Phoneme-Based Contextualization for Cross-Lingual Speech Recognition in End-to-End Models

Ke Hu*, Antoine Bruguier*, Tara N. Sainath, Rohit Prabhavalkar, Golan Pundak

Google, LLC, USA

{huk,tonybruguier,tsainath,prabhavalkar,golan}@google.com

Abstract

Contextual automatic speech recognition, i.e., biasing recognition towards a given context (e.g. user's playlists, or contacts), is challenging in end-to-end (E2E) models. Such models maintain a limited number of candidates during beam-search decoding, and have been found to recognize rare named entities poorly. The problem is exacerbated when biasing towards proper nouns in foreign languages, e.g., geographic location names, which are virtually unseen in training and are thus out-of-vocabulary (OOV). While grapheme or wordpiece E2E models might have a difficult time spelling OOV words, phonemes are more acoustically salient and past work has shown that E2E phoneme models can better predict such words. In this work, we propose an E2E model containing both English wordpieces and phonemes in the modeling space, and perform contextual biasing of foreign words at the phoneme level by mapping pronunciations of foreign words into similar English phonemes. In experimental evaluations, we find that the proposed approach performs 16% better than a grapheme-only biasing model, and 8% better than a wordpiece-only biasing model on a foreign place name recognition task, with only slight degradation on regular English tasks.

Index Terms: End-to-end model, contextual biasing, cross-lingual speech recognition, phoneme model

1. Introduction

End-to-end (E2E) models have attracted increasing attention recently. Instead of building an automatic speech recognition (ASR) system from different components such as the acoustic model (AM), language model (LM), and pronunciation model (PM), E2E models rely on a single neural network to directly learn speech-to-text mapping. Representative systems include a word-based connectionist temporal classification (CTC) model [1], recurrent neural network transducer (RNN-T) [2, 3], and attention-based models such as "Listen, Attend, and Spell" (LAS) [4]. Recent advances have shown that E2E models can outperform the state-of-the-art conventional system when trained on thousands of hours of data [5, 6].

In previous work [7], it has been shown that contextual information (i.e., phrases relevant to recognition in the current context such as contact names, geographic place names, songs, etc.) can improve ASR accuracy. Such phrases are often foreign words, or are rarely seen in training. Recognizing these phrases is challenging. Conventional ASR systems model them as independent contextual LM using an n-gram weighted finite state transducer (WFST), and compose it with a baseline LM for on-the-fly (OTF) rescoring [7, 8]. This idea is extended to a LAS model in [9], where an n-gram LM and a word-to-grapheme "speller" are composed to produce a contextual FST

for rescoring. The approach is similar to shallow fusion [10] which interpolates E2E model scores with an external LM in beam search. To bring more relevant words for biasing in E2E models, [11] proposes to push biasing weights to each subword unit and deals with over-biasing. Further improvements such as biasing before beam pruning, and wordpiece-based biasing have been proposed to achieve state-of-the-art biasing results [12, 6, 13]. Another class of contextual biasing uses an all-neural approach. Contextual-LAS (CLAS) is proposed in [11] to use a bias encoder to model contextual phrases as embeddings and shows significant improvement than OTF rescoring [8]. Phonetic information has been incorporated to CLAS to improve rare word recognition [14].

Although biasing is improved by these techniques, they do not address cross-lingual recognition. In [15], contextual biasing has been used to assist recognition of foreign words. With the phoneme mapping from a foreign language phoneme set to the recognizer's phoneme set, foreign words are modeled as a phoneme-level contextual FST for biasing. It is unclear whether such an approach can be directly applied to E2E models. Phoneme-only E2E systems have been shown to have inferior performance compared to grapheme or wordpiece models (WPM) in general [16, 17], but shows better recognition of rare words and proper nouns.

