# Analysing Temporal Sensitivity of VQ-VAE Sub-Phone Codebooks

*Jason Fong, Jennifer Williams, Simon King*

Centre for Speech Technology Research, University of Edinburgh, UK

{jason.fong, j.williams, simon.king}@ed.ac.uk

## Abstract

In this work we present an analysis of temporal sensitivity of VQ-VAE sub-phone token sequences. Previous work has demonstrated that VQ-VAE systems learn a type of sub-phone representation. However, a detailed examination of the representations themselves is currently lacking. We address this gap by exploring linguistic unit reorganisation. Our experiments show that sub-phone codebook sequences are temporally correlated enough to identify VQ codes that correspond to distinct linguistic units. We found that it is possible to extract VQ codes and re-arrange these linguistic units in a meaningful way (i.e. changing the word-order of a sentence). This work puts us one step closer to understanding how to modify pronunciations at a fine granularity, such as below the phone-level unit.

**Index Terms**: VQ-VAE, speech synthesis, representation learning

## 1. Introduction

Speech applications such as automatic speech recognition (ASR) and text-to-speech synthesis (TTS) have traditionally employed phones to describe speech. While they were originally conceived for rapid linguistic field transcription, they are now used as a representation of pronunciation. Phones omit much of the nuance that is inherent in human speech production. For example, phones do not represent the effects of co-articulation and are inadequate for capturing other connected speech effects such as partial vowel reduction or elision. There are established approaches to work around these limitations, including expanding the phone set to include allophones and syllabic consonants, or by constructing a set of application-specific context-dependent categories such as diphones (for TTS), or triphones / quinphones (for ASR). Until recently, these were the only viable choices for a discrete representation of speech in speech technology applications.

However, recent advances in neural modelling, notably self-supervised and semi-supervised techniques, offer an ability to learn speech representations which – by definition – *must* capture nuances of speech: the training objective is to be able to reconstruct speech [1, 2] or make contextual predictions [3, 4, 5]. Now with the aid of such learned, informationally-dense speech representations, we are in a position to rethink our approach to representing speech for applications such as TTS. Recent work has already shown that neural speech models learn semi-supervised representations that capture high-level linguistic aspects of speech from speech waveforms. These representations can greatly improve ASR with little data [5], and can also be used to generate speech [6, 7]. But no prior work has sought to use these representations to create new speech applications altogether.

In this paper we encode speech into a sequence of tokens drawn from a finite set of categories using a VQ-VAE model [2], then reconstruct it from that sequence of tokens. We wish to evaluate the adequacy of the token sequence as a representation of speech pronunciation. That evaluation takes form of concatenating token sequences to construct word sequences not seen in the training data. That is, we use "concatenative VQ-VAE synthesis" as a methodology for evaluating whether the learned inventory of categories would be a useful pronunciation representation for neural TTS.

The novelty of this work is in the manipulation of learned VQ token sequences. Our results bring us a step closer to nuanced control of speech pronunciation, beyond the capabilities of phone-based representations. The ability to manipulate and control speech using VQ tokens would open up a plethora of possible future applications, including accent modulation, targeted pronunciation feedback, voice actor performance post-production, and pronunciation control for TTS.

The main contribution of this work is measuring the extent to which VQ token sequences can be manipulated. It is desirable that the tokens correspond to speech in a monotonic and predictable manner. However, because the tokens are always learned as a temporal sequence from natural speech, they have an as-yet-unknown *temporal sensitivity*: they are not guaranteed to have a monotonic relationship to the acoustics, and they might be context- or even speaker-dependent. We offer what we believe to be the first demonstration that it is feasible to manipulate and exchange token sequences. We analyse increasingly challenging tasks, from copy synthesis to the production of novel speech by concatenating short phrases with matched vs. mismatched phonetic context and speaker identity

## 2. Related Work

In the work of [8] they compared how graphemes and phones affected the learned pronunciations in Tacotron sequence-to-sequence TTS. They found that the internal representations for graphemes and phones were consistent, suggesting it is possible to control pronunciations directly from graphemes. They also evaluated the representations externally to Tacotron. However, graphemes are a large unit and the corresponding neural embeddings may not be able to control nuances of pronunciation. In our work, we are modelling a smaller unit with VQ tokens which can provide much finer control for pronunciation over grapheme embeddings.

