



Phonological Features for 0-shot Multilingual Speech Synthesis

Marlene Staib¹, Tian Huey Teh¹, Alexandra Torresquintero¹, Devang S Ram Mohan¹,
Lorenzo Foglianti¹, Raphael Lenain^{2*}, Jiameng Gao¹

¹Papercup Technologies Ltd.

²Novoic

{marlene,tian}@papercup.com

Abstract

Code-switching—the intra-utterance use of multiple languages—is prevalent across the world. Within text-to-speech (TTS), multilingual models have been found to enable code-switching [1–3]. By modifying the linguistic input to sequence-to-sequence TTS, we show that code-switching is possible for languages unseen during training, even within monolingual models. We use a small set of phonological features derived from the International Phonetic Alphabet (IPA), such as vowel height and frontness, consonant place and manner. This allows the model topology to stay unchanged for different languages, and enables new, previously unseen feature combinations to be interpreted by the model. We show that this allows us to generate intelligible, code-switched speech in a new language at test time, including the approximation of sounds never seen in training.

Index Terms: speech synthesis, zero-shot, code-switching

1. Introduction

End-to-end TTS models such as Tacotron 2 are able to generate highly natural-sounding speech, mapping text inputs directly into acoustic outputs [4]. Recently, transformations of text used in traditional TTS, such as phonemes [5], have been found to improve naturalness over characters as inputs within end-to-end models [6]. Similar advantages have been found with phonological features (PFs), such as place and manner of articulation [7], when used in addition to, or in place of phonemes as inputs to DNN TTS models [2]. Using phonemes, PFs or both has proven an essential step in training multilingual models [1–3]. For low-resource settings, performance improvements in both TTS [8] and Automatic Speech Recognition [9] were found with PFs over phonemes alone, using either multilingual/multitask or transfer learning from high-resource TTS.

Zhang and colleagues [1] first noted the ability of a multilingual, multi-speaker Tacotron 2 to code-switch, i.e. produce natural-sounding speech of one speaker in two languages, only one of which has been previously seen for that speaker. In their setup, which uses phonemes as inputs, code-switching is only possible for languages within the training data. However, training data is only readily available for a fraction of the world’s 5000-7000 languages. The ability to generate appropriate pronunciations, minimally for foreign names, organisations or locations, without requiring large multilingual corpora, is therefore highly desirable in speech applications.

An alternative approach is presented in [10], which explores Unicode bytes as a possible input to a multi-speaker, multilingual Tacotron 2. The advantage over single-valued inputs such as characters or phonemes is that new characters can be added without changing the model topology. This makes it suitable

for transfer learning across languages. However, since Unicode only encodes typographic, and not phonological information, nothing is learned in this model about unseen byte combinations, likely requiring at least parts of the model to be relearned entirely when enrolling a new language.

Gutkin and colleagues [2, 3, 8, 11] demonstrate the possibility to synthesise a previously unseen language within multilingual models trained on 9-39 languages, partially with phylogenetic relationships between each other. They use PFs [2, 8] or a combination of PFs and phonemes [2, 3, 11] as inputs to their neural, multi-lingual TTS models. They show that various, automatically derived phonological feature sets can be used to either replace or supplement phonemes as input features, yielding improved intelligibility over a phoneme-only baseline across a variety of trained and even untrained languages [2]. To our knowledge, they do not attempt to synthesise any phonemes completely unseen in training. Notably, models which concatenate PFs to phonemes suffer the same constraints on extending the phoneme inventory as phoneme or character-based models, and do not allow for previously unseen phonemes to be synthesised without manual mapping or further training. Finally, these models require a substantial number of training languages to allow for the generation of a new language.

PFs offer a shared model topology across languages, similar to the “byte-like” representation used in [10], and maintain the connection to abstract, phonological categories, while also providing explanatory power on a level closer to the acoustics of an utterance [12]. While the applicability of a specific PF set to *all* languages is questionable, certain phonological contrasts, such as “front–back”, have been shown to generalise across various language families [13]. At a lower bound, where phonological categories such as “fricatives”, “rounded vowels”, etc. do not share any acoustic properties amongst themselves, PF vectors can be seen as unique identifiers, i.e. the “byte-version” of phonemes. If, as we expect, phonological categories have acoustic correlates, we are able to transfer what is learnt to new, unseen or infrequent combinations of sounds. In the case where the acoustics can be disentangled into (somewhat orthogonal) PFs, they would enable us to create new sounds, even including those not present in any human language.