In this work we propose to incorporate phonemes to a wordpiece E2E model as modeling units and use phoneme-level FST for contextual biasing. We propose a word-frequency based sampling strategy to randomly tokenize rare words into phonemes in the target sequence using a lexicon. This approach also mitigates accuracy regressions that have been observed when using phoneme-only E2E models [16, 17]. We train our model using only American English data and thus its wordpieces and phoneme set (no data from foreign languages). In inference, given a list of foreign words, we bias the recognition using an English phoneme-level biasing FST, which is built by first tokenizing the words into foreign phonemes and then mapping them to English phonemes using [15]. For example, given a navigation query "directions to Créteil" and the assumption that the French word "Créteil" is in our biasing list, "Créteil" is first tokenized to French phonemes as "k R e t E j", and then mapped to English phonemes "k r\ E t E j" for biasing[1]. The phoneme mapping is necessary since our modeling units contain only English phonemes. In decoding, we propose to incorporate the pronunciation FST of the biasing words to consume English phoneme symbols and produce foreign words, using the aforementioned foreign lexicon and phoneme mapping, i.e. "k r\ E t E j" → Créteil (details in Section 3.2 and 3.3). Wordpiece outputs are concatenated to form words. In experimental evaluations, we find that the proposed phoneme-based biasing using wordpiece-phoneme model

---

*These authors contributed equally to this work.

[1] X-SAMPA notations are used for phonemes.

successfully recognizes foreign words. It performs 16% relatively better in terms of WER than the grapheme-only biasing model, and 8% better than the wordpiece-only biasing model in a task of recognizing navigation queries containing French place names. The proposed model also has the advantage that it can be directly applied to other foreign languages for biasing without model scalability issues.

2. Prior Work

2.1. Shallow Fusion E2E Biasing

Shallow fusion has been used in E2E models for decoding [10] and contextual biasing [6]. Biasing phrases are first represented as n-gram WFST in the word level (G), and then left composed with a "speller" FST (S) to produce a contextual LM: $C = min(det(S \circ G))$. The speller transduces a sequence of subword units to corresponding words. The contextual LM (C) is then used to rescore the log-probability outputs of the E2E model during beam search:

$$\boldsymbol{y}^* = \arg\max_{\boldsymbol{y}} \log P(\boldsymbol{y}|\boldsymbol{x}) + \lambda \log P_C(\boldsymbol{y}) \qquad (1)$$

Here, $\boldsymbol{x}$ denotes acoustic observations, and $\boldsymbol{y}$ the subword-unit sequence. P is the probability estimation from the E2E model, and P_C is the biasing rescoring probability. λ controls the weight of contextual LM in rescoring.

For E2E biasing, [11] explored biasing at beginning of word, end of word, and at subword units. The authors find that unlike biasing at the end of word in conventional models, biasing at subword units with weight pushing prevents candidates from being pruned early from the beam. In [13], wordpieces have been shown to outperform graphemes in biasing since they create a sparser match of biasing units. All these improvements lead to significantly better biasing which is comparable to the state-of-the-art conventional model [6]. To avoid over-biasing, [13] also proposed to only activate biasing phrases when they are proceeded by a set of prefixes.

2.2. Phoneme Mapping

Cross-lingual phoneme mapping has been used in conventional systems for recognizing foreign words [15]. First, a phoneme mapping is learned by aligning the pronunciations between foreign and target languages using TTS-synthesized audio and a pronunciation learning algorithm [18]. In inference, foreign words are first built to an FST (G). Lexica (L) are constructed between target-language phonemes and foreign words using phoneme mapping, and then left composed with G to construct a dynamic class LM for decoding:

$$G'_d = Det(L) \circ G_d \qquad (2)$$

where d denotes a dynamic class label. In Section 3.2, we describe how phoneme mapping is incorporated to a wordpiece-phoneme E2E model for contextual biasing.

3. Phoneme-Based Biasing

The focus of this work is to bias toward rare cross-lingual words which are typically missing from the training set. We propose to do that by utilizing phonemes, which are not affected by orthography. Specifically, we augment the wordpiece modeling space of an E2E model with phonemes to train a wordpiece-phoneme model.

3.1. Wordpiece-Phoneme Model

A wordpiece-phoneme model differs from a wordpiece-only model in that it may decompose a few words to phonemes in training. The output of the model is a single softmax whose symbol set is the union of wordpiece and phoneme symbols. We use a pronunciation lexicon to obtain phoneme sequences of words. Since phonemes show strength in recognizing rare words [16], we want to present these words as phonemes more often. In a target sentence, we decide to randomly present the ith word as phonemes with a probability $p(i) = p_0 . \min(\frac{T}{c(i)}, 1.0)$ where p_0 and T are constants and $c(i)$ is an integer representing the number of time the word appears in our entire training corpus. Therefore, the words that appear T times or less will be presented as phonemes with probability p_0. For words that appear more than T times, the more frequent they are, the less likely they are presented as phonemes[2]. Note that the decision of whether to use wordpieces or phonemes is made randomly at each gradient iteration, and thus a given sentence could have different target sequences at different epochs. We use context-independent phonemes as in [16].

