Refining Multilingual Pronunciation through G2P and ASR Integration

Anonymous ACL submission

Abstract

Pronunciation dictionaries are indispensable for applications in speech synthesis and language learning, providing word pronunciations across diverse languages. Grapheme-to-Phoneme (G2P) models are pivotal in creating these dictionaries. However, variations in pronunciation can arise due to language, context, dialect, and acoustic conditions, potentially introducing inaccuracies. To address this, we introduce an approach to refine G2P model outputs by utilizing an alignment and weighting algorithm to integrate results from an acoustic phone recognizer across several high and low-resource languages.

1 Introduction

Pronunciation dictionaries are imperative for a wide variety of tasks such Text-to-Speech (TTS), Automatic Speech Recognition (ASR), Language learning applications and many more. Oftentimes, pronunciation dictionaries in these tasks are either hand-crafted, requiring domain or language expertise, or acquired automatically using G2P models. However, problems arise in low-resource settings where acquiring and training on appropriate data is difficult. To address this, methods such as data augmentation (Ryan and Hulden, 2020, Hauer et al., 2020, Zhao et al., 2022) and zero-shot prediction (Zhu et al., 2022, Li et al., 2022) are employed to improve performance.

Another method for creating pronunciation dictionaries involves using acoustic data to recognize phonemes. This approach is particularly useful for unseen or out-of-vocabulary words and helps account for dialectal variations. Li et al. (2020) trained a universal phone recognition model, while Xu et al. (2021) and Siminyu et al. (2021) fine-tuned pretrained models to predict phonemes in unseen languages.

Efforts have also been made to combine G2P and phone recognition models in low-resource settings. Garg et al. (2024) employs self-supervised learning to create training lexicons for G2P models to improve TTS, while Ribeiro et al. (2023) uses acoustic models to learn out-of-vocabulary words and retrain G2P models. Additionally, Route et al. (2019) adopts a multimodal approach by training the G2P model to learn audio features as a separate task.

These methods primarily use acoustic information as part of the training data for G2P models. This might pose an issue when the acoustic models do not perform well due to noise in the speech segments or a sub-par performance of the model as a whole. In this paper, we adapt a method to integrate the output pronunciations of the G2P and phone recognition models (Aquino et al., 2019) using suitable alignment and weighting strategies, taking into account any sub-par performance of the acoustic model. Section 2 outlines the problem statement, the algorithm, and the overall system flow. Section 3 provides details of the dataset and models used, and Section 4 presents results for the entire dataset as well as a few examples.

2 Methodology

In this section, we formally define the task and detail the system flow and algorithm used to combine the outputs of the G2P and phone recognition models.

2.1 Problem Statement

Given a sample of `<word, audio>`, where *word* is a single word with the orthography of any of the included languages and *audio* is its corresponding spoken segment, the objective is to determine the most precise pronunciation of the given sample.

2.2 System Flow

We create a dataset of `<word, audio>` pairs across 120 languages from Wiktionary. The details of

the dataset are mentioned in Section 3.1. Using this dataset, we train an elementary ASR system to predict phones using the Kaldi speech recognition toolkit (Povey et al., 2011) and a G2P model as a weighted finite state transducer. Each of the aforementioned models generates their top-k predictions, which are subsequently aligned and weighted to determine the optimal pronunciation sequence.

2.3 Needleman–Wunsch algorithm

The Needleman–Wunsch algorithm (Needleman and Wunsch, 1970) is a dynamic programming algorithm used to solve the problem of global sequence alignment. It is primarily used to align amino acids or nucleotide sequences in the field of bioinformatics. Given 2 sequences $X = x_1x_2 \dots x_n$ and $Y = y_1y_2 \dots y_m$, a substitution matrix S where S_{x_i,y_j} denotes the score for aligning characters x_i and y_j, and a gap penalty d, we aim to find aligned sequences X' and Y' of equal length such that the alignment score is maximized. Thus, given an alignment matrix A, the score for the alignment of characters a_i and b_j is given by the following recurrence relation

$$A_{i,j} = \begin{cases} A_{i-1,j-1} + S_{x_i,y_i} & \text{aligned} \\ A_{i,j-1} + d & \text{gap in X} \\ A_{i-1,j} + d & \text{gap in Y} \end{cases} \quad (1)$$

In the context of aligning pronunciations, x_i and y_j are phones in the International Phonetic Alphabet (IPA) and S_{x_i,y_j} denotes the similarity between x_i and y_j. This similarity metric is computed with PyPhone (Zhang, 2018), a Python package that determines the distance between phonemes using 21 weighted features for each phoneme, differentiating between vowels and consonants.

