Multimodal, Multilingual Grapheme-to-Phoneme Conversion for Low-Resource Languages

James Route, Steven Hillis, Isak C. Etinger,[*] Han Zhang,[*] Alan Black
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA, USA
{jroute, shillis, ice, awb}@cs.cmu.edu, hanz3@andrew.cmu.edu

Abstract

Grapheme-to-phoneme conversion (g2p) is the task of predicting the pronunciation of words from their orthographic representation. Historically, g2p systems were transition- or rule-based, making generalization beyond a monolingual (high resource) domain impractical. Recently, neural architectures have enabled multilingual systems to generalize widely; however, all systems to date have been trained only on spelling-pronunciation pairs. We hypothesize that the sequences of IPA characters used to represent pronunciation do not capture its full nuance, especially when cleaned to facilitate machine learning. We leverage audio data as an auxiliary modality in a multi-task training process to learn a more optimal intermediate representation of source graphemes; this is the first multimodal model proposed for *multilingual* g2p. Our approach is highly effective: on our in-domain test set, our multimodal model reduces phoneme error rate to 2.46%, a more than 65% decrease compared to our implementation of a unimodal spelling-pronunciation model—which itself achieves state-of-the-art results on the Wiktionary test set. The advantages of the multimodal model generalize to wholly unseen languages, reducing phoneme error rate on our out-of-domain test set to 6.39% from the unimodal 8.21%, a more than 20% relative decrease. Furthermore, our training and test sets are composed primarily of low-resource languages, demonstrating that our multimodal approach remains useful when training data are constrained.

1 Introduction

Graphemic and phonemic representations of words are often no more than loosely related within languages and can be in direct contradiction between them. These inconsistencies introduce errors into any application of speech technology which has to convert between these two representations: namely text-to-speech and speech-recognition systems.

Very early grapheme to phoneme systems were monolingual and often restricted to English due to dataset availability (Weide, 1998; Kingsbury et al., 1997; Sejnowski, 1987). These early systems were designed to address the problem of intra-language discrepancies through rule based transition systems. These systems required painstaking tailoring to individual languages, and their performance was largely limited to that language's domain. Recent work has extended finite state automata constructed in this way for high resource languages to very similar low resource languages by applying distance metrics and linguistic expertise (Deri and Knight, 2016), but this approach is limited in application and performance.

Relieving some of the burden of technical expertise, statistical methods surpassed rule-based ones, with emphasis on joint sequence modeling (Chen, 2003; Bisani and Ney, 2008; Jiampojamarn et al., 2007). These methods improved performance, but they mandate explicit training alignments. This can be avoided by using neural attentional models, as in Toshniwal and Livescu (2016). Their work makes clear the parallel between this sequence prediction task and more traditional machine translation; this parallel inspires the model proposed in Peters et al. (2017), which, motivated by similarities in vocabularies, spellings, writing systems, and phonemic inventories between low and high resource languages, applies multilingual MT techniques to train a massively multilingual g2p system.

This application is effective, but it—like all work on this task before it—neglects perhaps the most rich source of information on pronunciation available: speech data. All existing grapheme to phoneme systems have been trained on spelling-

[*] Equal contribution

Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo), pages 192–201
Hong Kong, China, November 3, 2019. ©2019 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17

pronunciation data alone, neglecting the audio modality largely due to constraints imposed by available datasets. Suspecting that the preprocessed IPA sequences used to represent pronunciation encode it insufficiently, we propose to learn more optimal grapheme representations and thus make more accurate phoneme predictions by novelly leveraging an auxiliary audio modality as part of a multi-task training process.

2 Datasets

We discuss two datasets in this paper. We focus on the newer Wilderness dataset (Black, 2019), which is multilingual and contains paired text and speech data. We compare results of our multimodal model with all baselines on the Wilderness data. We also include the Wiktionary dataset (Deri and Knight, 2016), which consists of textual data only, because it has been commonly used in prior works on multilingual g2p systems. Wiktionary and Wilderness have incompatible IPA character sets which prevent us from training a model on Wilderness and testing with Wiktionary. We report baseline results only on Wiktionary to offer an approximate means of comparison between the two datasets.

