Interspeech 2024
1-5 September 2024, Kos, Greece

# Learning Pronunciation from Other Accents via Pronunciation Knowledge Transfer

*Siqi Sun, Korin Richmond*

The University of Edinburgh, Edinburgh, UK
Siqi.Sun@ed.ac.uk, Korin.Richmond@ed.ac.uk

## Abstract

Bootstrapping has proven to be effective in transforming a conventional pipeline-based linguistic frontend to an integrated Sequence-to-Sequence (Seq2Seq) frontend for text-to-speech (TTS). However, for target accents with limited lexical coverage, the performance of bootstrapped Seq2Seq frontends would be greatly limited. In this work, we utilize multi-accent bootstrapping for rich-resource source accents and low-resource target accents to enable pronunciation knowledge transfer between them, effectively enlarging the lexical coverage of target accent. We formally analyze the effect of transfer between 3 English accents (word accuracy increase of 12%–17% absolute for transferred words) and how it scales with the number of annotated unique word types in the target accent. When annotating as few as 1k word types for the target accent, the transfer achieves a word accuracy of 81% for transferred words, approaching the generalisation ability of a baseline annotating 51k word types.
**Index Terms**: knowledge transfer, pronunciation learning, linguistic frontend, multi-accent, text-to-speech synthesis

## 1. Introduction

For many languages (e.g., English), predicting the pronunciation of unseen words is non-trivial, as these languages have irregular letter-to-sound mappings. For such languages, comprehensive lexical coverage is important for ensuring the pronunciation accuracy of a text-to-speech (TTS) synthesis system. Previous end-to-end (E2E) neural TTS systems [1, 2] have limited lexical coverage, as the current size of their training corpora of ⟨text, speech audio⟩ pairs [3, 4, 5] is two orders of magnitude smaller than a size that can achieve a comparable lexical coverage contained in a dictionary [6]. Recent work instead opts to use a separate linguistic frontend to generate the pronunciation sequence (mainly phonetic tokens) to serve as the input for TTS [7, 8, 9, 10], achieving better pronunciation accuracy [11].

The conventional pipeline-based linguistic frontend (including its built-in dictionaries) is difficult to build and scale to new accents or languages. More recent work has shown the possibility and benefits of replacing the pipeline with an integrated sentence-level sequence-to-sequence (Seq2Seq) neural network [12, 13, 14, 15], which only requires a large parallel corpus of ⟨text, pronunciation⟩ pairs for training. To overcome the lack of pronunciation training target, a natural choice is to take full advantage of the pre-existing pipeline-based linguistic frontend (which is the case for many languages) and apply a bootstrapping approach, in which a large amount of unlabelled text is run through the pipeline-based frontend to generate pronunciation sequences which in turn serve as the training target [12, 15].

A pipeline-based frontend typically relies on built-in dictionaries to look up the word pronunciation, and usually each accent is equipped with a distinct dictionary. For example, the long-standing English pipeline-based frontend Festival [16] can be equipped with UNISYN [17], which is a set of dictionaries for more than 10 English accents, including General American (GAM), standard British (RPX), Edinburgh (EDI) and so on. It happens sometimes (but not for UNISYN though) that one accent has a larger lexical coverage (i.e., a dictionary with more entries) and another accent has a more limited coverage (i.e., a dictionary with fewer entries). If the bootstrapping approach is applied to each accent separately, the performance of Seq2Seq frontend corresponding to the accent with limited lexical coverage would be greatly upper-bounded [15]. On the other hand, trivially expanding its dictionary requires significant linguistic knowledge and effort, which is usually unaffordable. Note in practice, the unlabelled text needs to cover the entire dictionary (i.e., each word in the dictionary appears at least once in the unlabelled text) for maximal lexical coverage.

For accents with limited lexical coverage (low-resource target accents), we aim to overcome the performance upper limit imposed by bootstrapping without manually expanding their dictionaries. In this work, we propose a straightforward solution to the above problem by applying the bootstrapping approach to multiple accents at once. In multi-accent bootstrapping, the pipeline-based frontend equipped with multiple accent dictionaries (possibly of various dictionary sizes) generates pronunciation sequences for each accent and its corresponding (and possibly non-overlapping) unlabelled text, which are as a whole used to bootstrap a multi-accent sentence-level Seq2Seq model.

