INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Phoneme Recognition through Fine Tuning of Phonetic Representations: a Case Study on Luhya Language Varieties

Kathleen Siminyu[*], *Xinjian Li*[^], *Antonios Anastasopoulos*[%],
David Mortensen[^], *Michael R. Marlo*[#], *Graham Neubig*[^]

[*]Georgia Institute of Technology, USA
[^]Carnegie Mellon University, USA
[%]George Mason University, USA
[#]University of Missouri, USA

ksiminyu3@gatech.edu

Abstract

Models pre-trained on multiple languages have shown significant promise for improving speech recognition, particularly for low-resource languages. In this work, we focus on phoneme recognition using Allosaurus, a method for multilingual recognition based on phonetic annotation, which incorporates phonological knowledge through a language-dependent allophone layer that associates a universal narrow phone-set with the phonemes that appear in each language. To evaluate in a challenging real-world scenario, we curate phone recognition datasets for Bukusu[1] and Saamia[2], two varieties of the Luhya language cluster of western Kenya and eastern Uganda. To our knowledge, these datasets are the first of their kind. We carry out similar experiments on the dataset of an endangered Tangkhulic language, East Tusom, a Tibeto-Burman language variety spoken mostly in India. We explore both zero-shot and few-shot recognition by fine-tuning using datasets of varying sizes (10 to 1000 utterances). We find that fine-tuning of Allosaurus, even with just 100 utterances, leads to significant improvements in phone error rates.[3]

Index Terms: multilingual speech recognition, low-resource languages, phonology

1. Introduction

While speech recognition has made great strides, ASR over languages with very little transcribed text is still a daunting task. One of the most promising directions in low-resource speech processing involves pre-training models, both to improve low-resource ASR itself and to use ASR as an auxiliary task to improve results in other low-resource task settings. For example, Wiesner et al. [1] explore pre-training by back-translation for end-to-end ASR, while Stoian et al. [2], Bansal et al. [3] use ASR as an auxiliary task to improve results for low-resource speech-to-text translation. Recently, Baevski et al. [4] introduced wav2vec 2.0, which can be used to achieve good ASR performance by fine-tuning on as little as 10 minutes of transcribed audio. Further, pre-training has been combined with multi-lingual learning; Wang et al. [5] improve cross-lingual transfer learning for ASR with speech translation, while Hsu et al. [6] formulate ASR for different languages as different tasks and use a model agnostic meta-learning algorithm (MAML) to learn the various initialization parameters.

Nonetheless, effective multilingual learning of acoustic and language models is difficult, given the underlying differences between languages in pronunciation and lexicon respectively. To help alleviate issues caused by pronunciation differences, Li et al. [7] have recently proposed a model (Allosaurus; details in §2), that calculates a language-universal phone distribution using a standard ASR encoder, then converts it to a language-specific phoneme distribution. *Phonemes* are sounds that support lexical contrasts in a *particular* language; *phones* are the sounds that are physically spoken (which are largely language independent); and *allophones* are the set of phones that correspond to a particular phoneme. This method showed promise both for recognition on the datasets on which the model was pre-trained, and also for *zero-shot* adaptation to new languages, where it was tested on languages for which no training data was available.

However, in realistic low-resource settings, it is common to have a *small amount* of training data in the language to which we would like to adapt. Our work examines the question: "given a pre-trained acoustic model and phone recognizer based on universal phone representations, how quickly can it be adapted to perform reasonable phone recognition in a new language?" To examine this question, we perform both zero-shot recognition and fine-tuning using datasets of sizes varying between 10 and 1000 utterances. As an important second contribution, we curate phone recognition datasets for Bukusu and Saamia, two varieties of Luhya, a cluster of Bantu languages largely spoken in Kenya and Uganda; to our knowledge these are the first datasets of their type. Using these datasets we perform fine-tuning experiments and find both respectable zero-shot results, and rapid improvements with very small amounts of adaptation, demonstrating the utility of fine-tuning of universal phonetic representations as a method for building very low-resource ASR models. We also carry out similar experiments on the dataset of an endangered Tangkhulic language, East Tusom, a Tibeto-Burman language variety spoken mostly in India.