2.1 Wiktionary

The Wiktionary dataset, introduced in Deri and Knight (2016), consists of single word spelling-pronunciation pairs scraped from the open-source multilingual dictionary maintained by Wikimedia. Entries are extracted from high resource language sites, which have instances for multiple languages. This heavily biases the distribution, with English, French, and German accounting for 51% of all pairs. Filtering for length, each Wiktionary pronunciation is mapped to Phoible phonemes after accounting for a phoneme distance metric original to this work. Following Peters et al. (2017), we use the cleaned pronunciations and randomly sample 10% of the corpus' training split to use for validation.

	Train	**Test**
Languages	311	507
Words	631,828	25,894
Scripts	42	45

Table 1: Corpus statics for Wiktionary dataset

2.2 Wilderness

We use the CMU Wilderness dataset[1], introduced in Black (2019), which contains of audio, aligned text, and word pronunciations for over 700 languages. The source material consists of versions of the New Testament, which speakers read in their own language. Pronunciations are generated from the audio by an HMM aligner and are transcribed in X-SAMPA (Wells, 1995), an extension of the Speech Assessment Methods Phonetic Alphabet (SAMPA). X-SAMPA was used to encode symbols of the International Phonetic Alphabet (IPA) into 7-bit ASCII before the advent of Unicode. We convert the X-SAMPA representations into true IPA characters.

We represent the audio data from the CMU Wilderness dataset as 39-dimensional MFCC (Mel Frequency Cepstral Coefficients) features (Sahidullah and Saha, 2012; Zheng et al., 2001; Ganchev et al., 2005; Ittichaichareon et al., 2012), a spectral-based parameter commonly used to vectorize audio data which represents the short-term power spectrum of an audio stream. The first 13 dimensions are the Mel frequency cepstral coefficients of the first 13 coefficients of the Fourier transform of the audio stream. The next 13 dimensions are the time-derivatives of those coefficients, and the last 13 are the double time-derivatives. The first 13 dimensions were calculated with the Librosa python package (McFee et al., 2015) method `librosa.feature.mfcc`. Other dimensions were calculated with the `librosa.feature.delta` method.

Directly comparing those dimensions has no physical meaning, so we normalize those features as

$$f_{i,u} \to \frac{f_{i,u} - \min_{u' \in U}(f_{i,u'})}{\max_{u' \in U}(f_{i,u'}) - \min_{u' \in U}(f_{i,u'})} \cdot 0.95^{i}$$

where U are the utterances and $i \in \{1..39\}$. We used a sliding window of 25ms with 10ms stride. MFCCs are not the only way to vectorize audio data, and they are not necessarily the best, but they are a sufficient representation to facilitate our experiments.

The Wilderness dataset ranks the quality of alignment for a language on the basis of the reconstruction score over a held out test set for a

[1] https://github.com/festvox/datasets-CMU_Wilderness

grapheme-based speech synthesizer trained on the remaining language data. Reconstruction score is measured in Mel Cepstral Distortion (MCD) (Toda et al., 2007), a scaled Euclidean distortion metric for comparing synthesized utterances to true ones. Lower is better. For this dataset, when MCD scores are less than 7, the synthesized outputs are usually intelligible, and when they are less than 6, the outputs are easily understood. We chose languages with MCD scores less than 6 for our experiments; see Table 2 for more on these languages.

Resources constrain our experiments to a total of 20 languages out of the available 700. Ten of those languages are used for training, development, and in-domain (ID) experiments; the remaining ten are used for out-of-domain (OoD) experiments. Fifteen different language families are represented. For training and validation, 1000 and 100 utterances are used for each ID language respectively. Note that all languages trained on are themselves low resource—a major departure from previous work. For more details on each of the languages, as well as expansions of the abbreviations, see Table 9 at the end of the paper.