In the resulting multi-accent Seq2Seq frontend, pronunciation knowledge can in principle be transferred between accents, effectively enlarging the lexical coverage of target accents. Specifically, the pronunciation knowledge of a word exclusively present in a rich-resource source accent dictionary can be transferred to the low-resource target accents, with the pronunciations of that word being absent in the original target accents' dictionaries. This effect is intuitive: in addition to the orthographic pattern (i.e., the word's character sequence), its pronunciation in one accent also serves as one important clue for predicting its pronunciation in another accent. For example, if we already know the general pronunciation rules for RPX and EDI in advance, then when we know the correct pronunciation of 'deserved' in RPX is '0 d i - 1 z @@r v d', it is not too difficult to infer its EDI pronunciation is '0 d i - 1 z er r v d'.

The contributions of this work are two-fold. Through experiments, we show: 1) the effect of pronunciation knowledge transfer; and 2) how the pronunciation knowledge transfer scales with the number of annotated word types[1] in the tar-

[1]Following standard terminology, a 'word token' is an individual occurrence of a distinct 'word type' in the text.


10.21437/Interspeech.2024-66

get accents (or more precisely the number of word types which are annotated in both the source accent and the target accent), which is important for the pronunciation modelling of those low-resource and underrepresented accents.

In order to analyze the effect of pronunciation knowledge transfer between accents without the interference caused by non-standard words (e.g., abbreviations, numbers, etc.), we do not include text normalisation (TN) in our Seq2Seq frontend modelling, as is done in [15].

## 2. Related work

Bootstrapping is widely adopted in the TTS field to facilitate the construction of neural network-based systems, at the same time as taking full advantage of previous pipeline-based systems. In [18, 19], a large set of unnormalised text is run through Google's Kestrel TN component to produce normalised text for training Seq2Seq TN models. In [12], text is run through a working production frontend to produce phone sequence target for training multilingual Seq2Seq TTS frontends. In [15], normalised text of LibriSpeech [3] is run through Festival [16] to produce pronunciation sequence for training a Seq2Seq TTS frontend.

Multilingual/Multi-accent pronunciation modelling has received much attention in recent years [12, 20, 14, 21, 22, 23, 24, 25]. Such work models all the languages/accents in a shared Seq2Seq model, which leads to a compact model capable of utilizing shared knowledge between different language/accent pronunciation systems, which can particularly benefit low-resource languages/accents with limited training data [21, 24]. The problem setting and model architecture in this work are similar to the aforementioned studies. However, this work differs in that we systematically analyze the effect of pronunciation knowledge transfer between accents, which to the best of our knowledge has never been deeply investigated before. This analysis is important, as it can serve as a guideline for helping the pronunciation modelling for those underrepresented accents with limited lexical coverage.

## 3. Model architecture

The multi-accent Seq2Seq frontend in this work follows a typical RNN-based Encoder-Attention-Decoder architecture [15], aiming to directly convert an input text sequence $\boldsymbol{x}_{1:S} = [x_1, x_2, \ldots, x_S]$ (i.e., a string of characters) into the pronunciation sequence $\boldsymbol{y}_{1:T} = [y_1, y_2, \ldots, y_T]$ (i.e., a string of phones, mixed with lexical stress markers, syllable boundaries, prosodic boundaries and so on), where $S$ and $T$ are the lengths of the text sequence and the pronunciation sequence, respectively. Here, the text encoder and the pronunciation decoder are shared by all the accents.

The input text sequence $\boldsymbol{x}_{1:S}$ is first encoded by a bi-directional LSTM encoder into a sequence of hidden vectors, which is concatenated step-wise with an accent embedding $\boldsymbol{l}$ looked up in an accent embedding table by the accent label $l$, to produce the final hidden vectors $\boldsymbol{h}_{1:S}$. The attention mechanism and decoder then transform $\boldsymbol{h}_{1:S}$ to the pronunciation sequence $\boldsymbol{y}_{1:T}$. We refer the reader to [15] for the model and representation detail. We adopt the monotonic GMM attention (V2) [26] to exploit the property of the task that the alignments between the characters and the pronunciation tokens are largely monotonic. The Seq2Seq model is trained using maximum likelihood estimation in a teacher-forcing manner. During inference, beam search is carried out to decode the best pronunciation sequence.