A growing area of interest involves methods to discover meaningful acoustic units from speech, often in an unsupervised manner, and then utilise them in downstream tasks. In [9], they explored Transformer VQ-VAE for zero-shot synthesis: generating speech without text or phone labels. They showed that the VQ-VAE architecture is well-suited to discover phone and sub-phone units and it is entirely self-supervised. High-quality speech can be synthesised directly from these small units. While this work is encouraging, they have not manipulated the sub-phone units directly, which is something that we explore.

Perhaps one of the best examples of how VQ tokens can be

applied to speech applications comes from the text-to-speech system called DiscreTalk [6]. In this work, the VQ tokens were generated with different down-sampling factors, which effectively controlled the duration of a single VQ token. If the down-sampling factor is large, then individual tokens in a sequence have a longer duration, and vice-versa. They trained a neural machine translation (NMT) system to predict a sequence of VQ tokens from text input, which is similar to grapheme-to-phoneme prediction in conventional TTS systems. The predicted VQ tokens were then used in TTS. They showed that tokens of longer duration facilitated learning TTS, but sometimes by sacrificing overall speech quality. While this finding is important, it is not clear how the size of the token affects more nuanced elements of speech such as co-articulation, or what other aspects of pronunciation could be optimized. Furthermore the VQ tokens predicted from text by their NMT model (and subsequently its resulting pronunciations) are fundamentally both unpredictable and uncontrollable. In this work we make the crucial first steps towards ascertaining whether VQ tokens specifically and neural speech representations more generally are a good candidate for controlling synthesised speech.

## 3. Data and Model

### 3.1. Data

We use VCTK [10] to both train our VQ-VAE model and generate samples for our listening test. Although it is a relatively small dataset (44 hours over 109 speakers), since the recordings are of high quality, and our WaveRNN decoder [11] is of sufficient model capacity, our resulting system is able to accurately generate speech for each of the VCTK speakers. VCTK contains voices from a variety of different ages and UK accents.

### 3.2. VQ-VAE Model

To discover discrete units into which speech can be encoded, then subsequently synthesised, we use a VQ-VAE model based on [2]. Our model differs from the original by using WaveRNN[1] as the vocoder instead of WaveNet [12], for faster training and inference.

The VQ-VAE encoder uses 10 1D convolutional layers to encode a sequence of waveform samples $\mathbf{x}_{1:T}$ into a sequence of 128-dim continuous latents $\mathbf{z}_{1:U}$. These latents are then discretised using a vector quantisation layer to create a sequence of code-words (i.e., codebook entries) $\mathbf{d}_{1:U}$, which henceforth we will call **tokens**. Our codebook contains 512 128-dim entries. We do not examine the effect of varying the codebook size, leaving it for future work. A WaveRNN decoder produces waveform samples at 22.05 kHz sample rate, conditioned on the sequence of tokens and a single speaker one-hot vector that is broadcast across all timesteps $\mathbf{s}_{1:U}$.

We train our model using all VCTK speakers for 1000 epochs (2.7 million iterations), which takes approximately 1 week on a single NVIDIA 2080Ti GPU. Since the focus of this study is to explore concatenative synthesis and *not* to examine VQ-VAE's ability to generalise to unseen speakers, we choose to train the model on all of VCTK. Thus we perform concatenative VQ-VAE synthesis from parts of training utterances, just as would be the case in waveform-domain concatenation [13].

