and a state-of-the-art spectrogram-based model (Spleeter) [16] in experiments.

*Lyrics Transcriber :* We use the acoustic model of the lyrics transcriber in [19]. The system was trained on 150 hours of a cappella singing recordings with a non-negligible amount of noise [21]. After decoding with a 4-gram LM, we rescore lattices with RNNLM [22] and report this value as the final transcription result.

*Data :* For testing, we use the benchmark evaluation set for the lyrics transcription and alignment tasks, namely JamendoLyrics [2] which consists of 20 music recordings (1.1 hours) from a variety of genres including metal, pop, reggae, country, hiphop, R&B and electronic. In addition, the lyrics transcription results are reported also on the Mauch [23] dataset which consists of 20 English language pop songs.

## 5. RESULTS & DISCUSSION

### 5.1. Audio-to-Lyrics Alignment

The word alignments are evaluated in terms of the standard audio-to-lyrics alignment metrics: mean and median average absolute error (AE) [24] in seconds, and correctly predicted segments (PCS) with a tolerance window of 0.3 seconds [23]. These metrics are computed for each sample in the evaluation set and the mean for each metric is reported as the final result.

We compare the performance of our system with the most recent successful lyrics alignment systems. SD1 [2] applies alignment on polyphonic recordings using an end-to-end system trained on a private dataset consisting of over 44 000 songs. In SD2, alignment is performed on source separated vocal track using the Wave-U-Net [20] architecture. VA [5] also uses an end-to-end model, but trained on the DALI (v1.0) dataset, which has over 200 hours of polyphonic music recordings and extracts the vocals using Spleeter. GC1 [3] uses the same training data, for constructing an acoustic model in the hybrid-ASR setting [25] and performs alignment on the original polyphonic mix as well. In addition to these models, we refer to our models which align words to the vocal tracks separated by Demucs and Spleeter as DE1 and DE2 respectively.

| | Mean AE | Median AE | PCS |
|---|---|---|---|
| SD1 [2] | 0.82 | 0.10 | 0.85 |
| SD2 [2] | 0.39 | 0.10 | 0.87 |
| VA [5] | 0.37 | N/A | 0.92 |
| GC1 [3] | **0.22** | **0.05** | **0.94** |
| DE1 | 0.31 | **0.05** | 0.93 |
| DE2 | 0.38 | **0.05** | 0.90 |

**Table 1**: Lyrics alignment results on the Jamendo dataset

According to Table 1, our system that uses the waveform-based source separation model (Demucs) for vocal extraction outperforms other methods that use end-to-end learning, and obtains competitive results to the state-of-the-art (GC1). Using Spleeter, we were able to

achieve similar results to VA [5], which also uses Spleeter for source separation. Note that the training data we have used is smaller and less diverse than the datasets used by all other systems, highlighting that there is room for performance improvement in our method. Moreover, the alignment performance is better when using Demucs instead of Spleeter in terms of mean AE and PCS, even though the median AE is the same.

Figure 3 shows a comparison of the system presented in this paper, DL{1,2}, and GL1 in regard to memory efficiency. The memory usage on the RAM is monitored every 10 seconds during a run over the Jamendo dataset. We have executed the code for the experiments on Intel® Xeo™ E5-2620 with 24 GM of RAM capacity. Below, Figure 3 and Table 2 show that the memory consumption of our system is less than GL1 by a margin more than an order of magnitude. The larger standard deviation in our case is potentially due to the varying lengths and complexities of the segmented music signals input to decoding.

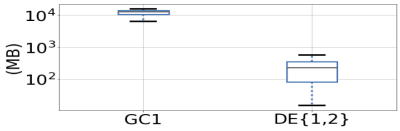


**Fig. 3**: Memory usage on RAM in megabytes (MB)

| | GC1 | DE{1,2} |
|---|---|---|
| Mean (Std.%) | 13,740 (8.8%) | **343** (31%) |
| Max | 16,745 | **748** |

**Table 2**: Statistics on memory usage in MB

### 5.2. Automatic Lyrics Transcription

In order to gain an insight as to whether the final lyrics-to-audio segments can be leveraged as sentence-level annotations for training data, we conduct lyrics transcription experiments. We compare word recognition rates for a pure inference case on VARs and on the segments extracted as described in Section 3.5. Additionally, we provide comparisons with the state-of-the-art in lyrics transcription from polyphonic music recordings: SD1 [2], GC1 and GC2 [3] (vocals extracted using [20]). For the evaluation, we use word (WER) and character (CER) error rate computed using the Kaldi toolkit.