In-Domain		Out-of-Domain	
Language	**MCD**	**Language**	**MCD**
SHIRBD	4.96	MYYWBT	5.80
COKWBT	5.37	SABWBT	5.80
LTNNVV	5.82	LONBSM	5.83
XMMLAI	5.20	NHYTBL	5.92
TS1BSM	5.24	ALJOMF	5.93
GAGIBT	5.26	BFABSS	5.20
KNETBL	5.68	HUBWBT	5.98
TPPTBL	5.72	TWBOMF	5.98
HAUCLV	5.74	ENXBSP	5.99
ESSWYI	5.79	POHPOC	5.29

Table 2: MCD scores for Wilderness languages[2]

	Verses	**Words**	**Length (min)**
Train	10,000	139,796	1060
Dev	1,000	13,937	106
ID Test	1,000	13,815	104
OoD Test	1,000	15,418	107

Table 3: Statistics for Wilderness-based corpus

3 Baseline

Multilingual neural machine translation techniques have recently been applied to the g2p problem (Peters et al., 2017) to accommodate the lack of data for low-resource languages. With many low-resource languages sharing similar writing systems with high-resource languages, orthographic representations of words in any language are mapped to the corresponding phonemic representations in a multisource sequence-to-sequence model. We reproduce their architecture as our performance baseline using OpenNMT (Klein et al., 2017) on the Wiktionary and Wilderness datasets. Briefly, the source graphemes (augmented with language tags) and target phonemes are first processed as character-based embedding sequences. The model uses an encoder-decoder structure and the global attention layer proposed by Luong et al. (2015). We selected this model because it achieved state-of-the-art results on Wiktionary and represents a strong baseline for sequence-to-sequence model performance on g2p.

Two common evaluation metrics for g2p models are Phoneme Error Rate (PER) and Word Error Rate (WER). Phoneme Error Rate represents the Levenshtein distance over the target and predicted phonemes, normalized by the target sequence length. Word Error Rate represents the percentage of predicted phoneme words which do not exactly match their target phoneme words. For our experiments, we extend the concept of Word Error Rate to a metric that we term Sequence Error Rate (SER), which measures the percentage of incorrectly predicted phoneme sequences. This alteration is necessary because Wilderness utterances consist of multiple words, and the phoneme sequences are not segmented by word. WER

Examples	**SER**	**PER**
Example #1: ‘An example’		
Predicted: [ə n ɪ g z ɑɛ m p ə l]		
Gold: [ə n ɪ g z ɑɛ m p ə l]	0.00	0.00
Example #2: ‘And a second’		
Predicted: [ɑɛ n d ə s ə k ə n]		
Gold: [ɑɛ n d ə s ɛ k ə n d]	100.00	20.00
Total Scores	**50.00**	**10.00**

Table 4: Examples for SER and PER calculations

and SER are functionally identical for Wiktionary, which comprises single-word grapheme-phoneme pairs.

We note that other multilingual g2p systems exist, such as Deri and Knight (2016) and Epitran (Mortensen et al., 2018), although we do not include these systems in the results. The Peters et al. (2017) model previously outperformed the Deri and Knight (2016) system on Wiktionary by a significant margin, and Epitran is a rule-based system that does not support the vast majority of the low-resource languages we use.

4 Multimodal Approach

Multimodal models have been frequently explored for feature mining (e.g., text, image, audio). Multimodal learning commonly focuses on three areas: fusion of information, cross-modality learning, and shared representation mining (Ngiam et al., 2011). A deep multimodal learning method for automatic speech recognition was designed (Mroueh et al., 2015) to fuse both audio and visual modalities. In this case, the latent audio and video features were concatenated and used jointly for the prediction of speech. Recent work on multimodal sentiment analysis (Pham et al. (2018b) and Pham et al. (2018a)) demonstrated that an auxiliary task of translating from a source to one or more target modalities results in a joint representation that captures interactions between the modalities. We base our model on this approach and apply it to a sequence prediction task on multilingual data.