## 4. Experiments

### 4.1. Setup and dataset

In this work, we opt to model three different English accents as an example, which are EDI, GAM and RPX[2]. Correspondingly, `unilex-edi`, `unilex-gam` and `unilex-rpx` [17] are used as the dictionaries and define the phone sets for the pronunciation output. Festival [16] is used as the pipeline-based frontend. The ⟨unlabelled text, accent⟩ pairs are run through the pipeline to generate the corresponding pronunciation sequence. Note that the sets of unlabelled text corresponding to each accent can either overlap or not, depending on the experiment conducted below. We follow the pre-processing steps in [15].

For the experiments conducted in this work, the unlabelled text does not cover the entire dictionary, and therefore the actual lexical coverage for an accent is limited to the in-dictionary words covered[3] by the corresponding text. To adjust the lexical coverage, we can easily vary the number of sentences included in the training text, as is done in Section 5.3. To evaluate the pronunciation knowledge transfer for the target accent(s), we just need to evaluate on those (in-dictionary) *transferred words* that are uncovered[3] by the training text of the target accent(s) but covered by that of the source accent(s).

The normalised transcriptions for three training subsets of LibriSpeech [3] are used to form the training text. Only those sentences which do not contain out-of-dictionary words are kept, resulting in 206k sentences (51,429 word types). The normalised text of Dev-clean [3] is used to form the validation text, resulting in 2,703 sentences. The normalised text of Hi-Fi TTS [5] is used to form the test set (named Hifi-4k in this work). Only those sentences containing in-dictionary words uncovered by the training text are kept, resulting in 4,413 sentences (49.2k covered word tokens, 2,424 uncovered word types and 4,547 uncovered word tokens). To demonstrate the effect of pronunciation knowledge transfer, Hifi-4k is also used as the augmentation dataset to enlarge the lexical coverage of source accent(s).

### 4.2. Model configurations

Four sets of model configurations are evaluated in this work:

1. `Uni-base`: 3 uni-accent frontends (EDI, GAM and RPX).
2. `Multi-base`: a single multi-accent frontend, modelling EDI, GAM and RPX jointly, as shown in Figure 1(a).
3. `Multi-aug-$source$`: 6 multi-accent frontends as above, each augmented with Hifi-4k for one or two source accents during training (EDI, GAM, RPX, EDI+GAM, EDI+RPX and GAM+RPX), as exemplified in Figure 1(b). Note the training text (and hence the lexical coverage) of LibriSpeech corresponding to each accent completely overlap.
4. `Multi-disjoint-aug-$source$`: augmented multi-accent frontends as above, but the target accent has no overlap with the source accent(s) in LibriSpeech training text, as exemplified in Figure 1(c) (Here EDI is the target accent). To control the amount of annotated word types in the target accent, we sample N sentences from the first half of LibriSpeech text for the target accent. The second half is kept fixed for the source accent(s).

All the models use the same architecture presented in Section 3. The encoder and decoder each have 2 layers. The en-

[2]The codebase and datasets are made available at `https://github.com/sunsiqitos/multi_accent_s2s_frontend`

[3]Here, 'covered/uncovered' is interchangeable with 'seen/unseen' in meaning, respectively.


(a) Multi-base (b) Multi-aug-RPX (c) Multi-disjoint-aug-RPX

Figure 1: *Model configurations evaluated in this work. LS stands for LibriSpeech. Shaded area indicates the part of training text used in the configuration. (c) depicts the case when EDI is used as the target accent.*

coder, decoder and accent embedding dimensions are set to 256, 256 and 32, respectively. The hidden dimension is set to 384 for `Uni-base` and 512 for multi-accent models. Dropout rate is set to 0.3. We use 5 mixture components for GMM attention. The learning rate is set to 5e-5. Adam optimizer is used, where $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The batch size is set to 40. The number of parameters of `Uni-base` and the multi-accent models are 5.1M and 8.7M, respectively. We trained each model for 4 days on one GeForce RTX 2080Ti card. The beam size is 30.