According to Table 3, unlike the alignment results, using Spleeter for separating the vocals seems to be more beneficial for lyrics transcription. Notice that while the gap is small between the recognition rates of DE1 and DE2 on the Mauch dataset, it is much higher on the Jamendo dataset. For both DE1 and DE2, the transcription rates are consistently higher on the audio-to-lyrics segments compared to VAR, implying that our segmentation method can be exploited in a semi-supervised setting.

During pure inference on VAR, our system outperforms the state-of-the-art on Jamendo, while the performance on the Mauch dataset is still far behind. In all cases, our system outperforms other systems that apply transcription on separated vocal tracks by a great margin. Note this could be due to the fact that all of the previous methods use datasets consisting of polyphonic recordings even though they are much larger in size.

Overall, the results show that the application of lyrics transcription and alignment on separated vocal tracks can achieve competitive performance to state-of-the-art systems working directly on polyphonic music input. It should be noted that the choice of source separation has a crucial but inconsistent effect on the final transcription and alignment results. While it is not yet possible to draw a general conclusion regarding which method works better for which task, we have shown that the effect of vocal extraction on the performance of these tasks is worthy to consider when developing music source separation algorithms.

| | *WER* | | *CER* | |
|---|---|---|---|---|
| | Mauch | Jamendo | Mauch | Jamendo |
| SD1 [2] | 70.09 | 77.80 | 48.90 | 49.20 |
| GC1 [3] | ***44.02*** | *59.57* | N/A | N/A |
| GC2 [3] | 78.85 | 71.83 | N/A | N/A |
| DE1 - VAR | 60.92 | 62.55 | 44.15 | 47.02 |
| DE1 - segmented | 50.44 | 55.47 | 38.65 | 42.11 |
| DE2 - VAR | 57.36 | **51.76** | **41.52** | **37.26** |
| DE2 - segmented | 49.92 | **44.52** | **38.41** | **32.90** |

**Table 3**: Lyrics transcription results

## 6. CONCLUSION

We presented a novel system that segments polyphonic music recordings with respect to its given lyrics, and further aligns the words with the audio signal. We have reported competitive results with the state-of-the-art while outperforming other end-to-end based models. Through lyrics transcription experiments, we provided evidence supporting the capability of our system to be exploited for generating additional training data for a variety of MIR tasks. As a pilot study, we have conducted the first experiments on the effect of different source separation models on the lyrics transcription and alignment tasks. The recognition rates on separated vocals show that our system performs better than the previous best systems by a considerable margin while showing comparable performance with the state-of-the-art in ALT from polyphonic music recordings on a public benchmark evaluation set. Moreover, it is shown that our acoustic model can be exploited for both of the ALTA tasks.

It should be noted that the system presented here achieves this performance requiring considerably lower computational and data resources compared to the best performing published work to this date, which is shown via a quantitative comparison regarding the memory consumption during runtime. From this perspective, this framework is shown to be applicable in use cases where low resource solutions are required, such as large-scale web services and mobile applications. As an additional advantage, our approach does not rely on a pretrained language model, which makes it possible for it to be extended for a multilingual setup. As a final remark, we have publicly shared the code[3] for open science and reproducibility.

## 7. REFERENCES

[1] Georgi Bogomilov Dzhambazov and Xavier Serra, "Modeling of phoneme durations for alignment between polyphonic audio and lyrics," in *Sound and Music Computing Conference (SMC)*, 2015.

[3]Can be accessed from `https://github.com/emirdemirel/ASA_ICASSP2021`

[2] Daniel Stoller, Simon Durand, and Sebastian Ewert, "End-to-end lyrics alignment for polyphonic music using an audio-to-character recognition model," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019.

[3] Chitralekha Gupta, Emre Yılmaz, and Haizhou Li, "Automatic lyrics transcription in polyphonic music: Does background music help?," *arXiv preprint arXiv:1909.10200*, 2019.