We develop a recurrent sequence-to-sequence model with attention that learns a robust joint representation for graphemes and speech data across multiple languages, which is then used to predict a phoneme sequence given only graphemes[3]. We hypothesize that the inclusion of the speech modality will enable the model to learn a better multilingual representation than text alone, and that a multimodal representation will generalize to unseen languages better than a text-only model. A key feature of our model is that speech data are only required for training; during inference the model only uses grapheme inputs.

Our model is an LSTM sequence-to-sequence model with a single encoder and two decoders (Figure 1). One decoder predicts MFCC coefficients from graphemes (auxiliary task) and the other predicts IPA character sequences (primary task). Each task corresponds to a separate loss function.

During training, three sequences are available to the model: grapheme characters X_t, speech MFCCs S_t, and phoneme characters Y_t. The encoder is a biLSTM, with the output based on the previous hidden state and the current grapheme character in the input sequence:

$$h_{e,t} = LSTM(h_{e,t-1}, X_t) \tag{1}$$

The decoders use the same basic architecture with minor differences. The MFCC decoder consists of an LSTM whose input is a concatenation of the previous MFCC output $\hat{S}_{t-1}$ and previous attention context $a_{s,t-1}$. The LSTM hidden state is fed to an MLP to produce the attention query $q_{s,t}$. The sequence of encoder hidden states is passed through two separate MLPs to obtain attention keys and values K and V. A dot-product global attention mechanism from Vaswani et al. (2017) follows. The resulting attention context $a_{s,t}$ is then projected by MLP down to a 39-dimension MFCC vector $\hat{S}_t$.

$$h_{s,t} = LSTM(h_{s,t-1}, [\hat{S}_{t-1}; a_{s,t-1}]) \tag{2}$$
$$q_{s,t} = MLP(h_{s,t}) \tag{3}$$
$$K, V = MLP(h_e), MLP(h_e) \tag{4}$$
$$a_{s,t} = \sum softmax(q_{s,t}K^T)V \tag{5}$$
$$\hat{S}_t = MLP(a_{s,t}) \tag{6}$$

The phoneme decoder follows the same design except that its output $\hat{Y}_t$ is a distribution over the IPA character vocabulary. No parameters are shared between the decoders.

$$h_{y,t} = LSTM(h_{y,t-1}, [\hat{Y}_{t-1}; a_{y,t-1}]) \tag{7}$$
$$q_{y,t} = MLP(h_{y,t}) \tag{8}$$
$$K, V = MLP(h_e), MLP(h_e) \tag{9}$$
$$a_{y,t} = \sum softmax(q_{y,t}K^T)V \tag{10}$$
$$\hat{Y}_t = softmax(MLP(a_{y,t})) \tag{11}$$

Model parameters are learned during training by empirical risk minimization over input graphemes and paired MFCC vectors and phoneme characters $\{X_t, S_t, Y_t\}$, across all languages in the training set. A separate loss is

[3] Model code is available at https://github.com/jamesrt95/Multimodal-Multilingual-G2P