3.2. Biasing FST for Phonemes

In inference, cross-lingual biasing words are converted to English phonemes to rescore the phoneme outputs of the wordpiece-phoneme model. In our work, phoneme mapping is represented by a dictionary which contains human generated source-language to target-language phoneme pairs [15], and the X-SAMPA phoneme set is used for all languages. For example, given a French word "Créteil", we tokenize it into phonemes using the French pronunciation lexicon, i.e. "Créteil" → "k R e t E j", and then map the French phonemes to English phonemes one by one: "k R e t E j" → "k r\ E t E j". Note that the mapping is needed since our wordpiece-phoneme model contains only English phonemes. The English phoneme sequence is then used to construct a phoneme-level FST for biasing. Weight pushing is used to assign weights at the phoneme level and failure arcs are added to avoid over-biasing similar to [11]. Figure 1 shows an example of a contextual FST for the word "Créteil" at the phoneme level. The biasing FST is then used to rescore the phoneme outputs of the wordpiece-phoneme model on the fly, using Eq. (1).
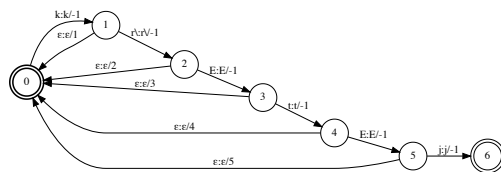

Figure 1: *Contextual FST for the word "Créteil" using a sequence of English phonemes "k r\ E t E j".*

3.3. Decoding Graph

To generate words as outputs, we search through a decoding graph similar to [16] but accept both phonemes and wordpieces. An example is shown in Figure 2. The decoding FST has wordpiece loops around state 0 (we show only a few for simplicity), but also has a pronunciation section (states 1 through 14). The
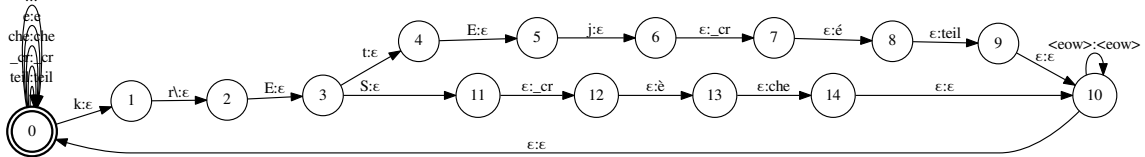
Figure 2: *Decoding graph for the words "crèche" (daycare) with English cross lingual pronunciation "k r\ E S" and "créteil" (a city) with pronunciation "k r\ E t E j". For clarity, we omitted most wordpieces for the state 0.*

pronunciation section is a prefix tree with phonemes as inputs, and outputs are wordpieces of the corresponding word produced by the WPM in Section 3.1. Specifically, for each word in the biasing list, we look up pronunciations from the lexicon and split the word into its constituent wordpieces. Input phoneme labels are accepted and transduced into wordpieces. Input wordpiece labels are accepted by the wordpiece loops. The final output symbols, which are always wordpieces, are concatenated into words.

Based on [16], we add two improvements to the decoding strategy. First, during decoding we consume as many input epsilon arcs as possible thus guaranteeing that all wordpieces in word are produced when all corresponding phonemes are seen in the input. Second, we merge paths that have the same *output* symbols. Given the nature of our training and decoding, a given word can be output either directly in wordpieces, or transduced from phonemes to wordpieces. Since the input symbols are different, each hypothesis has a different probability. We keep track of equivalent hypotheses and recombine them by adding their probabilities, assigning the total probability to the most likely hypothesis, and dropping the others from the beam.