Briefly summarized, with this work we:

- provide a phoneme recognition benchmark on Bukusu and Saamia, two under-resourced Luhya language varieties, as well as East Tusom, an endangered Tibeto-Burman language variety,
- show that fine-tuning of Allosaurus can lead to phone error rate (PER) reductions of more than 40%, and
- show that few-shot fine-tuning using only 100 transcribed utterances can lead to up to 59% PER reductions

2. Methodology

2.1. Allosaurus Layer

Allosaurus, (Figure 1; [7]), comprises a language independent encoder and phone predictor, and a language dependent allophone layer and a loss function associated with each lan-

[1] Bukusu data.
[2] Saamia data.
[3] Fine-tuning code on github.

http://dx.doi.org/10.21437/Interspeech.2021-1434

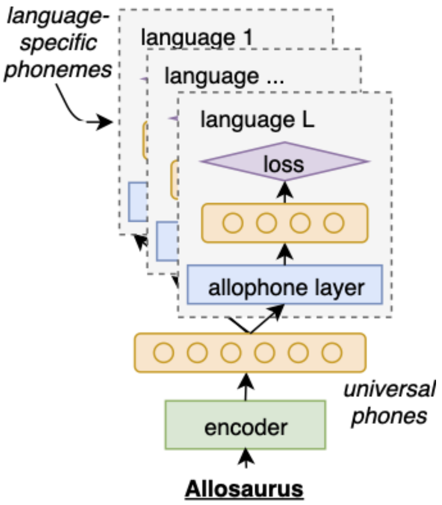

Figure 1: *Allosaurus (a multilingual pre-trained phoneme recognition model) predicts over a shared phone inventory, then maps into language-specific phonemes with an allophone layer.*

guage. This combination makes Allosaurus appropriate for multilingual phoneme recognition, because it allows it to handle phonemes while modeling the underlying phones, unlike other traditional multilingual models. The encoder first produces the distribution $h \in \mathbb{R}^{|P_{\text{uni}}|}$ over the universal phone inventory P_{uni}, then the allophone layer transforms h into phoneme distribution $g^i \in \mathbb{R}^{|Q_i|}$ of each language. The allophone layer uses a trainable allophone matrix $W^i \in \mathbb{R}^{|Q_i| \times |P_{\text{uni}}|}$ to describe allophones in a way similar to the signature matrix $S^i = \{0,1\}^{|Q_i| \times |P_{\text{uni}}|}$ which describes the association of phones and phonemes in each language L_i. The allophone matrix W^i is first initialized with S^i, and is allowed to be optimized during the training process. An L2 penalty is added to penalize divergence from the original signature matrix S^i. The allophone layer computes its logit distribution g^i by finding the most likely allophone realization in P_{uni} with maxpooling.

$$g^i_j = \max(\{w^i_{j,k} \cdot h_k ; 1 \leq k \leq |P_{\text{uni}}|\}), \tag{1}$$

where $g^i_j \in \mathbb{R}$ is the logit of j-th phoneme in g^i for language L_i, $w^i_{j,k} \in \mathbb{R}$ is the (j,k) cell of the allophone matrix W^i, $h_k \in \mathbb{R}$ is the logit of k-th phone in h. Intuitively, if the j-th phoneme has the k-th phone as an allophone, $w^i_{j,k}$ would be near 1, otherwise $w^i_{j,k}$ would be near 0. Therefore, the phoneme logit of g^i_j is decided by the largest allophone logit h_k. The phoneme distribution g^i is further fed into the loss function. This method for phoneme prediction can be used with any underlying multilingual ASR system. Here the parameters are optimized by minimizing CTC loss, Graves et al. [8], for all training languages, with the addition of regularization of the allophone layer controlled by hyperparameter α.

$$\mathcal{L} = \sum_{1 \leq i \leq |L|} (\mathcal{L}^i_{ctc} + \alpha \|W^i - S^i\|^2_2). \tag{2}$$

2.2. Universal Phone Recognition

Not only does the allophone layer abstract away from the language-specific phonemes, which contributes to the improvement in the multilingual acoustic modeling, the model also gives us the capability to predict universal phones themselves. By applying a greedy decoding strategy over the phone distribution h, we can obtain a phone sequence in which all phones P_{uni} in the training languages are candidates. Combined with large training language sets, the universal inventory covers most common narrow phones appearing in most languages.