Given the top-k pronunciations from the phone recognition and the G2P models, we first align the k pronunciations for each model using the Needleman–Wunsch algorithm, ensuring each pronunciation attains the same length by introducing gaps as necessary. We then compute a probability distribution for all the phones at each position. Consequently, each model produces a list of <phone, probability>, which are subsequently aligned in the same manner to identify the most probable phone at each position.

We must also consider the performance of the phone recognizer and weight its phones accordingly. Our goal is to enhance the output of the G2P model without degrading or overpowering it. Therefore, if the probability p of the phone recognizer's predicted phone falls below a certain threshold, we reduce the weight w of that phone in the recombination process. For our experiments, we set the weighting scheme as

$$w = \begin{cases} 1 & \text{if } p > 0.5 \\ 0.5 & \text{if } 0.5 \leq p < 0.2 \\ 0.2 & \text{if } p \leq 0.2 \end{cases} \quad (2)$$

3 Experimental Setup

3.1 Dataset

We compiled a dataset consisting of <word, pronunciation, audio> entries using the latest Wikimedia dump dated May 2, 2024. The entire Wikimedia dump contains approximately 10M words across over $4,500$ languages. Among these, only about $162,000$ words across 120 languages have associated pronunciations and audio files. Within this, 20 languages have less that 100 samples and 50 languages have less than 10 samples, highlighting the scarcity of data for low-resource languages. Each speech utterance averages 1.4 seconds in duration, as they consist of single words, resulting in a total of 62 hours of speech in the dataset. All pronunciations are provided in the IPA format, and the corresponding speech segments are originally stereo sampled at 44.1 kHz, subsequently down-sampled to single-channel, 16 kHz.

3.2 Models

Phone Recognizer. We extracted MFCC acoustic features, computed cepstral mean and variance normalization (CMVN) and used it to train a HMM-GMM acoustic model to directly predict phones instead of the orthographic representation of the word, thereby eliminating any extra errors introduced by the transformation. We used the SRILM toolkit to train a 6-gram Language model. Then, we bootstrapped the model with mono-phone training using $10,000$ samples. Subsequently, we performed tri-phone training passes using Δ and $\Delta\Delta$ features. We then carried out speaker-independent training using linear discriminative analysis (LDA) and maximum likelihood linear transform (MLLT). Finally, we performed speaker adaptive training with constrained maximum likelihood linear regression (fMLLR) and computed phone-level alignments for the corresponding speech segments.

G2P. Phonetisaurus (Hansen, 2020), implemented as a Python package, is employed to train a G2P model using a weighted finite state transducer framework. The training lexicon is the same as that used for the Kaldi phone recognizer.

3.3 Metrics

We evaluate the performance of our method using the following metrics, applied to the predictions of the G2P and phone recognition models as well as their optimal combination.

Phone Error Rate (PER). After aligning the predicted and ground-truth pronunciations, we calculate the sum of insertion, deletion, and substitution errors, and then divide this total by the length of the aligned sequence.

Cost-based Phone Error Rate (C-PER). This metric is similar to PER, where insertions and deletions receive equal penalties. However, substitutions are penalized according to the similarity between the involved phonemes. Specifically, substitutions between vowel pairs are penalized at half the rate compared to substitutions involving consonants.

Average Phone Rank of Truth (PhR) . This metric calculates the average rank of the correctly identified phone among all possible predictions at that position in the sequence. If no phone is correctly predicted, the rank for that position is arbitrarily set to a higher value. A lower value, approaching 0, indicates that the phone had the highest probability at that position and therefore was the top choice.

4 Results

Figure 1 illustrates the change in PER of the G2P model when used independently versus with the phone recognizer (a positive change indicates that the combination performed better than G2P) across $32,000$ utterances and 76 languages. Our method outperforms the standalone G2P model by 8.5% on average for all high-resource languages (languages with more than $3,000$ samples in the training set). For medium- and low-resource languages, our method also performs better on average.

For most high-resource languages, the G2P and phone recognition models perform similarly, but their combination yields better results. This demonstrates the effectiveness of our method (and by extension, recombination and weighting algorithms) in enhancing pronunciations.

However, our dataset includes a long tail of low-resource languages with fewer than 10 samples in the training set. For some of these languages, neither the G2P nor the phone recognition models perform well, resulting in sub-optimal performance of our method. To address this, we weight the outputs of the phone recognizer as described in Section 2.3. This approach prevents the degradation of the G2P output due to poor performance on speech segments. Additionally, this weighting helps in situations where the phone recognizer lacks confidence in its output due to noise or other challenging acoustic conditions, regardless of the language.