4. Experiments

4.1. Data Sets

Our training set contains 35 million English utterances with a total of around 27,500 hours. These utterances are sampled from Google's general English traffic, and are anonymized and hand-transcribed for training. To increase training diversity, clean utterances are artificially corrupted by using a room simulator, varying degrees of noise, and reverberation such that the overall signal-to-noise ratio (SNR) is between 0dB and 30dB, and an average SNR is 12dB [19]. The noise sources are from YouTube and daily life noisy environmental recordings.

Utterances with cross-lingual words are hardly present in our data set, and thus we use a TTS engine, parallel-wavenet [20], to synthesize utterances for evaluation. We choose French as the foreign language, and the utterances consist of navigation queries (e.g. "directions to Créteil"). There are in total 1K utterances and we refer to this set as the *Directions* test set. For each utterance, the bias set contains 1K words including the ground-truth place name and unrelated French place names. Since all biasing words are in a foreign language, they have never been seen in training. In decoding, all biasing words are used to construct a contextual FST with each arc having the same weight. In later evaluation, this weight is tuned independently for different models.

On the other hand, to evaluate how the wordpiece-phoneme model performs on the regular English recognition task, we sampled a total of 30.5K English utterances from general Google traffic as the no-biasing test set.

4.2. Model Training

Similarly to [6], an input utterance is divided to 25-ms frames, windowed and shifted at a rate of 10 ms. A 80-dimensional log-Mel feature is extracted at each frame, and the current frame and two frames to the left are concatenated to produce a 240-dimensional log-Mel feature. These features are then downsampled at a rate of 30 ms. We use RNN-T as the sequence-to-sequence model. Similar to [6], the encoder of the RNN-T consists of 8 Long Short-Term Memory (LSTM) [21] layers and the prediction network contains 2 layers. Each LSTM layer contains 2,048 hidden units followed by a 640-dimensional projection layer. A time-reduction layer is added after the second layer to improve the inference speed without accuracy loss. Outputs of the encoder and prediction network are fed to a joint-network which has 640 hidden units, which is then followed by a softmax layer containing 4,096 output units. Specifically, the output units contain 41 context-independent phonemes and the rest are wordpieces.

As described in Section 3.1, we use a lexicon containing about 430,000 words with their frequencies to determine when to use phoneme sequences. The lexicon contains words and their frequencies from training data, and is trimmed by removing homophones (e.g. "flower" and "flour"), homographs (e.g. "live" as a verb or adjective), and pronunciation variants (e.g. "either"). It thus only contains entries that are unambiguous when going from spelling to pronunciation or the other way around. We do not generate phonemes for out-of-lexicon words using a trained grapheme-to-phoneme model. The intuition is that when pronunciation is not known, it is simpler and cleaner to let the E2E model infer the pronunciation rather than bring in another independently trained model. In addition, we use the written form of a transcript and do not use any verbalizer. Thus, words like "$9.95" were never presented as phonemes. In model training, a <eow> symbol is inserted between words to identify spacing. The model contains around 120M parameters in total. All RNN-T models are trained in Tensorflow [22] on 8 × 8 Tensor Processing Units (TPU) slices with a global batch size of 4,096.

4.3. WERs and Comparisons

We compare the biasing results of the wordpiece-phoneme model to a grapheme-only model and a wordpiece-only model. The latter two models have the same structure as the wordpiece-phoneme model. The difference is that the grapheme model has 76 graphemes as outputs and the wordpiece model has 4,096 wordpieces. This leads to around 117M and 120M parameters for the grapheme model and wordpiece model, respectively. Note that the two model's output symbols are in English and they are trained using all-English data described in Section 4.1. For these two models, biasing is done at the grapheme level or wordpiece level alone using the English transliterated versions of French biasing words. WERs of the *Directions* set are shown in Table 1.

Table 1: *Comparison of WERs (%) of different models for a foreign place name recognition task and Google general English traffic.*

Model	Biasing unit	*Directions* w/o Bias	*Directions* w/ Bias	General Traffic (English)
Grapheme	Graphemes	36.0	32.8	5.9
WPM	Wordpieces	38.3	30.0	5.6
Wordpiece-Phoneme	Phonemes	35.9	27.5	5.7
+ wordpiece biasing	Wordpieces and phonemes	35.9	26.9	5.7

First, we see that all three models perform poorly without biasing. This is because the place names are in French and they have never been seen in training, i.e. an word OOV rate of nearly 100%[3]. Secondly, we see in Table 1 that all models performs substantially better with biasing. The WER reductions range from 9%-23% relatively for different models when compared to the no-bias case. Comparing different biasing strategies, we find that the wordpiece-phoneme model performs the best: 16% relatively better than the grapheme model, and 8.3% better than the wordpiece model. We attribute the superior performance of the wordpiece-phoneme model to the robustness of phonemes to OOV words, as observed in [16].