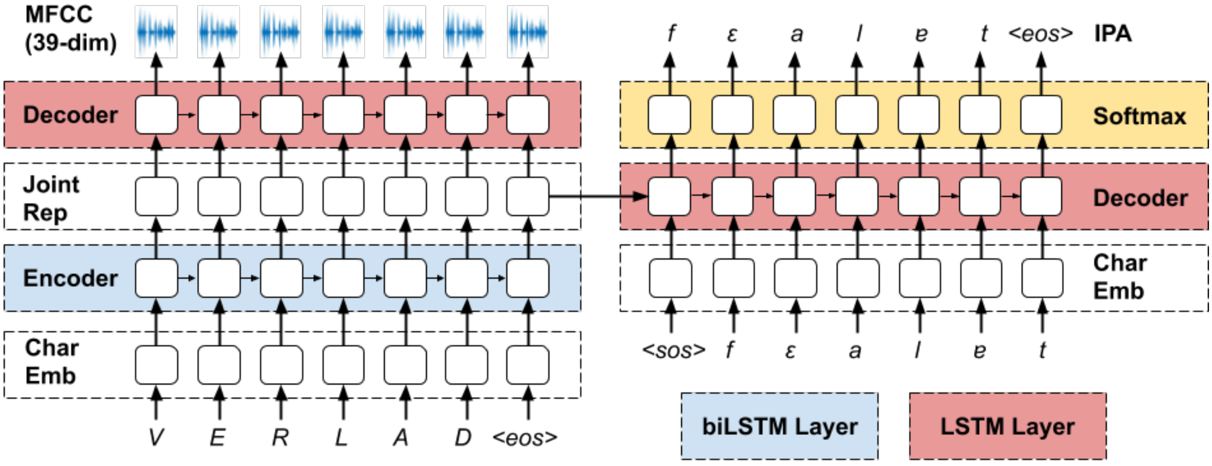

Figure 1: Diagram of Multimodal g2p Model

calculated from the output of each decoder. We use mean-squared error as loss function ℓ_S for the MFCC output and cross-entropy as loss function ℓ_Y for the IPA output. The entire network is trained end-to-end using a weighted sum of the two losses where λ is a hyperparameter.

$$\mathcal{L}_S = \mathbb{E}[\ell_S(\hat{S}, S)] \tag{12}$$

$$\mathcal{L}_Y = \mathbb{E}[\ell_Y(\hat{Y}, Y)] \tag{13}$$

$$\mathcal{L} = \mathcal{L}_Y + \lambda \mathcal{L}_S \tag{14}$$

The encoder learns a joint embedding that models interactions between the grapheme and speech modalities. This is accomplished via gradient descent, as parameter updates for the encoder and MFCC decoder are dependent on the grapheme and speech sequences. The model is then able to infer speech data when given only grapheme inputs. At test time, the model is given only grapheme inputs and the MFCC output is ignored. We then perform beam search over the IPA decoder output to generate the final predicted sequence.

5 Experiments

First, we implemented the Peters et al. (2017) baseline model separately on the Wiktionary and Wilderness datasets. We then trained two variants of our sequence-to-sequence model on the in-domain Wilderness data to compare the effects of the multimodal representation. The first variant was multimodal (referred to as the *Multimodal Model*). The parameters for this model are given in Table 5. The second variant was unimodal (referred to as the *Unimodal Model*) and treated as an additional baseline. During training for this model, the loss term for the MFCC decoder was ignored, so learned parameters were based solely on the grapheme inputs and phoneme outputs. The unimodal model also used the parameters given in the table, except the MSE loss weight was zero.

We selected layer size parameters for the both models that were similar to Peters et al. (2017) so that differences in performance could be more clearly attributed to the multimodal training process. We set teacher forcing to 90% so that the model's inferences were not completely dependent on seeing correct labels at each time step. For the multimodal model, we weighted the MSE loss from MFCC prediction at 0.1 because it was an auxiliary objective, and the model's learning process tended to be more stable when weighted lower than the primary cross-entropy objective. We used results from the dev set to choose this value. We also averaged the MFCC values over 10 consecutive frames; this helped the model to learn more quickly and allowed for larger batch sizes.

The models were each trained on all languages in the training set (i.e., each model was trained to be multilingual). The training set was shuffled so that there was no systematic ordering of languages during training. The models were then evaluated separately on the in-domain and out-of-domain test sets.