Tables 1, 2 and 3 display the ground-truth (GT) pronunciation of a specific word along with the predictions of the combination, G2P and phone recognition models as the 1st, 2nd and 3rd rows respectively.

	Word	**Lang**	**GT**	
	Document	Occitan	dukymen	
	Output	**PER**	**C-PER**	**PhR**
1	dukymɛn	14	0	1.0
2	dɔkymɛnt	38	14	2.12
3	tʁdukym ˈɛn	33	23	2.2

Table 1: Pronunciation of the word *Document* from the *Occitan* language with metrics.

Table 1 illustrates that a multilingual G2P model alone may not perform optimally for languages like Occitan, where the "o" in *Document* is phonetically represented as "u" instead of "ɔ". While this distinction may not be evident in the orthography, the phone recognizer identifies it in the spoken utterance and assigns a probability of 0.8 to "u", whereas the G2P model assigns a probability of only 0.5 to "ɔ". Additionally, the C-PER is 0, despite a substitution error at position 5 between "ɛ" and "e". This highlights that while the pronunciation may not be flawless, such low-cost errors are acceptable and sufficiently accurate, especially for low-resource languages.

In Table 2, the G2P model predicts the first phone of УАШЪО as "ʊ" instead of "w", despite the remainder of the pronunciation being correct. Similar to Table 1, the phone recognizer confidently assigns a probability of 0.7 to the phone "w", even though its other phones have probabilities of 0.4. This is indicated by the PhR score of 3.17, which shows that the correct phone has the 3rd-highest probability on average among all other

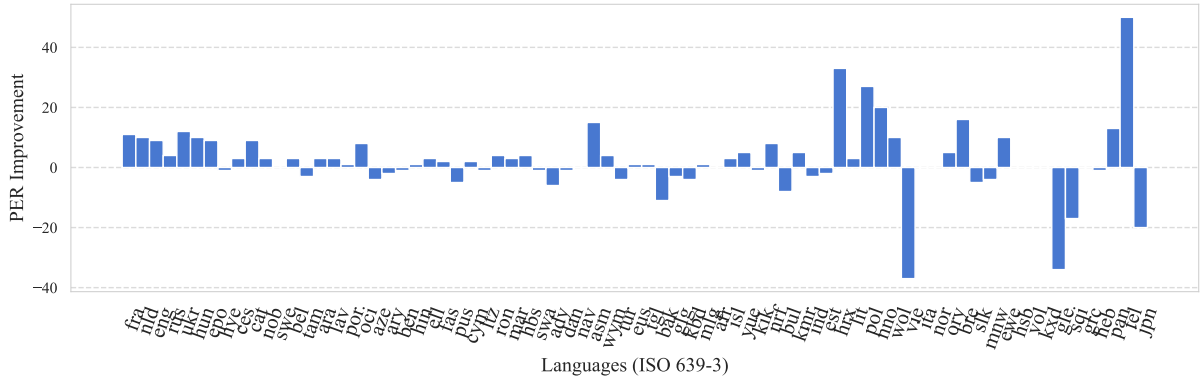

Figure 1: PER improvement compared to G2P after recombination across 76 languages, identified by their ISO-639-3 codes. Languages are sorted left to right based on their sample counts in the dataset.

Word		Lang	GT	
УАШЪО		Adyghe	waːʃʷa	
	Output	**PER**	**C-PER**	**PhR**
1	waʃʷa	25	1	0.17
2	ʊaːʃʷa	25	25	0.5
3	wax	80	61	3.17

Table 2: Pronunciation of the word УАШЪО from the *Adyghe* language with metrics.

phones across all positions in the sequence.

Word		Lang	GT	
ПАЎСТАННЕ		Belarusian	paʊ̯stanʲːɛ	
	Output	**PER**	**C-PER**	**PhR**
1	paʊ̯stanʲːɛ	0	0	0
2	paʊ̯stanʲːɛ	0	0	0
3	poʊ̯ustanʲːiɛ	44	14	1.56

Table 3: Pronunciation of the word ПАЎСТАННЕ from the *Belarusian* language with metrics.

In Table 3, the output from the phone recognizer does not influence the output from the G2P model. This occurs because all the phones generated by G2P have probabilities close to 1, whereas the phones that differ according to the acoustic model have much lower probabilities. Consequently, we reduce the weight of the phone recognizer, resulting in minimal incorporation of its phones in the final pronunciation. This situation exemplifies a scenario where the phone recognizer performs inadequately and thus should not diminish the performance of the G2P model.