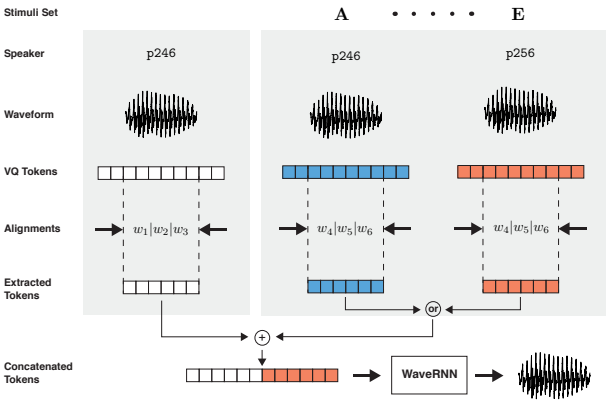


Figure 1: *Overview of the concatenative VQ-VAE synthesis method. The token sequence corresponding to word sequence $w_1, w_2, w_3$ is being concatenated with the token sequence corresponding to $w_4, w_5, w_6$, from which a waveform is synthesised using WaveRNN. Pictured is a set* **E** *stimuli being generated for the* `p246+p256` *speaker combination. The particular sub-sequence of words $w_4, w_5, w_6$ used depends on the stimuli set currently being generated, either* **A**, **B**, **C**, **D**, *or* **E**.

## 4. Method

We describe our method for re-arranging the VQ token subsequences that correspond to 3-word chunks. What is most important is that we find a way to meaningfully manipulate speech in the VQ token domain. Here, we work at the word-level because it is a first approximation to being able to manipulate VQ tokens at a lower level such as below the phone-level. As we have described, a successful outcome at the word-level signifies that VQ tokens may be a feasible representation for controlling pronunciation.

Figure 1[2] shows an overview of the concatenative VQ-VAE synthesis pipeline that extracts tokens corresponding to words in two separate utterances, these tokens are then used to generate a single listening test stimuli. The extraction process for one utterance works as follows: we first encode the audio of one utterance $U_1$ and encode it into a stream of tokens using the VQ-VAE encoder. We then use the timestamps for a sub-sequence of words (e.g. $w_1, w_2, w_3$) to identify and extract their corresponding subset of VQ-VAE tokens $\mathbf{d}_{left}$. We then repeat this process for a second utterance $U_2$ to get another subset of tokens $\mathbf{d}_{right}$ corresponding to $w_4, w_5, w_6$. We then concatenate them together to get $\mathbf{d}_{left} \bigoplus \mathbf{d}_{right}$, and use them to condition the WaveRNN to generate speech that resembles $w_1, w_2, w_3 \bigoplus w_4, w_5, w_6$. Note that we use '$\bigoplus$' to signify the concatenation point between sequences of words or tokens.

Word-level alignments are required to find the correspondence between tokens and words, and are found using the Montreal forced aligner [14]. We do not perform any analysis or adjustment of alignments here, so they are a potential source of error.

We also found that the generation of words corresponding to the start of $\mathbf{d}_{left}$ and the end of $\mathbf{d}_{right}$ were sometimes cut-off or not realised. Subsequently we experimented with inserting $\mathbf{d}_{sil}$ before $\mathbf{d}_{left}$ and after $\mathbf{d}_{right}$, finding that it helps recover some words. Examples can be found on our sample

[1] https://github.com/mkotha/WaveRNN

[2] Credit to Christine Wan for help with this diagram.

page[3]. When generating our listening test stimuli we pad by 50 timesteps of $\mathbf{d}_{sil}$ on both sides. We also chose to use 6-word stimuli rather than 4-word ones so that the words adjacent to those at the concatenation point are not cut off.

# 5. Experiments

## 5.1. Word Transcription Task

We used a fill-in-the-blank transcription task. We generated 6-word stimuli of the form $w_1, w_2, w_3 \bigoplus w_4, w_5, w_6$. For each stimulus, participants were asked to transcribe the word *after* the concatenation point (i.e., $w_4$) by being presented with the transcription *minus* the word-in-question. For example, for the sentence 'red and green $\bigoplus$ looking any further' using $\mathbf{d}_{left}$ corresponding to 'red and green' and $\mathbf{d}_{right}$ to 'looking any further', participants were presented with the transcription 'red and green <blank> any further'.