[4] Bidisha Sharma, Chitralekha Gupta, Haizhou Li, and Ye Wang, "Automatic lyrics-to-audio alignment on polyphonic music using singing-adapted acoustic models," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 396–400.

[5] Andrea Vaglio, Romain Hennequin, Manuel Moussallam, Gael Richard, and Florence d'Alché Buc, "Multilingual lyrics-to-audio alignment," in *International Society for Music Information Retrieval Conference (ISMIR)*, 2020.

[6] Meinard Müller, Frank Kurth, and Tido Röder, "Towards an efficient algorithm for automatic score-to-audio synchronization.," in *ISMIR*, 2004.

[7] Simon Dixon, "An on-line time warping algorithm for tracking musical performances.," in *IJCAI*, 2005, pp. 1727–1728.

[8] Anna Kruspe and IDMT Fraunhofer, "Llyrics alignment using hmms, posteriorgram-based dtw and phoneme-based levenshtein alignment," .

[9] 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," *arXiv preprint arXiv:1906.10606*, 2019.

[10] Peter Bell, Mark JF Gales, Thomas Hain, Jonathan Kilgour, Pierre Lanchantin, Xunying Liu, Andrew McParland, Steve Renals, Oscar Saz, Mirjam Wester, et al., "The MGB challenge: Evaluating multi-genre broadcast media recognition," in *Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 687–693.

[11] Pedro J Moreno, Chris Joerg, Jean-Manuel Van Thong, and Oren Glickman, "A recursive algorithm for the forced alignment of very long audio segments," in *Fifth International Conference on Spoken Language Processing*, 1998.

[12] Peter Bell and Steve Renals, "A system for automatic alignment of broadcast media captions using weighted finite-state transducers," in *Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015.

[13] Pedro J Moreno and Christopher Alberti, "A factor automaton approach for the forced alignment of long speech recordings," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009.

[14] Emir Demirel, Sven Ahlback, and Simon Dixon, "A recursive search method for lyrics alignment," in *MIREX 2020 Audio-to-Lyrics Alignment and Lyrics Transcription Challenge*, 2020.

[15] Fabian-Robert Stöter, Stefan Uhlich, Antoine Liutkus, and Yuki Mitsufuji, "Open-unmix: A reference implementation for music source separation," *Journal of Open Source Software*, 2019.

[16] Romain Hennequin, Anis Khlif, Felix Voituret, and Manuel Moussallam, "Spleeter: A fast and state-of-the art music source separation tool with pre-trained models," 2019, Late-Breaking/Demo Session at ISMIR.

[17] Alexandre Défossez, Nicolas Usunier, Léon Bottou, and Francis Bach, "Music source separation in the waveform domain," *arXiv preprint arXiv:1911.13254*, 2019.

[18] Josef R Novak, D Yang, N Minematsu, and K Hirose, "Phonetisaurus: A WFST-driven phoneticizer," *The University of Tokyo, Tokyo Institute of Technology*, 2011.

[19] Emir Demirel, Sven Ahlbäck, and Simon Dixon, "Automatic lyrics transcription with dilated convolutional networks with self-attention," in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020.

[20] Daniel Stoller, Sebastian Ewert, and Simon Dixon, "Wave-U-Net: A multi-scale neural network for end-to-end audio source separation," *arXiv preprint arXiv:1806.03185*, 2018.

[21] "Smule sing! 300x30x2 dataset," accessed August, 2020, `https://ccrma.stanford.edu/damp/`.

[22] Hainan Xu, Tongfei Chen, Dongji Gao, Yiming Wang, Ke Li, Nagendra Goel, Yishay Carmiel, Daniel Povey, and Sanjeev Khudanpur, "A pruned RNNLM lattice-rescoring algorithm for automatic speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018.

[23] Matthias Mauch, Hiromasa Fujihara, and Masataka Goto, "Integrating additional chord information into HMM-based lyrics-to-audio alignment," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 200–210, 2011.

[24] Annamaria Mesaros and Tuomas Virtanen, "Automatic alignment of music audio and lyrics," in *Proceedings of the 11th Int. Conference on Digital Audio Effects (DAFx-08)*, 2008.

[25] Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahremani, Vimal Manohar, Xingyu Na, Yiming Wang, and Sanjeev Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Interspeech*, 2016.