We extend the work in [2, 3, 8, 11] by showing that PFs enable code-switching into an untrained language within a small multilingual, or even a monolingual model. Most importantly, we investigate the model’s ability to synthesise sounds completely unseen in training (as opposed to an untrained language containing only previously encountered phonemes). While we envision the application of this research to be in code-switching, we conduct our experiments by synthesizing full sentences in an untrained language, marking an extreme case where *all* words are “code-switched”. Further applications may include TTS for low resource languages, as our experiments simulate a zero-resource setting (irrespective of a particular choice of language).

*work done while at Papercup Technologies Ltd.

2. Phonological features

2.1. Feature set

A range of PF sets have been put forward, including structured primes—nuclear units used to compose phonemes [14], binary features—such as “voice”, “tense”, “vocalic” [15], or multi-valued ones—such as place and manner of articulation [7]. Within the latter, some features can be seen as continuous, for example the horizontal dorsum position for vowels [9].

In this paper, we use the following set of 10 categorical, multi-valued PFs, 9 of which are directly read from the IPA [16]: *consonant/vowel*, *voicing (voiced/unvoiced)*, *vowel frontness*, *vowel openness*, *vowel roundedness*, *stress on vowel*, *consonant place*, *consonant manner*, and *diacritic (e.g., nasalised, velarised)*. Features relating to only vowels are set to NULL for consonants, and vice versa. The tenth feature, “*symbol type*”, is used to integrate symbols that mark, e.g., silences, the end of a sentence, or word boundaries, with all phoneme symbols sharing a single value on this feature.

Each multi-valued feature is 1-hot encoded into a varying number of binary variables, making up a total of 60 binary features. Phoneme identity is not used as a separate feature, as in other work [3, 8, 11], since this inhibits the encoding of unseen phonemes in our 0-shot experiments.

2.2. Model architecture

2.2.1. Baseline model

We use a (monolingual/multi-lingual), multi-speaker variant of Tacotron 2 [1, 4] as our baseline, mapping from phonemes to mel-filterbank features (MFBs). The input text is transformed into phonemes using a linguistic frontend (see 4.2). The Griffin-Lim vocoder [17] is used to map from MFBs to waveform.

2.2.2. Phonological features model

In our proposed PF model, the phoneme embedding table in the baseline is replaced with a single, linear feedforward layer on top of our binary input features. The output dimensionality of this layer is the same as the embedding size of the baseline model. Since the number of binarised PFs in the input feature vector is less than the size of the phoneme inventories used in our experiments (see 4.1), the number of parameters used to represent the input is less than in our baseline model.

A phonemic transcription is obtained from text in the same way as in our baseline model, and then further mapped to its PF representation using a dictionary-lookup based on the IPA. Some additional mappings between IPA and resource-specific symbols were necessary, due to the lexical resources used (see 4.2). Here, we chose the IPA as it is a widely adopted resource. In principle, this framework extends to other PF sets [18–20] for which automatic extraction is also possible [2].

3. 0-shot TTS

3.1. Synthesizing unseen phonemes

In our proposed model, our method (AUTO) of synthesizing unseen phonemes is straightforward: PFs of new, out-of-sample (OOS) phonemes are simply inferred from the IPA. PFs can be directly fed to the network without any modifications.

Table 1: *Data set statistics, including the number of hours in the training set, number of unique phonemes and out-of-sample target phonemes in the test language German (OOS).*

Corpus	Hours	Speakers	Phonemes	OOS
VCTK	27.5	109	73	14
MIX	35.4	141	89	9

3.2. Baselines

To demonstrate the effectiveness of our approach, we consider two baselines of “0-shot” TTS in an untrained language: 1) RANDOM, which maps OOS phonemes to a new, randomly initialised (untrained) vector in the phoneme embedding table; and 2) MANUAL, for which we manually map all OOS phonemes to their “closest” existing phoneme in the trained embedding table of the Tacotron 2 encoder. We define the closest phoneme as one with a minimal number of differing PFs (e.g., the rounded version of an unrounded vowel), which sounds close to the target sound to a native speaker. In rare cases, a perceptually closer match is found that has less overlap in PF space—e.g., when mapping [ʀ] to [ɹ] (see 4.6 for a discussion). RANDOM serves as a lower bound, demonstrating what can be achieved without additional linguistic knowledge. The comparison of AUTO against MANUAL tests whether our model can go beyond the training data and approximate new sounds from never seen combinations of features, thus outperforming an expert mapping. Minimally, we expect our model to be able to automatically and reliably “pick” the closest sound, thereby performing on par with MANUAL.