# 5. Results

## 5.1. Multi-accent baseline versus uni-accent baselines

To quantify how well multi-accent bootstrapping works, we first compare `Multi-base` with 3 `Uni-base` models. Following [15], we evaluate them in three respects: sentence-level performance; word-level performance for covered words (i.e., memorisation); and word-level performance for uncovered words (i.e., generalisation). For sentence-level evaluation, we compute:

- AlignErr: the number of predicted sequences with at least one alignment error, so measuring the alignment robustness.
- PBER: prosodic boundary error rate between the predicted sequence and the pipeline's prediction.

For the word-level performance, we evaluate the segmented word-level prediction against the pipeline's prediction. We limit the evaluation only to in-dictionary words, for which we have the ground-truth pronunciation. All the metrics are evaluated on word tokens. The following metrics are computed:

- WAcc: word accuracy considering phones, stresses and syllable boundaries.
- WAccP: word accuracy considering phones only.
- PER: phone error rate.

As shown in Table 1, all four models achieve impressive results, demonstrating the effectiveness of multi-accent bootstrapping. For the sentence-level result, the models achieve a similarly high alignment robustness (e.g., AlignErr $<= 0.04\%$). For covered words, all models achieve a similarly high word accuracy and low PER, matching those in [15]. For uncovered words, `Multi-base` is on a par with or significantly better ($p < 0.05$) than `Uni-base` models, except for WAccP for GAM. Overall, `Multi-base` is on a par with or significantly better than `Uni-base`.

## 5.2. Pronunciation knowledge transfer between accents

To investigate the effect of pronunciation knowledge transfer, we compare `Multi-base` with `Multi-aug-$source$`. We use EDI as the target accent and focus on WAcc on *transferred words*. Therefore, Hifi-4k (EDI) is used for evaluation, while Hifi-4k ($source$) is used as the augmentation dataset to enlarge the lexical coverage of the source accent(s). The results when using GAM or RPX as the target accent are very similar, but are not shown to save space.

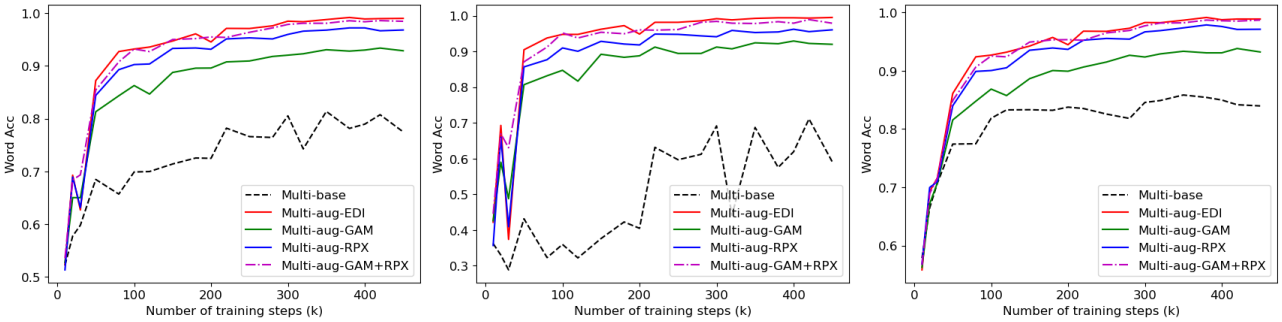
Table 1: *`Multi-base` vs. 3 `Uni-base`s. The number in bold indicates it is not significantly different from the best value in that cell (i.e., $p > 0.05$ in two-sample proportions z-test).*