6 Results

The results on the Wilderness datasets are presented in Table 7. We are only able to provide a direct comparison between the performance of the baseline model and of our models on the Wilder-

Enc. type	biLSTM
Dec. type	LSTM
Enc. & dec. layers	1
Attention type	Dot
Hidden layer size	128
Source emb. size	64
Target emb. size	64
Batch size	16
Optimizer	Adam
Learning rate	1e-3
Teacher forcing rate	0.9
MSE loss weight (λ)	0.1
Training epochs	14
Beam size	10

Table 5: Multimodal Model Parameters

ness data: the Wiktionary dataset uses a different and incompatible IPA character vocabulary, which prevents us from training a model on Wilderness and testing on Wiktionary. We report baseline results on Wiktionary to offer an approximate means of comparison between Wiktionary (an established dataset) and Wilderness, which is newly created.

Model	SER	PER
Peters et al. Baseline Model	43.23	37.85
Our Impl. of Peters et al.	**37.87**	**26.00**

Table 6: Comparison of Models on Wiktionary Dataset

For the Wilderness data, we report results on two test sets (In-Domain and Out-of-Domain) to illustrate generalization to unseen languages. The ID test set consists of 100 unseen utterances from each of the same 10 languages used in training, whereas the OoD test set consists of 100 utterances each from 10 languages that were not used in training.

7 Discussion

Although we were pleasantly surprised to see the performance of our implementation of the baseline system from Peters et al. (2017) increase so drastically from the results they report on the Wiktionary dataset, we take little credit for this result; it can perhaps be attributed to improvements made to the OpenNMT platform over the past two years, but we replicated their experiments as faithfully as we were able.

Model	SER	PER
In-Domain Test Results		
Baseline Model	46.90	25.06
Unimodal Model	31.20	7.05
Multimodal Model	**9.50**	**2.46**
Out-of-Domain Test Results		
Baseline Model	84.20	43.16
Unimodal Model	49.30	8.21
Multimodal Model	**38.10**	**6.39**

Table 7: Comparison of Models on Wilderness Dataset

On the other hand, we are happy to take credit for the relative performances of our models on the Wilderness dataset. We attribute much of the improvement to a more expressive attention mechanism and to improved hyperparameter tuning, as our underlying model used similar layer sizes to the baseline.

Our hypothesis about the value of including audio data during training is heartily confirmed by the performance of our multimodal model: the multimodal model performs better for both metrics not only on in-domain languages but also on very different, wholly unseen languages. Our multimodal approach to the task of grapheme to phoneme conversion improves both performance and generalization.

We note the multimodal model's SER is much worse on out-of-domain languages than in-domain ones, albeit still surpassing the unimodal model's

In-Domain			Out-of-Domain		
Lang	**PER**	**SER**	**Lang**	**PER**	**SER**
SHI	5.24	14.00	MYY	14.10	100.00
COK	3.07	14.00	SAB	6.59	50.00
LTN	1.92	8.00	LON	1.51	14.00
XMM	1.91	8.00	NHY	18.60	22.00
TS1	1.62	6.00	ALJ	2.60	4.00
GAG	2.71	7.00	BFA	7.41	90.00
KNE	1.29	3.00	HUB	1.60	5.00
TPP	5.03	20.00	TWB	2.86	7.00
HAU	0.56	7.00	ENX	17.00	71.00
ESS	0.23	8.00	POH	3.36	18.00

Table 8: Multimodal Model Error Rates by Language

In-Domain			Out-of-Domain		
Code	**Name**	**Family**	**Code**	**Name**	**Family**
SHIRBD	Shilha	Afro-Asiatic	MYYWBT	Macuna	Tucanoan
COKWBT	Cora, Santa Teresa	Uto-Aztecan	SABWBT	Buglere	Chibchan
LTNNVV	Latin	Indo-European	LONBSM	Elhomwe	Niger-Congo
XMMLAI	Manadonese Malay	Austronesian/Indo-Euro.	NHYTBL	Nahuatl	Uto-Aztecan
TS1BSM	Tsonga	Niger-Congo	ALJOMF	Alangan	Austronesian
GAGIBT	Gagauz	Turkic	BFABSS	Bari	Nilo-Saharan
KNETBL	Kankanaey	Austronesian	HUBWBT	Huambisa	Jivaroan
TPPTBL	Tepehua	Totonacan	TWBOMF	Tawbuid	Austronesian
HAUCLV	Hausa	Afro-Asiatic	ENXBSP	Enxet	Mascoyan
ESSWYI	Yupik	Eskimo-Aleut	POHPOC	Pokomchi	Mayan