4. Experiments

4.1. Data

We use two corpora for experimentation: VCTK [21], an open-source, multi-speaker, multi-dialect corpus in English, and Adrianex, a proprietary, multi-speaker corpus in Mexican Spanish. We compare the performance of the baseline and proposed model, using A) VCTK only or B) both corpora combined (MIX). German is used as a target language (see 4.5.1). Statistics, including the number of OOS phonemes in German, are shown in Table 1. The two data settings are used to explore the relationship between the number of unseen phonemes, and target language intelligibility. Audio was downsampled to 24 kHz, and 128-dimensional MFBs were extracted every 10 ms over a window of 50 ms.

4.2. Linguistic frontend

For English, phonemic transcriptions of the input text are obtained from the Received Pronunciation (RP) version of the Combilex dictionary [22]. For Spanish, pronunciations for each word are obtained using a set of Mexican Spanish pronunciation rules, modified from [23]. For test sentences in German, the lexicon from the German MARY-TTS voice [24] is used. For all models including the baseline, resource-specific phoneme-sets need to be mapped into a shared inventory, such as the IPA¹.

Diphthongs, which are represented as a single symbol in the English and German dictionary, are split into their compo-

¹Mapping tables for the listed resources, as well as an IPA-PF lookup dictionary are available at <https://github.com/papercup-open-source/phonological-features>

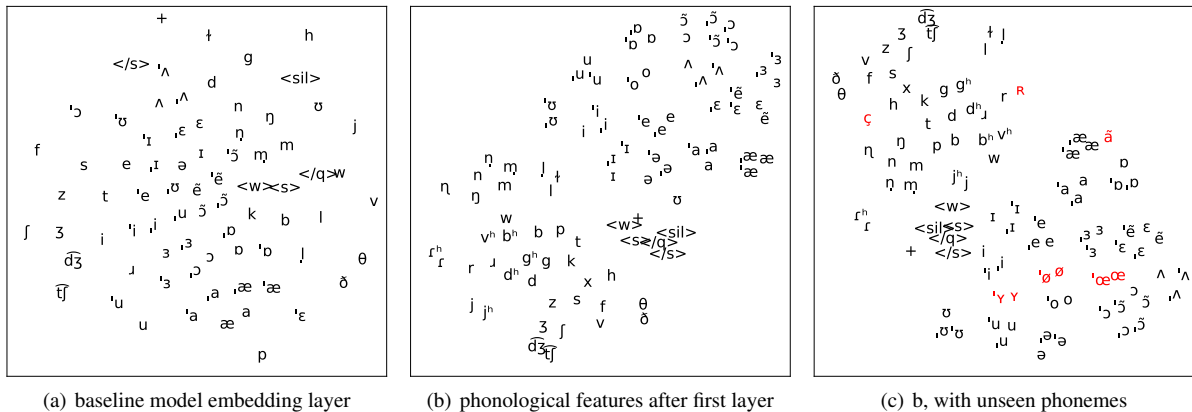


Figure 1: *t*-SNE plot of phoneme representations (first encoder layer) from the baseline and proposed model in the MIX data condition. Red in (c): additional phonemes unseen in training. (Note: (c) is a different *t*-SNE view from (b) of the same representation space.)

nent vowels. Lexical stress is added to the vowel in a stressed syllable. In the baseline model, stressed vowels are added as a separate symbol in the phoneme embedding table. For diphthongs, stress is added on the first vowel [25]. Vowel length information, which is available from the German, but not the English lexicon, is discarded.

4.3. Training protocol

We train our models for 200k iterations, with a batch size of 32 and an initial learning rate of $10e-3$, which is decreased to $10e-4$ after 100k iterations. Following [4], teacher-forcing in conjunction with dropout is used during training, to align predicted and true MFB output sequences, \hat{y} and y . The L1 norm between \hat{y} and y is used as a loss function for Maximum Likelihood training. To discourage over-reliance on teacher-forcing in the early stages of training, we start with a Prenet layer size of 64, and increase it to 256 after 40k iterations. We found this initial reduction to be essential for the attention network to successfully train across different data settings. All other training parameters are as described in [4].