Furthermore, this protocol can take into account phone inventories that have already been created for many languages in the world by linguists. For example, PHOIBLE [9] is a database of phone inventories for more than 2000 languages and dialects, allowing our model to be applied to these languages with some degree of accuracy. If the phone inventory for language L_i is P_i, we can restrict the decoder to only produce phones in $P_i \cap P_{\text{uni}}$ by filtering out other phones. When the universal inventory P_{uni} covers most frequent phones in the world, we could expect that $P_i \approx P_i \cap P_{\text{uni}}$.

2.3. Fine-tuning of Universal Phonetic Representations

In Li et al. [7], the universal phonetic representations were tested in a *zero-shot* setting, where the model was used as-is on languages not occurring in the training data. However, in many cases a small amount of labeled data *does* exist in the target language we wish to recognize. In this paper, we further propose and demonstrate results for *fine-tuning of universal phone representations* as an efficient and expedient way to utilize this data. Depending on the type of transcriptions that are available for the new language (phonetic or phonemic), we can fine-tune either using only the shared universal phone output layer, or create a new allophone layer for the new language and use a language-specific loss. In this work, we fine-tune the encoder which produces a distribution over the universal phone inventory.

We fine-tune with a small learning rate, 0.01, to avoid catastrophic forgetting [10] for a maximum of 250 epochs, and choose the best performing model based on a small validation set. We use simple stochastic gradient descent as the optimization algorithm.

3. Phoneme Recognition Benchmark for Luhya Language Varieties

In this section we describe the background and the data sources for our Luhya phoneme recognition benchmarks and the data source for the East Tusom benchmark.

3.1. Background

Bukusu and Saamia are members of the Luhya cluster of ∼25 Bantu languages. Though Luhya languages are spoken in both western Kenya and eastern Uganda, “Luhya” typically refers to the ∼19 Kenyan communities that were politically united in the first half of the 20th century.[4] Luhya is used as a label of both ethnicity and language for this group of culturally, politically, and linguistically heterogeneous communities.

Together, Kenyan Luhya communities number around 6.8 million people and are the second-largest ethnic group in Kenya.[5] Bukusu is the largest Kenyan Luhya community with more than 1 million members; there are more than 85,000 members of the Saamia community in Kenya, and another 279,000 in Uganda.[6]

Marlo et al. [12] provide a recent classification of Luhya languages, showing that despite their diversity, the Luhya language varieties are indeed more closely related to one another

[4] See MacArthur [11] for a recent statement about modern Luhya history.

[5] Details here: 2019 Kenya Census. Retrieved March 2020.

[6] Details here: 2002 Uganda Census. Retrieved October 2020.

than they are to their nearest Bantu neighbors to the west and south. All modern linguistic studies such as Mutonyi [13] and Botne et al. [14] on Saamia focus on the languages of individual sub-communities, who are each recognized to have their own distinct speech form. Whether each language variety should be considered as a “dialect” or a “language” is a matter of debate, but see Angogo [15] for a study of mutual intelligibility among Kenyan Luhya varieties.

Early missionary activity introduced competing orthographies for Luhya languages. Despite some standardization efforts, there are not universally agreed upon orthographic conventions for writing all Luhya varieties, which are linguistically diverse and have different sound systems.

3.2. Access to Data

An inescapable aspect of working in low-resource languages is identifying suitable data.

The CMU Wilderness Multilingual Speech Dataset covers over 700 languages providing audio, aligned text and word pronunciations. Saamia, one of the Luhya varieties, is available in the CMU Wilderness dataset. On average each language has around 20 hours of sentence-length transcriptions. Data is mined from reading of the New Testament from Bible.is.[7]

The co-author Michael Marlo, who has ongoing documentation projects on several Luhya language varieties, also provided access to recordings from his unpublished dictionary of Bukusu. Most recordings include two or more pronunciations of the same word. Similarly, the East Tusom dataset, composed of transcribed audio from a comparative wordlist was made available by the co-author David Mortensen through his documentation work of the language.

3.3. Data Preparation

Saamia Data The CMU Wilderness GitHub repository[8] contains code to download data directly from the Bible.is website as they do not have permission to redistribute this data. It contains 18.2 hours of Saamia data. Alignments, short waveforms plus transcripts, can then be reconstructed for each language from a packed version contained in the GitHub repository.