| | EDI | GAM | RPX |
|---|---|---|---|
| | Uni / Multi | Uni / Multi | Uni / Multi |
| **Sentence level** | | (4,413 sentences, 53.7k word tokens) | |
| AlignErr | **2 / 1** | **2 / 2** | **1 / 2** |
| PBER (%) | **2.69 / 2.56** | **2.68 / 2.54** | 3.00 / **2.55** |
| **Word level (covered)** | | (49.2k word tokens, 182k phones) | |
| WAcc (%) | **99.96 / 99.95** | **99.94 / 99.94** | **99.95 / 99.96** |
| WAccP (%) | **99.96 / 99.96** | **99.95 / 99.96** | **99.97 / 99.96** |
| PER (%) | **0.014 / 0.016** | **0.019 / 0.018** | **0.014 / 0.014** |
| **Word level (uncovered)** | | (4.5k word tokens, 33.8k phones) | |
| WAcc (%) | 77.1 / **78.9** | **76.6 / 75.0** | 76.5 / **79.9** |
| WAccP (%) | 81.5 / **83.7** | **81.1** / 79.3 | 81.5 / **84.8** |
| PER (%) | 3.59 / **3.12** | **4.24 / 4.19** | 3.72 / **2.97** |

When using EDI as the target accent, we expect `Multi-base` to give the performance lower bound, while `Multi-aug-EDI` to give the performance upper bound, as indeed it is evaluated on the covered words. More importantly, we investigate whether being augmented with other source accent(s) can close the gap between `Multi-base` and the full-supervision model (i.e., `Multi-aug-EDI`).

The results are shown in Figure 2(a), which plots WAcc against the number of training step. As expected, `Multi-base` gives the lower bound (81.4%) and `Multi-aug-EDI` gives the upper bound (99.0%). More importantly, `Multi-aug-GAM`, `Multi-aug-RPX` and `Multi-aug-GAM+RPX` achieve 93.4%, 97.2% and 98.6%, respectively. Note for an uncovered word in Hifi-4k, we find the probability of its pronunciations being identical in the source accent and the target accent EDI is at most one-third (30.2% for GAM/EDI, and 25.2% for RPX/EDI). In other words, at most one-third of the uncovered words could have been guessed correctly simply due to their pronunciations being identical in the source and target accents, which could have possibly increased WAcc of `Multi-base` from 81.4% to 87.6%. However, the improvements in our experiments surpass this by a large margin, confirming the pronunciation knowledge transfer between accents. Here, word pronunciation knowledge in source accent(s) (GAM and/or RPX) is transferred to the target accent EDI.

Note in Figure 2(a), WAcc of `Multi-base` is noisy during training. We divide the words into two groups according to




(a) all words (b) NNP words (c) Non-NNP words

Figure 2: *Word accuracy w.r.t. the training step for uncovered/transferred words (target accent: EDI).*

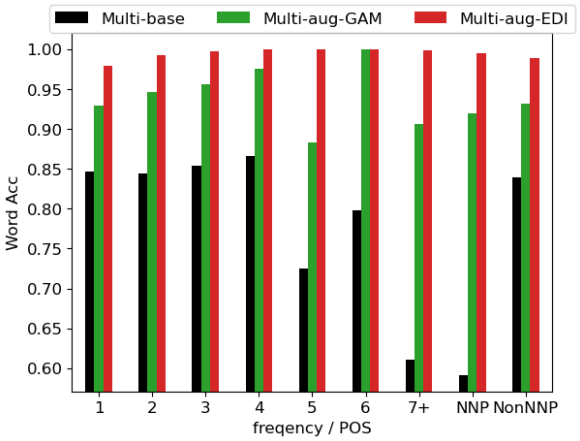



Figure 3: *Word accuracy w.r.t. word frequency and POS for uncovered/transferred words (target accent: EDI).*

Table 2: *The scaling of pronunciation knowledge transfer (target accent: EDI). The number in bold indicates it is not significantly different ($p > 0.05$) from the value achieved by* `Multi-aug-RPX` *(src: source accent, tgt: target accent).*

| Model | # sent. | # word types in tgt | # word types in src & tgt | Transferred WAcc (%) |
|---|---|---|---|---|
| `Multi-disjoint-aug-RPX` | 100 | 1,149 | 1,144 | 80.9 |
| | 200 | 1,607 | 1,597 | 85.7 |
| | 400 | 2,972 | 2,941 | 89.3 |
| | 1k | 5,612 | 5,468 | 92.6 |
| | 2k | 8,396 | 8,122 | 94.0 |
| | 4k | 11,990 | 11,497 | 95.1 |
| | 10k | 18,715 | 17,519 | 95.9 |
| | 20k | 25,409 | 23,015 | **97.2** |
| | 40k | 32,837 | 28,336 | **96.8** |
| | 100k | 43,409 | 34,332 | **97.0** |
| `Multi-aug-RPX` | 206k | 51,429 | 51,429 | **97.2** |