4.4. Learned representations in the encoder

We use *t*-SNE plots [26] to inspect the phoneme representation spaces learned by our proposed versus the baseline model (Fig. 1). In both models, we observe a meaningful arrangement of phonemes that lends itself to phonological interpretation. For instance, vowels tend to be closer to each other than to consonants, as is the case for voiced versus unvoiced versions of a consonant, and phonemic classes such as nasals, fricatives, stops, etc. However, there are no identifiable clusters in the baseline model, and neighbouring phonemes appear more or less equidistant from each other, likely serving as unique identifiers without further meaningful axes of variation. In contrast, tight clusters emerge in the projection space of our proposed model for vowels versus consonants, nasals, stops, fricatives, etc., suggesting a richer, more meaningful space learnt. Fig. 1(b) and 1(c) also suggest an intuitive information hierarchy; with PFs clustering by broader group (vowel/consonant/other) first, then—within consonants—by manner second, place third, and everything else (voicing, diacritic) last.

Moreover, Fig. 1(c) shows that new, unseen phonemes get projected into intuitive places within that space without further

Table 2: Test set statistics, including number of sentences (*n* sents), sentence length (*sent. len.*) and Unseen Phoneme Rate (UPR) in percent, in each data setting (where μ is the mean).

	n sents	sent. len.	UPR(VCTK)%	UPR(MIX)%
1	30	7	0 – 5.6 ($\mu=2.9$)	0 – 4.3 ($\mu=1.8$)
2	101	7 – 26	0 – 6.5 ($\mu=2.3$)	0 – 5.4 ($\mu=1.5$)

training. This supports the expectation that, even if the model is unable to interpolate between feature combinations to produce new sounds, it will be able to automatically map unseen sounds to the closest seen sound in feature space, with high accuracy.

4.5. Formal listening evaluations

4.5.1. Method

Listening tests were conducted to test whether 1) 0-shot German speech from AUTO is more intelligible than RANDOM, and competitive with MAPPED; 2) AUTO outperforms RANDOM, and is competitive with MAPPED in terms of listener preference; and 3) listener preference is the same for the baseline and the proposed model in the trained language (English).

For 3, the test set consisted of a random subset of 96 validation set sentences from 6 selected, RP English VCTK speakers (3 male, 3 female). For 1 and 2, sentences were randomly sampled from Wikipedia articles in German, subject to having seven words (or more, as required), all in-vocabulary. The same 6 VCTK speakers were used to synthesise these two test sets. Statistics on the different test sets are shown in Table 2. The Unseen Phoneme Rate (UPR) was calculated as the number of OOS phonemes in a target utterance, divided by the total number of phonemes in that utterance.

Listener preference was evaluated using A/B preference tests. For intelligibility, listeners were asked to transcribe the generated samples, and word level accuracy was measured. A block design was used to reduce the amount of variability arising from different listeners and different sentence-model pairings. All tests were performed online by native listeners, recruited through Amazon Mechanical Turk [27]. For preference tests, language proficiency was assessed with a preliminary task of transcribing a sample in the target language. For the transcription task, we discarded responses from participants with an

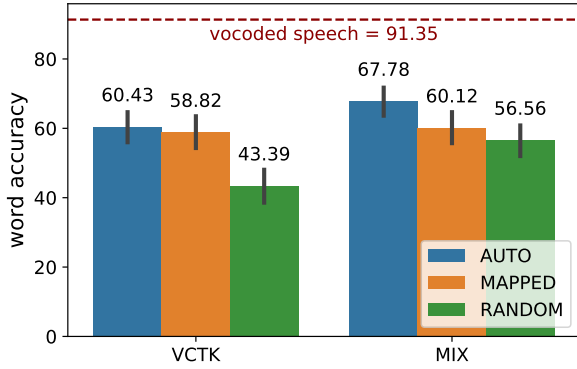


Figure 2: Results for intelligibility of 0-shot German.

overall Word Error Rate (WER) above 80%, prior to analysis. Listeners in this category appeared to be non-native speakers or people who had not seriously attempted the task. After filtering, the total number of listeners was 20 for intelligibility and 30 for preference tests.

4.5.2. Results

Figure 2 shows that AUTO produces significantly more intelligible speech compared to RANDOM in both data conditions (using pair-wise t-test comparisons; $p < 0.001$), and outperforms MANUAL in the MIX data setup ($p < 0.05$). Interestingly, word level accuracy is similar across data conditions for MANUAL, while AUTO and RANDOM improve with the additional language present in MIX, compared to VCTK alone. The degradation in intelligibility in all tested methods against vocoded human speech is likely due to the strong English accent present in the 0-shot samples. Producing accent-free, 0-shot speech, perhaps relying on more sophisticated representations of typology and language [11, 28] or with adversarial losses [1] remains a challenging task for future research.