Bukusu Data The dictionary recordings contain a word uttered several times in various forms, e.g. noun recordings include variations of the headword, singular and plural forms of the word while verbs include variations of the headword, infinitive and stem. The recordings also include pronouns, adjectives, adverbs, numerals and conjunctions. The dictionary contains 3.7 hours of data. The preparation process for these recordings included using SoX[9] to change the sampling rate and in some cases the sampling encoding. We used the WebRTC[10] voice activity detection tool to split the audios on silences, and manually inspected the result to ensure all segments and transcriptions matched up properly.

East Tusom Data This recently published dataset, Tusom2021[11], was prepared explicitly for the task of phone recognition and therefore needed no pre-processing. It contains 55.3 minutes of audio data.

Orthography-to-Phone Mapping While the Bukusu and East Tusom transcriptions were already in the International Phonetic Alphabet (IPA), it was necessary to develop an orthography to phone (IPA) mapping for Saamia to enable us obtain phonetic transcriptions of our data using Epitran [16]. These mappings were developed in collaboration with linguists and will be released as additions to the Epitran IPA transcriber [16].

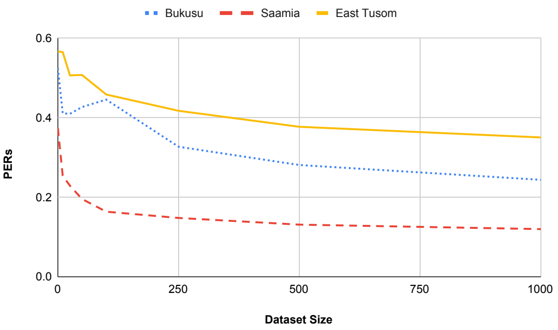

Figure 2: *Dataset sizes against PERs curves at training epoch 40 for Bukusu, Saamia and East Tusom.*

4. Experiments

The experimental splits used for the datasets were as follows:

- Bukusu: 6442 (train), 1001 (dev), 2458 (test)
- Saamia: 7254 (train), 1000 (dev), 1500 (test)
- East Tusom: 1600 (train), 400 (dev), 392 (test)

4.1. Experimental Setting

Given a pre-trained acoustic model and phone recognizer based on universal phone representations, we are interested in understanding how much data leads to improvements in model performance via fine-tuning. We thus create datasets of different sizes; 10, 25, 50, 100, 250, 500 and 1000. This selection of dataset sizes is intended to be an approximately doubling progression. These datasets are transcribed with phones, we therefore evaluate using phone error rate during fine-tuning of the languages and use the same test set across different dataset sizes for each language. All fine-tuning is done on one model therefore using the same encoder, a 6-layer stacked bidirectional LSTM with hidden size of 1024 in each layer. Fine-tuning in each instance is carried out for 250 epochs, beyond which we noted no significant variation.

4.2. Experimental Results

Table 1 shows top-line PER results of inferences in various conditions; the results of zero-shot inference using the pre-trained Allosaurus model, the results when the allophone layer is constrained to phones found in the respective languages, the best results after fine-tuning using a 100-sized dataset, the best results after fine-tuning using a 1000-sized dataset and the results after fine-tuning using the entire training set of each language dataset. We deduce that fine-tuning, even with just 100 utterances, leads to significant improvements in PER.

Figure 2 shows the variation of PERs across various dataset sizes at training epoch 40 for all 3 languages. We note that as expected, the PER mostly decreases as dataset size increases, particularly for Saamia.

Figure 3 shows the accuracy over training epochs for dataset sizes 10, 100 and 1000. Across all varieties (and when an adequate number of training instances are available) fine-tuning quickly improves the PER after a few epochs. We note that smaller fine-tuning datasets, of just 10 or 100 instances in Bukusu and 10 instances in East Tusom, only require a hand-

[7] Bible.is website

[8] CMU Wilderness GitHub repository

[9] SoX-Sound eXchange

[10] python interface to the WebRTC Voice Activity Detector

[11] East Tusom Github repo

Table 1: *Fine-tuning of Allosaurus (even with just 100 utterances) leads to significant improvements in PER. The PER improvements (Δ) reported are a percentage relative to the language-constrained baseline.*