## 5.2. Experimental Conditions

Our listening test contains 5 sets each with 40 stimuli, so that each participant rates the same 200 stimuli:

- **A**: Copy-synthesis
- **B**: Matched context + Matched Speaker
- **C**: Matched context + Mismatched Speaker
- **D**: Mismatched context + Matched Speaker
- **E**: Mismatched context + Mismatched Speaker

These sets are designed to help us answer the following three questions: Does concatenative synthesis result in less intelligible speech than copy-synthesis (Set **A** vs. Sets **B**, **C**, **D**, **E**)? Is intelligibility affected by extracting tokens from audio spoken by two different speakers (Sets **B**, **D** vs. Sets **C**, **E**)? Is intelligibility affected when the two words adjacent to the concatenation boundary, $w_3$ and $w_4$, are chosen so that their linguistic contexts, according to their surrounding words in their original sentences, mismatch (Sets **B**, **C** vs. Sets **D**, **E**)?

We determine whether linguistic context matches between two words by comparing their adjacent triphones. For example if $w_3$ is 'hello' (HH EH L OW) and $w_4$ is 'world' (W ER L D), then we compare the rightmost triphone of 'hello' to the leftmost one of 'world'. Given that the rightmost triphone of 'hello' is L OW W, if the leftmost one of 'world' is OW W ER then we consider it a matching linguistic context, and if it were AH W ER then we consider it mismatching. When choosing a $w_4, w_5, w_6$ for a given $w_1, w_2, w_3$ we make sure not to choose those which contain a $w_4$ that is either a stopword or a word that has been generated before for a particular speaker.

## 5.3. Materials

We used 40 unique sequences $w_1, w_2, w_3$ each of which could be followed by one of 5 unique sequences of words $w_4, w_5, w_6$ (thereby creating 200 unique sentences in total, described in 5.2). Each unique sequence $w_1, w_2, w_3$ was presented 5 times during the listening test (once per stimuli set), and each time is coupled with a unique $w_4, w_5, w_6$, making a total of 200 sentences, noting that these may not all be grammatically-correct.

[3] https://jonojace.github.io/SSW21-concatenative-vqvae

Table 1: *Speaking rate information (average number of seconds per phone).*

| Duration Type | p246 | p256 | p345 | p374 |
|---|---|---|---|---|
| non-sil Phone | 0.088 | 0.085 | 0.083 | 0.118 |
| sil Phone | 0.114 | 0.106 | 0.278 | 0.278 |

## 5.4. Speakers

We took care in choosing the speakers to build our stimuli. We found in preliminary experiments that conditioning the WaveRNN using tokens extracted from slower-speaking voices resulted in more intelligible speech, when performing either copy-synthesis or concatenative reconstructions. Subsequently we chose 4 slow-speaking voice talents for our experiments; p246, p256, p345, and p374 whose speaking rates are summarised in Table 1.

We generate 50 stimuli from each of our 4 speakers, made up of 10 stimuli from each stimuli set. For example, if the main speaker is p246 and the secondary speaker is p256 (for answering mismatched speakers question) then we will generate 10 stimuli for **A** **B** and **D** using tokens only from p246, and 10 stimuli each for **C** and **E** using $\mathbf{d}_{left}$ from p246 and $\mathbf{d}_{right}$ from p256. The 4 main and secondary speaker combinations that we use are p246+p256, p256+p345, p345+p374, and p374+p246. To condition the WaveRNN we always use the main speaker to condition $\mathbf{d}_{left} \bigoplus \mathbf{d}_{right}$ even if the original speaker of $\mathbf{d}_{right}$ is the secondary speaker, therefore our model performs voice conversion on the latter halves of the stimuli in sets **C** and **E**.

## 5.5. Listening Test

We built our listening test on the Qualtrics[4] platform [5], and recruited participants using Prolific[6]. Using the following filters we recruited 50 participants, each of whom are from the UK, have no literacy difficulties, and have at least a 90% approval rating on Prolific. The order of the 200 stimuli are randomised on a per participant basis.

In order to ensure that our results were accurate we performed a manual check of all participants' answers to correct misspellings (e.g. contenders and contendors), typos (e.g. fresh and frsh), and homophone ambiguity errors (e.g. rode, rowed, and road).