We find a weak negative correlation between UPR and word accuracy across data settings in RANDOM ($r = -0.12, p < 0.05$), but not in AUTO or MANUAL. This suggests that, while RANDOM is exposed to pronunciation (and subsequent perception) errors arising from unseen phonemes, AUTO and MANUAL are both effective strategies to overcome or at least attenuate them. Further research is needed to determine how and why pronunciations in AUTO improve with more languages, more data or a larger input phoneme inventory. Since we were unable to measure a clear relationship between UPR and word accuracy in AUTO, phoneme coverage alone may not fully explain this effect. Another possibility is that, rather than just exploiting a greater number of learned phonemes, AUTO is actually able to learn a more sophisticated phoneme projection space when trained on MIX.

No clear trend in listener preference was found for any of the compared methods (Fig. 3(a)). A potential explanation is that listeners’ preference was heavily influenced by the strong English accent present in the 0-shot speech, which may have overshadowed the preference for any particular model. It is also conceivable that the improved phoneme representation primarily affects *pronunciation* quality, which manifests itself mostly in intelligibility, over other metrics.

We also did not observe a significant difference in listener preference for the baseline versus our proposed model in English, in either of the data settings (Fig. 3(b)), indicating that

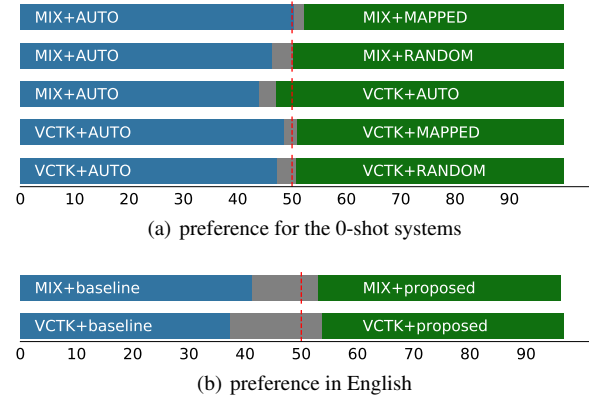


Figure 3: Results of the listening evaluations.

the ability to do 0-shot TTS in a new language in our proposed model does not come at the cost of reduced naturalness or quality in the trained language.

4.6. Informal listening evaluations

Through informal listening, we find that most of the OOS phonemes collapse to neighbouring in-sample phonemes in the audio output. Most often, this produces a perceptually agreeable mapping, e.g. from [ç] to [j] and [y] to [i]. In one case, within the monolingual (VCTK) model, the collapse is inappropriate, mapping the trill [ʀ] to the stop [g]. This does not happen for the multilingual (MIX) model, where [ʀ] sounds either like the alveolar trill [r], or approximant [ɹ] in the generated output². When a feature is completely unobserved in training (such as “trill” in the VCTK data setting), a manual mapping may be preferable to human listeners, e.g., from [ʀ] to [ɹ] (sharing few PFs, but having a common graphemic symbol ‘r’).

5. Conclusion and Future Work

By replacing the character input in Tacotron 2 with a relatively small set of IPA-inspired features, we were able to create a model topology which is language independent, and allows for the automatic approximation of sounds unseen in training, exceeding or matching the performance of various baselines including a resource-intensive expert mapping approach.

Further research is needed to validate this work for different pairs of source and target languages. This involves a more thorough investigation of the effect of the number of training languages, the amount of overlap between phoneme and PF inventories, as well as other typological and phylogenetic relationships. Another avenue of interest would be to further disentangle PFs, as well as accent, in order to generate unseen combinations of features more accurately.

6. Acknowledgements

We would like to thank our advisor, Simon King, for his input to this research, Doniyor Ulmasov, for his questions which inspired the idea for this paper, and the anonymous reviewers for their valuable feedback.