	Bukusu		Saamia		East Tusom	
	PER	Δ	PER	Δ	PER	Δ
Allosaurus	72.8		63.7		67.5	
+ constraint	52.5		37.4		56.7	
+ fine-tuning (100)	41.2	21.5%	15.5	58.5%	44.8	20.9%
+ fine-tuning (1000)	17.3	67.0%	11.7	65.7%	34.6	38.9%
+ fine-tuning (all)	5.2	90.1%	9.2	75.4%	33.1	41.6%

ful of fine-tuning epochs to reach their best performance. In the case of Bukusu, Figure 3(top) makes apparent that the 10 to 100 dataset size PERs are somewhat consistent, even appearing to worsen as fine-tuning progresses. This is likely due to the nature of the data, where each training instance is a single dictionary recording and each audio contains one word. This means the dataset size is equivalent to the number of words or utterances in the training set, so dataset sizes 10 to 100 contain very little data. East Tusom exhibits behaviour similar to Bukusu as shown in Figure 3(bottom) likely due to the fact that the dataset is also composed of individual utterances. We see that PERs for the 10 and 100-sized datasets in some of the initial training epochs become slightly worse than the zero-shot results with the language-constrained baseline before improving. In the case of the 10-sized dataset, the improvements over zero shot results with the language-constrained baseline are negligible, however when the datasets get bigger, we start to see better improvements with few training epochs before getting to the point of early stopping, 45 epochs.

A manual inspection of the differences of the test set outputs of the baseline and the fine-tuned models reveal interesting patterns. The Bukusu and Saamia baseline models underperform on long vowels, outputting short ones; the Bukusu fine-tuned model corrects this mistake in over 400 cases (e.g. the i→iː mistake is corrected 119 times) and the Saamia one in over 600 cases(e.g. the a→aː mistake is corrected 314 times). The Saamia model also under-performs on consonant blends(e.g. the d→nd mistake is corrected 294 times and n→nd mistake is corrected 233 times). The Bukusu baseline model also seems to confuse the liquids (l, ɾ and r) an issue that fine-tuning largely addresses. On the other hand, the fine-tuned model over-recognizes long vowels, and it seems slightly more likely to drop or insert phonemes spuriously.

In the case of East Tusom, the baseline model appears to struggle with differentiating the vowels with many of the corrections made by the fine-tuned model being replacing one vowel with another. (eg. o→u, a→o, e→i, a→õ)

5. Implications and Future Work

We show that fine-tuning of the Allosaurus model is a promising approach to phone recognition for unseen low-resource languages. The results indicate improvements over the language-constrained baseline with as little as 10 training instances (of course, using more fine-tuning data leads to even further improvements).

Future work could include tuning of the parameters of the Allosaurus model itself, or more sophisticated training techniques such as layer-by-layer adaptation of the encoder or allophone layers [17] and/or use of meta-learning [6]. In addition, as some language pairs are more closely related than others, it might be interesting to experiment with further fine-tuning on multiple languages from a language group and with mixed fine-tuning [18] where we sample some data from the original data and some from the target language. Finally, as many languages, such as the Luhya languages we covered here, are tonal it would be interesting to include tones in the pretraining of Allosaurus and perform tonal recognition. Finally, while we focus on phone recognition here, it would be of great utility to incorporate similar techniques into a full-fledged speech recognition systems.