# 6. Results

## 6.1. Stimuli Set Comparisons

We present results from our intelligibility test partitioned across each stimuli set in Figure 2. We find that the copy-synthesis stimuli (**A**) are the most intelligible. This is likely because no concatenative synthesis is performed, and as such the resulting token sequences will not suffer from potential alignment errors and will appear 'natural' to the WaveRNN.

The results of **B** and **C** show that concatenative VQ-VAE synthesis can produce intelligible speech reliably, additionally since they outperform **D** and **E** we can conclude that concatenative synthesis works better when linguistic contexts match.

[4] https://www.qualtrics.com/uk/

[5] Test building automated using https://github.com/CSTR-Edinburgh/qualtreats

[6] https://www.prolific.co/

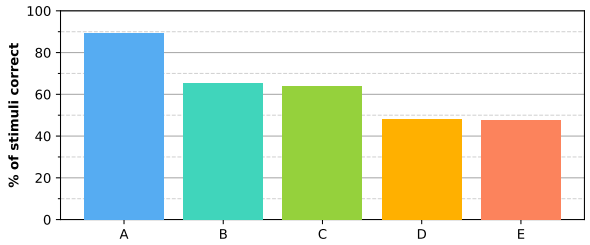


Figure 2: *Intelligibility results across the stimuli sets* **A**, **B**, **C**, **D**, **E** *described in Subsection 5.2. We present the percentage of stimuli within a set answered correctly by participants.*

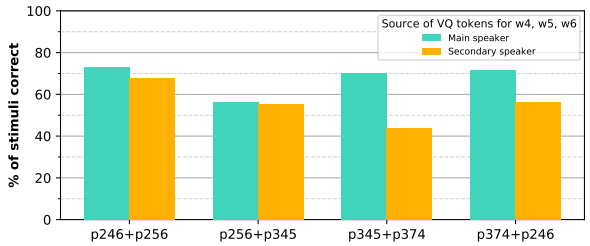


Figure 3: *Intelligibility results for the 4 speaker combinations described in Subsection 5.1. For each speaker combination we present the percentage of stimuli correct when both* $\mathbf{d}_{left}$ *and* $\mathbf{d}_{right}$ *are extracted from the main speaker's speech (left-side teal coloured bars) and when* $\mathbf{d}_{left}$ *is extracted from the main speaker and* $\mathbf{d}_{right}$ *is extracted from the secondary speaker (right-side orange coloured bars). Note that due to the copy-synthesis set* **A**, *the total number of stimuli when speakers are matched and when speakers are mismatched differ, being 120 and 80 respectively.*

Comparing the results of **B** vs **C** and **D** vs **E** we find that synthesising using concatenated token sub-sequences extracted from different speakers has only a small negative on intelligibility. The closeness of these results combined with our observations that the speaker identity of samples do not change mid sentence are testament to the ability of VQ-VAE to learn codebook embeddings that are disentangled from speaker identity. Disentanglement is achieved due to the use of speaker inputs to the decoder and the extreme bottle-necking of the input signal in both the time and feature dimensions (resulting in a low bit-rate encoding). These results are promising for future concatenative neural synthesis work: it is clearly possible to mix and match VQ tokens between different speakers. This could enable new applications such as correcting a system's pronunciations via cheap-to-obtain speech exemplars rather than more expensive phonetic transcriptions.

### 6.2. Speaker Combination Experiments

Figure 3 shows our intelligibility results partitioned across the 4 speakers. We observe three findings: First, that mismatched speakers across the concatenation boundary results in lower intelligibility in general. Second, that mismatched vs matched speaker intelligibility differs between speaker combinations, e.g. the difference for `p256+p345` is very small (1.9% increase), whereas the difference is larger for `p345+p374` (59% increase). Upon reflection of the speaking rates in Table 1 we do not find a strong correlation between speaking rate and intelligibility, subsequently differences in intelligibility could be due to other factors such as per speaker co-articulation habits or accent differences for example. Third, we find that certain speakers are less intelligible in general, speaker `p256` for example. We further investigated this by visualising the total duration of silences for each speaker in Figure 4 where we see that speaker `p256` has the smallest amount of total silence, which may make interword boundaries harder to identify for both the VQ-VAE encoder and forced aligner. Additionally the speaker's heavier more informal accent may also have an effect on intelligibility.