their POS tags, i.e., proper nouns (NNP) and non proper nouns (Non-NNP), and view the results in Figure 2(b) and 2(c) separately. We can see the noise is mainly caused by NNP words, and pronunciation knowledge transfer can greatly smooth it. In other words, the transfer is more valuable for NNP words, which is also shown in the last two columns of Figure 3, where the transfer closes the gap between `Multi-base` and the full-supervision model (i.e., `Multi-aug-EDI`) more significantly for NNP than for Non-NNP. The first 7 columns of Figure 3 shows the transfer largely has a similar behaviour as the full-supervision model in better "memorising" *transferred words* that are covered more frequently in the source training set.

### 5.3. Scaling of pronunciation knowledge transfer

To investigate how pronunciation knowledge transfer scales with the number of annotated word types in the target accent (or more precisely the word types which are annotated in both the source and target accents), we choose EDI as the target accent and RPX as the source accent. To be more realistic, we assume they have no overlap in the training text. Specifically, we evaluate the performance of `Multi-disjoint-aug-RPX` on Hifi-4k (EDI) when varying the number of annotated word types in EDI through sampling different numbers of training sentences. The results are shown in Table 2.

Compared to the best value (97.2%) achieved by `Multi-aug-RPX`, there is no significant difference when the number of annotated word types is reduced from 51k to 25k. Moreover, with as few as 1k annotated word types, pronunciation knowledge transfer achieves a WAcc of 80.9%, which is not significantly different ($p > 0.05$) from 81.4% achieved by `Multi-base`, which requires annotating 51k word types for the target accent. In other words, to predict the pronunciation of some word in a low-resource target accent, we can trade the effort in annotation for collecting its pronunciation in another rich-resource source accent, which is often more trivial.

## 6. Conclusions

To overcome the performance upper limit imposed by bootstrapping for a low-resource target accent, a multi-accent bootstrapping approach is utilized in this work, which enables transferring pronunciation knowledge from rich-resource source accent(s) to the target accent. Through experiments on 3 English accents, we: 1) show the multi-accent bootstrapping achieves a remarkable performance (word accuracy $> 99.9\%$ for memorisation and $> 75\%$ for generalisation); 2) formally analyze the effect of transfer (word accuracy increase of 12%-17% absolute for *transferred words*); and 3) show how the transfer scales with the number of annotated word types in the target accent (word accuracy of 81% for transferred words when annotating only 1k word types for the target accent).

## 7. Acknowledgements

This work was supported in part by the UKRI Centre for Doctoral Training in Natural Language Processing, funded by the UKRI (grant EP/S022481/1), the University of Edinburgh, School of Informatics and School of Philosophy, Psychology & Language Sciences and Huawei.

## 8. References

[1] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. C. Courville, and Y. Bengio, "Char2wav: End-to-end speech synthesis," in *Proc. ICLR*, 2017.

[2] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech 2017*, 2017, pp. 4006–4010.

[3] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "LibriSpeech: An ASR corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[4] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "LibriTTS: A corpus derived from LibriSpeech for text-to-speech," in *Proc. Interspeech 2019*, 2019, pp. 1526–1530.

[5] E. Bakhturina, V. Lavrukhin, B. Ginsburg, and Y. Zhang, "Hi-Fi multi-speaker English TTS dataset," in *Proc. Interspeech 2021*, 2021, pp. 2776–2780.

[6] J. Taylor and K. Richmond, "Analysis of pronunciation learning in end-to-end speech synthesis," in *Proc. Interspeech 2019*, 2019, pp. 2070–2074.

[7] J. Fong, J. Taylor, K. Richmond, and S. King, "A comparison of letters and phones as input to sequence-to-sequence models for speech synthesis," in *Proc. 10th ISCA Speech Synthesis Workshop*, 2019, pp. 223–227. [Online]. Available: http://dx.doi.org/10.21437/SSW.2019-40

[8] T. Hayashi, R. Yamamoto, K. Inoue, T. Yoshimura, S. Watanabe, T. Toda, K. Takeda, Y. Zhang, and X. Tan, "Espnet-TTS: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7654–7658.