²Samples can be found at <https://research.papercup.com/samples/phonological-features-for-0-shot>

7. References

- [1] Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Z. Chen, R. Skerry-Ryan, Y. Jia, A. Rosenberg, and B. Ramabhadran, "Learning to speak fluently in a foreign language: Multilingual speech synthesis and cross-language voice cloning," *INTERSPEECH*, pp. 2080–2084, 2019.
- [2] A. Gutkin, M. Jansche, and T. Merkulova, "FonBund: A library for combining cross-lingual phonological segment data," *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, pp. 2236–2240, 2018.
- [3] I. Demirsahin, M. Jansche, and A. Gutkin, "A unified phonological representation of South Asian languages for multilingual text-to-speech," *SLTU*, pp. 80–84, 2018.
- [4] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerry-Ryan *et al.*, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4779–4783, 2018.
- [5] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, 2009.
- [6] J. Fong, J. Taylor, K. Richmond, and S. King, "A comparison between letters and phones as input to sequence-to-sequence models for speech synthesis," *10th ISCA Speech Synthesis Workshop*, pp. 223–227, 2019.
- [7] R. Jakobson, C. G. Fant, and M. Halle, *Preliminaries to speech analysis: The distinctive features and their correlates*. Cambridge, Massachusetts: The MIT Press, 1951.
- [8] A. Gutkin, "Uniform multilingual multi-speaker acoustic model for statistical parametric speech synthesis of low-resourced languages," *INTERSPEECH*, pp. 2183–2187, 2017.
- [9] S. Stüker and A. Waibel, "Porting speech recognition systems to new languages supported by articulatory feature models," *Speech and Computer, SPECOM*, 2009.
- [10] B. Li, Y. Zhang, T. Sainath, Y. Wu, and W. Chan, "Bytes are all you need: End-to-end multilingual speech recognition and synthesis with bytes," *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5621–5625, 2019.
- [11] A. Gutkin and R. Sproat, "Areal and phylogenetic features for multilingual speech synthesis," *INTERSPEECH*, pp. 2078–2082, 2017.
- [12] S. King and P. Taylor, "Detection of phonological features in continuous speech using neural networks," *Computer Speech & Language*, vol. 14, no. 4, pp. 333–353, 2000.
- [13] C. Johny, A. Gutkin, and M. Jansche, "Cross-lingual consistency of phonological features: An empirical study," *INTERSPEECH*, pp. 1741–1745, 2019.
- [14] J. Harris, *English sound structure*. Oxford, UK: Blackwell Publishers, 1994.
- [15] N. Chomsky and M. Halle, *The sound pattern of English*. New York, New York: Harper & Row, 1968.
- [16] International Phonetic Association, International Phonetic Association Staff *et al.*, *Handbook of the International Phonetic Association: A guide to the use of the International Phonetic Alphabet*. Cambridge University Press, 1999.
- [17] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.
- [18] S. Moran and D. McCloy, Eds., *PHOIBLE 2.0*. Jena: Max Planck Institute for the Science of Human History, 2019. [Online]. Available: <https://phoible.org/>
- [19] D. R. Mortensen, P. Littell, A. Bharadwaj, K. Goyal, C. Dyer, and L. Levin, "Panphon: A resource for mapping IPA segments to articulatory feature vectors," *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 3475–3484, 2016.
- [20] D. Dediu and S. Moisiuk, "Defining and counting phonological classes in cross-linguistic segment databases," *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pp. 1955–1962, 2016.
- [21] V. Christophe, Y. Junichi, and M. Kirsten, "CSTR VCTK Corpus: English multi-speaker corpus for CSTR voice cloning toolkit," *The Centre for Speech Technology Research (CSTR)*, 2016.
- [22] K. Richmond, R. A. Clark, and S. Fitt, "Robust LTS rules with the Combilex speech technology lexicon," *INTERSPEECH*, pp. 1295–1298, 2009.
- [23] P. Oplustil, *Multi-style Text-to-Speech using Recurrent Neural Networks for Chilean Spanish*. The University of Edinburgh, 2016.
- [24] M. Schröder and J. Trouvain, "The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching," *4th ISCA Speech Synthesis Workshop*, 2001.
- [25] P. Ladefoged and K. Johnson, *A course in phonetics*. Nelson Education, 2014.
- [26] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
- [27] K. Crowston, "Amazon Mechanical Turk: A research tool for organizations and information systems scholars," in *Shaping the Future of ICT Research. Methods and Approaches*. Springer, 2012, pp. 210–221.
- [28] Y. Tsvetkov, S. Sitaram, M. Faruqui, G. Lample, P. Littell, D. Mortensen, A. W. Black, L. Levin, and C. Dyer, "Polyglot neural language models: A case study in cross-lingual phonetic representation learning," *Proceedings of NAACL-HLT 2016*, pp. 1357–1366, 2016.