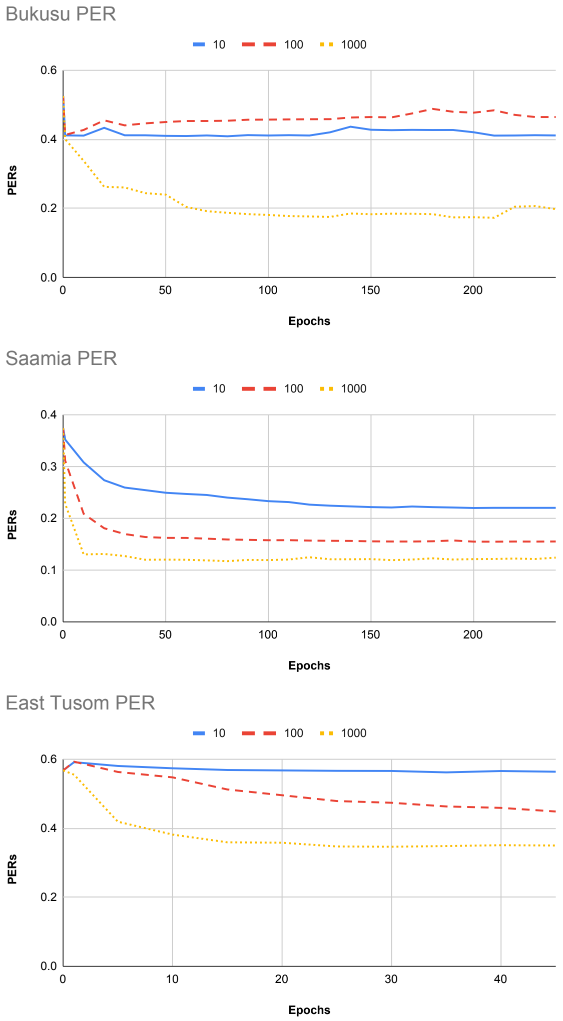

Figure 3: *Bukusu, Saamia and East Tusom accuracy over training epochs for dataset sizes 10, 100 and 1000. Due to early stopping in the case of East Tusom, we plot over 45 epochs in that instance and over 250 epochs for the others.*

6. References

[1] M. Wiesner, A. Renduchintala, S. Watanabe, C. Liu, N. Dehak, and S. Khudanpur, “Pretraining by backtranslation for end-to-end asr in low-resource settings,” *arXiv preprint arXiv:1812.03919*, 2018.

[2] M. C. Stoian, S. Bansal, and S. Goldwater, “Analyzing asr pretraining for low-resource speech-to-text translation,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7909–7913.

[3] S. Bansal, H. Kamper, K. Livescu, A. Lopez, and S. Goldwater, “Pre-training on high-resource speech recognition improves low-resource speech-to-text translation,” *arXiv preprint arXiv:1809.01431*, 2018.

[4] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” *arXiv preprint arXiv:2006.11477*, 2020.

[5] C. Wang, J. Pino, and J. Gu, “Improving cross-lingual transfer learning for end-to-end speech recognition with speech translation,” *arXiv preprint arXiv:2006.05474*, 2020.

[6] J.-Y. Hsu, Y.-J. Chen, and H.-y. Lee, “Meta learning for end-to-end low-resource speech recognition,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7844–7848.

[7] X. Li, S. Dalmia, J. Li, M. Lee, P. Littell, J. Yao, A. Anastasopoulos, D. R. Mortensen, G. Neubig, A. W. Black *et al.*, “Universal phone recognition with a multilingual allophone system,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 8249–8253.

[8] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[9] S. Moran and D. McCloy, “Phoible 2.0,” *Jena: Max Planck Institute for the Science of Human History*, 2019.

[10] R. M. French, “Catastrophic forgetting in connectionist networks,” *Trends in cognitive sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[11] J. MacArthur, *Cartography and the Political Imagination: Mapping Community in Colonial Kenya*. Ohio University Press, 2016.

[12] M. R. Marlo, R. Grollemund, T. Nguyen, E. Platner, S. Pribe, and A. Thein, “A phylogenetic classification of Luyia language varieties,” in *Proceedings of the 49th Annual Conference on African Linguistics*, G. Sibanda, D. Ngonyani, and J. Choti, Eds. Berlin: Language Science Press, forthcoming.

[13] N. Mutonyi, “Aspects of Bukusu Morphology and Phonology,” Ph.D. dissertation, Ohio State University, Columbus, OH, 2000.

[14] R. Botne, H. Ochwada, and M. Marlo, *A grammatical sketch of the Lusaamia verb*. Köln: Köppe, 2006.

[15] R. M. Angogo Kanyoro, *Unity in diversity: A linguistic survey of the Abaluyia in Western Kenya*. Vienna: Afro Publications, 1983.

[16] D. R. Mortensen, S. Dalmia, and P. Littell, “Epitran: Precision G2P for many languages,” in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, N. C. C. chair), K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, and T. Tokunaga, Eds. Paris, France: European Language Resources Association (ELRA), May 2018.

[17] J. Howard and S. Ruder, “Universal language model fine-tuning for text classification,” *arXiv preprint arXiv:1801.06146*, 2018.

[18] G. Neubig and J. Hu, “Rapid adaptation of neural machine translation to new languages,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, Oct.-Nov. 2018, pp. 875–880. [Online]. Available: https://www.aclweb.org/anthology/D18-1103