Table 9: More Information on Wilderness languages

(Table 8). The out-of-domain languages contain characters that are out of vocabulary (OOV) from the training set, and in most cases OOV characters comprise 15-20% of the input sequence. One mistake in the output results in the entire sequence being scored incorrect for SER, so even small PER increases can lead to large swings in SER. In particular, the large increase in SER is primarily due to four languages in the Out-of-Domain test set. In the case of Macuna (MYY, 100 SER), the IPA character ʉ appears in nearly every utterance but never occurs in the training set, so the model is unable to predict it. Bari (BFA, 90.0 SER) is similar, where ŋ is highly common but never appears in the training set. Enxet (ENX, 71.0 SER) and Buglere (SAB, 50.0 SER) both frequently contain ɲ, which occurs only once in the training set.

We also note that our reimplementation of the Peters et al. (2017) baseline produces a lower Sequence Error Rate on the single-word utterances in the Wiktionary dataset than on the multi-word utterances in the Wilderness sets. Longer sequence pairs result in more opportunities for a model to make a mistake. This effect is acute for the sequence-level error, but even for PER, an incorrect output at one timestep may lead to cascading mistakes at future timesteps. The comparable PER scores on the Wiktionary and In-Domain Wilderness set suggest that the datasets are comparable in difficulty. Although we are unable to directly measure the multimodal model's performance on Wiktionary, its substantial improvements on a comparable task convince us of its efficacy.

8 Future Work

With recent advancements in language embeddings, we identify significant potential for improving the generalization of the model to unseen languages. Including language tags was shown to be beneficial in previous work, and we predict that exchanging the three-character tag for a high-dimensional embedding to capture taxonomic relationships between languages would only magnify the effect. Similarly, we have demonstrated the advantages of incorporating audio data during training, but MFCCs are not necessarily the most effective method of vectorizing that audio data. It would be interesting to investigate the effects of using other techniques, such as those in Haque et al. (2019) and Chung and Glass (2018), for generating high-dimensional representations of audio data.

We trained our model on approximately 0.1% of the data included in the Wilderness dataset, leaving tremendous opportunity for further learning. The incorporation of more training data is likely to improve results on its own, but it may also facilitate the use of a Transformer encoder-decoder model (Vaswani et al., 2017), which we know to require larger datasets than the LSTM variants.

We are very interested in experimenting with graphemes encoded in non-Roman scripts. This capacity is one of the most compelling facets of the Peters et al. (2017) model, but we were unable to explore it with our multimodal model: the New Testament text is almost always Romanized in the Wilderness data. We were furthermore unable to effectively evaluate our multimodal model on the

Wiktionary data after training on the Wilderness, as the IPA character space over the Wilderness dataset is much smaller than that of the Wiktionary dataset. In the future, we would like to reconcile these differences, both in order to evaluate our multimodal model on the Wiktionary test set and to explore its performance over widely varying scripts.

Acknowledgments

We thank Bhiksha Raj for helpful comments on our development of the multimodal model. We also appreciate the suggestions and comments from the anonymous reviewers.

References

Maximilian Bisani and Hermann Ney. 2008. Joint-sequence models for grapheme-to-phoneme conversion. *Speech communication*, 50(5):434–451.

Alan W. Black. 2019. Cmu wilderness multilingual speech dataset. *ICASSP*.

Stanley F Chen. 2003. Conditional and joint models for grapheme-to-phoneme conversion. In *Eighth European Conference on Speech Communication and Technology*.