5 Limitations

Given that our phone recognizer is a basic HMM-GMM model, we experience relatively high PER, especially for consonants. Although our alignment and weighting algorithm effectively ignores these erroneous phones when merging the output with that of the G2P model, a more advanced universal phone recognizer would be beneficial in cases where the G2P model also misidentifies consonants. Additionally, our dataset contains a long tail of languages with fewer than 10 samples in the training set, resulting in subpar few-shot performance for some low-resource languages. Employing neural models for both G2P and phone recognition could potentially enhance the overall performance of the method across most languages.

6 Conclusion

In this work, we introduce a method that leverages an acoustic phone recognition model to enhance G2P pronunciations. We created our dataset by scraping Wiktionary, collecting data for over 100 languages and 160,000 words. Our approach improves performance for high-resource languages, achieving an average PER reduction of 8.5%, and performs reasonably well for medium- and low-resource languages. This demonstrates that combining G2P and ASR outputs using effective alignment and weighting strategies can improve pronunciations, accommodating variations across languages and dialects. This method facilitates the creation of pronunciation dictionaries for a wide range of languages using only basic orthography and single-word spoken utterances.

References

Angelina A. Aquino, Joshua L. Tsang, Crisron Rudolf Lucas, and Franz A. de Leon. 2019. G2p and asr techniques for low-resource phonetic transcription of tagalog, cebuano, and hiligaynon. *2019 International Symposium on Multimedia and Communication Technology (ISMAC)*, pages 1–5.

Abhinav Garg, Jiyeon Kim, Sushil Khyalia, Chanwoo Kim, and Dhananjaya Gowda. 2024. Data-driven grapheme-to-phoneme representations for a lexicon-free text-to-speech. *arXiv preprint arXiv:2401.10465*.

Michael Hansen. 2020. Phonetisaurus: Python wrapper for the phonetisaurus grapheme to phoneme tool.

Bradley Hauer, Amir Ahmad Habibi, Yixing Luan, Arnob Mallik, and Grzegorz Kondrak. 2020. Low-resource G2P and P2G conversion with synthetic training data. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 117–122, Online. Association for Computational Linguistics.

Xinjian Li, Siddharth Dalmia, Juncheng Li, Matthew Lee, Patrick Littell, Jiali Yao, Antonios Anastasopoulos, David R Mortensen, Graham Neubig, Alan W Black, et al. 2020. Universal phone recognition with a multilingual allophone system. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8249–8253. IEEE.

Xinjian Li, Florian Metze, David Mortensen, Shinji Watanabe, and Alan Black. 2022. Zero-shot learning for grapheme to phoneme conversion with language ensemble. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2106–2115, Dublin, Ireland. Association for Computational Linguistics.

Saul B. Needleman and Christian D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453.

Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukáš Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlíček, Yanmin Qian, Petr Schwarz, Jan Silovský, Georg Stemmer, and Karel Vesel. 2011. The kaldi speech recognition toolkit. *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*.

Sam Ribeiro, Giulia Comini, and Jaime Lorenzo-Trueba. 2023. Improving grapheme-to-phoneme conversion by learning pronunciations from speech recordings. pages 999–1003.

James Route, Steven Hillis, Isak Czeresnia Etinger, Han Zhang, and Alan W Black. 2019. 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)*, pages 192–201.

Zach Ryan and Mans Hulden. 2020. Data augmentation for transformer-based G2P. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 184–188, Online. Association for Computational Linguistics.

Kathleen Siminyu, Xinjian Li, Antonios Anastasopoulos, David R. Mortensen, Michael R. Marlo, and Graham Neubig. 2021. Phoneme recognition through fine tuning of phonetic representations: a case study on luhya language varieties. In *Interspeech*.

Qiantong Xu, Alexei Baevski, and Michael Auli. 2021. Simple and effective zero-shot cross-lingual phoneme recognition. *arXiv preprint arXiv:2109.11680*.

Ling Zhang. 2018. Pyphone: A python package for phonetic distance metric.

Chendong Zhao, Jianzong Wang, Xiaoyang Qu, Haoqian Wang, and Jing Xiao. 2022. r-g2p: Evaluating and enhancing robustness of grapheme to phoneme conversion by controlled noise introducing and contextual information incorporation. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6197–6201. IEEE.

Jian Zhu, Cong Zhang, and David Jurgens. 2022. Byt5 model for massively multilingual grapheme-to-phoneme conversion. In *Interspeech*.