Since the wordpiece-phoneme model contains both wordpieces and phonemes as modeling units, we can further perform wordpiece biasing in addition to phoneme-based biasing by building a wordpiece FST in parallel to the phoneme FST. This further reduces the WER by 2%, as shown in the bottom row in Table 1. This shows that wordpiece and phoneme biasing are complementary to each other. We note that the same weights are used for both phoneme and wordpiece biasing, and empirically we did not find significant improvements by using different weights. On the other hand, for wordpiece model biasing, our results are consistent with the observation in [13] that the wordpieces perform better than graphemes because of its sparsity in matching longer units.

To further understand how biasing helps recognizing French place names, we present some wins of wordpiece-phoneme model in Table 2. We can see that biasing helps produce the correct French words, and in contrast, phonetically similar but wrong English words are produced when without biasing. On the other hand, we present some typical recognition errors in Table 3. We see that errors are mainly due to phonetically similar words in French. We will analyze how the biasing performance changes as the number of biasing words changes in Section 4.4.

To ensure there is no regression in no-biasing scenarios, we compare three models in decoding regular English utterances from general Google traffic. In decoding, we turn the biasing mechanism off by using an empty list of biasing phrases. As shown in the last column of Table 1, the wordpiece model performs better than the grapheme model as in [6]. The wordpiece-phoneme model performs a little better than the grapheme model, and we attribute that to the higher frequency of wordpieces during training. Compared to the wordpiece model, the wordpiece-phoneme model has a slight degradation (0.1% absolute WER). This is due to the introduction of phonemes in modeling. One potential approach to improve regression is to incorporate an English external language model for phonemes in rescoring, similarly to the wordpiece-based rescoring in [10]. However, we note that the regression is significantly smaller than the all-phoneme model in [16] .

---

[3]Note that we transliterate the ground truth transcripts of the *Directions* set to English for WER computation for the grapheme and wordpiece models since the they cannot produce words with French diacritics.

Table 2: *Comparison of correctly recognized French words (w/ biasing) and errors (w/o biasing).*

w/ Bias (correct)	w/o Bias
directions to Champs-Élysées	directions to shaw city
directions to Bouches-du-Rhône	directions to bushnell
directions to Ardèche	directions to adesh

Table 3: *Examples of wrongly recognized French words.*

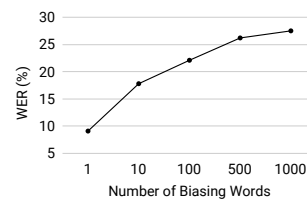| Estimated	Reference
directions to Citroën	directions to Saint-Honoré
directions to Marne-la-Vallée	directions to Marcq-en-Barœul
directions to Métropole	directions to Megève

Figure 3: *WER (%) as a function of number of biasing words.*

4.4. Effect of Number of Biasing Words

Given examples in Table 3, we are curious how competing biasing words affect recognition. We thus randomly choose a fixed number of biasing words (including the ground truth one) and vary the number to see how WER changes. Figure 3 shows that WER for the *Directions* set is 9.1% when only the ground truth word is present (i.e. number of biasing words is 1), and the rate increases quickly when the total number of biasing words increases. We attribute the quick degradation to the significant matching confusion in the phoneme prefixes of the words as the number of biasing words increases (as confirmed by phonetically similar French place names in Table 3). One interesting direction would be to increase the length of the phonemic units to create a sparser match.