Yu-An Chung and James Glass. 2018. Speech2vec: A sequence-to-sequence framework for learning word embeddings from speech. *arXiv preprint arXiv:1803.08976*.

Aliya Deri and Kevin Knight. 2016. Grapheme-to-phoneme models for (almost) any language. In *ACL*.

Todor Ganchev, Nikos Fakotakis, and George Kokkinakis. 2005. Comparative evaluation of various mfcc implementations on the speaker verification task. In *Proceedings of the SPECOM*, volume 1, pages 191–194.

Albert Haque, Michelle Guo, Prateek Verma, and Li Fei-Fei. 2019. Audio-linguistic embeddings for spoken sentences. *arXiv preprint arXiv:1902.07817*.

Chadawan Ittichaichareon, Siwat Suksri, and Thaweesak Yingthawornsuk. 2012. Speech recognition using mfcc. In *International Conference on Computer Graphics, Simulation and Modeling (ICGSM'2012) July*, pages 28–29.

Sittichai Jiampojamarn, Grzegorz Kondrak, and Tarek Sherif. 2007. Applying many-to-many alignments and hidden markov models to letter-to-phoneme conversion. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 372–379.

Paul Kingsbury, Stephanie Strassel, Cynthia McLemore, and Robert MacIntyre. 1997. Callhome american english lexicon (pronlex). *Linguistic Data Consortium, Philadelphia*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proc. ACL*.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. *CoRR*, abs/1508.04025.

Brian McFee, Colin Raffel, Dawen Liang, Daniel P.W. Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. 2015. librosa: Audio and Music Signal Analysis in Python. In *Proceedings of the 14th Python in Science Conference*, pages 18 – 24.

David R. Mortensen, Siddharth Dalmia, and Patrick Littell. 2018. Epitran: Precision G2P for many languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).

Y. Mroueh, E. Marcheret, and V. Goel. 2015. Deep multimodal learning for audio-visual speech recognition. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2130–2134.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. 2011. Multimodal deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pages 689–696, USA. Omnipress.

Ben Peters, Jon Dehdari, and Josef van Genabith. 2017. Massively multilingual neural grapheme-to-phoneme conversion. *CoRR*, abs/1708.01464.

Hai Pham, Paul Pu Liang, Thomas Manzini, Louis-Philippe Morency, and Barnabas Poczos. 2018a. Found in translation: Learning robust joint representations by cyclic translations between modalities. *arXiv preprint arXiv:1812.07809*.

Hai Pham, Thomas Manzini, Paul Pu Liang, and Barnabas Poczos. 2018b. Seq2seq2sentiment: Multimodal sequence to sequence models for sentiment analysis. *arXiv preprint arXiv:1807.03915*.

Md Sahidullah and Goutam Saha. 2012. Design, analysis and experimental evaluation of block based transformation in mfcc computation for speaker recognition. *Speech Communication*, 54(4):543–565.

Terry Sejnowski. 1987. Net talk: A parallel network that learns to read aloud. *Complex Systems*, 1:145–168.

Tomoki. Toda, Alan. Black, and Keiichi. Tokuda. 2007. Voice converstion based on maximum-likelihood estimation of speech parameter trajectory. *IEEE Transaction of Audio, Speech and Language Processing*, 15(8):2222–2236.

Shubham Toshniwal and Karen Livescu. 2016. Jointly learning to align and convert graphemes to phonemes with neural attention models. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 76–82. IEEE.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Robert L Weide. 1998. The cmu pronouncing dictionary. *URL: http://www. speech. cs. cmu. edu/cgibin/cmudict*.

John C Wells. 1995. Computer-coding the ipa: a proposed extension of sampa. *Revised draft*, 4(28):1995.

Fang Zheng, Guoliang Zhang, and Zhanjiang Song. 2001. Comparison of different implementations of mfcc. *Journal of Computer science and Technology*, 16(6):582–589.