[9] J. Shen, Y. Jia, M. Chrzanowski, Y. Zhang, I. Elias, H. Zen, and Y. Wu, "Non-attentive Tacotron: Robust and controllable neural TTS synthesis including unsupervised duration modeling," *ArXiv*, vol. abs/2010.04301, 2020.

[10] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=piLPYqxtWuA

[11] X. Tan, T. Qin, F. K. Soong, and T.-Y. Liu, "A survey on neural speech synthesis," *ArXiv*, vol. abs/2106.15561, 2021.

[12] A. Conkie and A. M. Finch, "Scalable multilingual frontend for TTS," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6684–6688, 2020.

[13] J. Pan, X. Yin, Z. Zhang, S. Liu, Y. Zhang, Z. Ma, and Y. Wang, "A unified sequence-to-sequence front-end model for Mandarin text-to-speech synthesis," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6689–6693.

[14] G. Comini, S. Ribeiro, F. Yang, H. Shim, and J. Lorenzo-Trueba, "Multilingual context-based pronunciation learning for text-to-speech," in *Proc. INTERSPEECH 2023*, 2023, pp. 631–635.

[15] S. Sun, K. Richmond, and H. Tang, "Improving Seq2Seq TTS frontends with transcribed speech audio," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1940–1952, 2023.

[16] R. A. Clark, K. Richmond, and S. King, "Multisyn: Open-domain unit selection for the Festival speech synthesis system," *Speech Communication*, vol. 49, no. 4, pp. 317 – 330, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167639307000398

[17] S. Fitt, "Documentation and user guide to UNISYN lexicon and post-lexical rules," 2000. [Online]. Available: https://www.cstr.ed.ac.uk/projects/unisyn/

[18] R. Sproat and N. Jaitly, "RNN approaches to text normalization: A challenge," *ArXiv*, vol. abs/1611.00068, 2016.

[19] H. Zhang, R. Sproat, A. H. Ng, F. Stahlberg, X. Peng, K. Gorman, and B. Roark, "Neural models of text normalization for speech applications," *Comput. Linguist.*, vol. 45, no. 2, p. 293–337, Jun. 2019. [Online]. Available: https://doi.org/10.1162/coli_a_00349

[20] H.-Y. Kim, J.-H. Kim, and J.-M. Kim, "Fast bilingual grapheme-to-phoneme conversion," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track*. Hybrid: Seattle, Washington + Online: Association for Computational Linguistics, Jul. 2022, pp. 289–296. [Online]. Available: https://aclanthology.org/2022.naacl-industry.32

[21] B. Peters, J. Dehdari, and J. van Genabith, "Massively multilingual neural grapheme-to-phoneme conversion," in *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*. Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 19–26. [Online]. Available: https://aclanthology.org/W17-5403

[22] J. Route, S. Hillis, I. Czeresnia Etinger, H. Zhang, and A. W. Black, "Multimodal, multilingual grapheme-to-phoneme conversion for low-resource languages," in *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 192–201. [Online]. Available: https://aclanthology.org/D19-6121

[23] M. Yu, H. D. Nguyen, A. Sokolov, J. Lepird, K. M. Sathyendra, S. Choudhary, A. Mouchtaris, and S. Kunzmann, "Multilingual grapheme-to-phoneme conversion with byte representation," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8234–8238.

[24] K. Vesik, M. Abdul-Mageed, and M. Silfverberg, "One model to pronounce them all: Multilingual grapheme-to-phoneme conversion with a Transformer ensemble," in *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Online: Association for Computational Linguistics, Jul. 2020, pp. 146–152. [Online]. Available: https://aclanthology.org/2020.sigmorphon-1.16

[25] J. Zhu, C. Zhang, and D. Jurgens, "ByT5 model for massively multilingual grapheme-to-phoneme conversion," in *Proc. Interspeech 2022*, 2022, pp. 446–450.

[26] E. Battenberg, R. Skerry-Ryan, S. Mariooryad, D. Stanton, D. Kao, M. Shannon, and T. Bagby, "Location-relative attention mechanisms for robust long-form speech synthesis," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6194–6198.