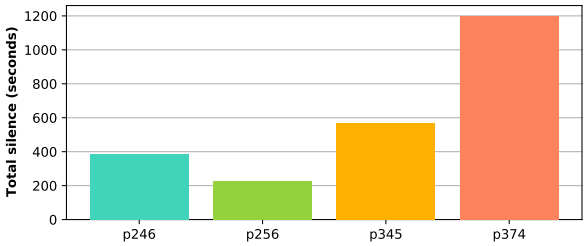


Figure 4: *Total duration of all silence phones for each speaker.*

## 7. Qualitative Analysis

In this section we present our findings regarding the types of errors that participants made. We are particularly interested in the cases of incorrect transcription because it helps us better understand how well we can manipulate VQ tokens at a finer-granularity. One of our hypotheses was that concatenating $\mathbf{d}_{left}$ and $\mathbf{d}_{right}$ together would mainly cause the sounds adjacent to the boundary to be affected. Surprisingly however we found that there are instances where the first phone of $w_4$ was correctly heard, but the rest of the word was largely intelligible, causing participants to hallucinate an incorrect answer. Since we did not discover a cause for this phenomena we include examples on our samples page and leave further investigation to future work.

## 8. Conclusion

In this work we present an analysis of the sensitivity of VQ-VAE tokens to their surrounding context by using concatenations of tokens extracted from disparate sentences to decode audio. We primarily find that 'unit selection' speech generation is possible in the discrete latent space. Furthermore by extracting tokens from sentences selected from a variety of specific conditions we discover that VQ-VAE tokens are temporally highly linguistic context dependent, but not speaker context dependent. Together these two results are promising for future speech systems as they suggest that readily available audio exemplars can be used to alter aspects of speech such as pronunciation without resorting to expensive hand-transcribed labels such as phonetic transcriptions. We further observe that within our pipeline speaking rate and duration of silences can affect downstream reconstruction intelligibility. In future work we plan to investigate neural concatenative synthesis cross-lingually, make tokens less context dependent without sacrificing reconstruction quality, and remove the reliance of our system on pretrained forced aligners and instead use word-level alignments obtained in an unsupervised fashion.

## 9. Acknowledgements

This work was partially supported by the EPSRC Centre for Doctoral Training in Data Science, funded by the UK Engineering and Physical Sciences Research Council (grant EP/L016427/1) and University of Edinburgh.

## 10. References

[1] W.-N. Hsu, Y. Zhang, and J. Glass, "Unsupervised learning of disentangled and interpretable representations from sequential data," *arXiv preprint arXiv:1709.07902*, 2017.

[2] A. v. d. Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," *arXiv preprint arXiv:1711.00937*, 2017.

[3] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[4] Y.-A. Chung and J. Glass, "Generative pre-training for speech with autoregressive predictive coding," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 3497–3501.

[5] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *arXiv preprint arXiv:2006.11477*, 2020.

[6] T. Hayashi and S. Watanabe, "Discretalk: Text-to-speech as a machine translation problem," *arXiv preprint arXiv:2005.05525*, 2020.

[7] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. Mohamed, and E. Dupoux, "Speech resynthesis from discrete disentangled self-supervised representations," *arXiv preprint arXiv:2104.00355*, 2021.

[8] A. Perquin, E. Cooper, and J. Yamagishi, "An investigation of the relation between grapheme embeddings and pronunciation for tacotron-based systems," 2021.

[9] A. Tjandra, S. Sakti, and S. Nakamura, "Transformer vq-vae for unsupervised unit discovery and speech synthesis: Zerospeech 2020 challenge," 2020.

[10] J. Yamagishi, C. Veaux, K. MacDonald *et al.*, "CSTR VCTK Corpus: English Multi-Speaker Corpus for CSTR Voice Cloning Toolkit (version 0.92)," 2019.

[11] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2410–2419.

[12] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[13] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, vol. 1. IEEE, 1996, pp. 373–376.

[14] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal forced aligner: Trainable text-speech alignment using kaldi." in *Interspeech*, vol. 2017, 2017, pp. 498–502.