5. Conclusion

In this work we proposed a wordpiece-phoneme RNN-T model and phoneme-level contextual biasing to recognize foreign words. Biasing at the phoneme level enables us to avoid the OOV problem in the wordpiece model. Evaluating on a test set containing navigation queries to French place names, we show the proposed approach performs significantly better than a state-of-the-art grapheme and wordpiece model, by 16% and 8%, respectively in terms of relative WER reductions. Wordpiece biasing is complimentary to phoneme biasing and adds a further 2% reduction. Lastly, since wordpieces perform better than graphemes [6] in E2E modeling, it would be interesting to explore longer phonemic units such as phoneme pieces for biasing.

6. Acknowledgement

7. References

[1] H. Soltau, H. Liao, and H. Sak, "Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition," *arXiv preprint arXiv:1610.09975*, 2016.

[2] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[3] K. Rao, H. Sak, and R. Prabhavalkar, "Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 193–199.

[4] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4960–4964.

[5] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4774–4778.

[6] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang *et al.*, "Streaming end-to-end speech recognition for mobile devices," *ICASSP*, 2019. [Online]. Available: https://arxiv.org/pdf/1811.06621.pdf

[7] P. Aleksic, M. Ghodsi, A. Michaely, C. Allauzen, K. Hall, B. Roark, D. Rybach, and P. Moreno, "Bringing contextual information to Google speech recognition," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[8] K. Hall, E. Cho, C. Allauzen, F. Beaufays, N. Coccaro, K. Nakajima, M. Riley, B. Roark, D. Rybach, and L. Zhang, "Composition-based on-the-fly rescoring for salient n-gram biasing," 2015.

[9] I. Williams, A. Kannan, P. Aleksic, D. Rybach, and T. N. Sainath, "Contextual speech recognition in end-to-end neural network systems using beam search," *Proc. Interspeech 2018*, pp. 2227–2231, 2018.

[10] A. Kannan, Y. Wu, P. Nguyen, T. N. Sainath, Z. Chen, and R. Prabhavalkar, "An analysis of incorporating an external language model into a sequence-to-sequence model," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1–5828.

[11] G. Pundak, T. N. Sainath, R. Prabhavalkar, A. Kannan, and D. Zhao, "Deep context: end-to-end contextual speech recognition," *arXiv preprint arXiv:1808.02480*, 2018.

[12] M. Schuster and K. Nakajima, "Japanese and korean voice search," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 5149–5152.

[13] D. Zhao, T. N. Sainath, D. Rybach, D. Bhatia, B. Li, and R. Pang, "Shallow-fusion end-to-end contextual biasing," *To appear in Interspeech 2019*, 2019.

[14] A. Bruguier, R. Prabhavalkar, G. Pundak, and T. N. Sainath, "Phoebe: Pronunciation-aware contextualization for end-to-end speech recognition," in *to appear in Proc. ICASSP, 2019*. IEEE.

[15] A. Patel, D. Li, E. Cho, and P. Aleksic, "Cross-lingual phoneme mapping for language robust contextual speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5924–5928.

[16] T. N. Sainath, R. Prabhavalkar, S. Kumar, S. Lee, A. Kannan, D. Rybach, V. Schogol, P. Nguyen, B. Li, Y. Wu *et al.*, "No need for a lexicon? Evaluating the value of the pronunciation lexica in end-to-end models," *ICASSP*, 2017.

[17] K. Irie, R. Prabhavalkar, A. Kannan, A. Bruguier, D. Rybach, and P. Nguyen, "Model unit exploration for sequence-to-sequence speech recognition," *arXiv preprint arXiv:1902.01955*, 2019.

[18] A. Bruguier, D. Gnanapragasam, L. Johnson, K. Rao, and F. Beaufays, "Pronunciation learning with RNN-transducers," *Proc. Interspeech 2017*, pp. 2556–2560, 2017.

[19] C. Kim, A. Misra, K. Chin, T. Hughes, A. Narayanan, T. Sainath, and M. Bacchiani, "Generation of large-scale simulated utterances in virtual rooms to train deep-neural networks for far-field speech recognition in google home," 2017.

[20] A. v. d. Oord, Y. Li, I. Babuschkin, K. Simonyan, O. Vinyals, K. Kavukcuoglu, G. v. d. Driessche, E. Lockhart, L. C. Cobo, F. Stimberg *et al.*, "Parallel wavenet: Fast high-fidelity speech synthesis," *arXiv preprint arXiv:1711.10433*, 2017.

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

[22] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 2016, pp. 265–283.