Phone-ing it in: Towards Flexible, Multi-Modal Language Model Training using Phonetic Representations of Data

Colin Leong
University of Dayton
cleong1@udayton.edu

Daniel Whitenack
SIL International
dan_whitenack@sil.org

Abstract

Multi-modal techniques offer significant untapped potential to unlock improved NLP technology for local languages. However, many advances in language model pre-training are focused on text, a fact that only increases systematic inequalities in the performance of NLP tasks across the world's languages. In this work, we propose a multi-modal approach to train language models using whatever text and/or audio data might be available in a language. Initial experiments using Swahili and Kinyarwanda data suggest the viability of the approach for downstream Named Entity Recognition (NER) tasks, with models pre-trained on phone data showing an improvement of up to 6% F1-score above models that are trained from scratch. Preprocessing and training code will be uploaded to https://github.com/sil-ai/phone-it-in.

1 Introduction

Pre-trained language models are increasingly applied in ways that are agnostic to targeted downstream tasks (Brown et al., 2020). This usage has led to a proliferation of large language models trained on enormous amounts of data. For example, the recent Megatron-Turing NLG 530B model was trained on the Pile, which includes 800GB+ of text (Gao et al., 2021), and other large language models utilize large portions of the 200TB+ common crawl data.[1] These large data sets include impressive amounts of text, but all languages are not represented equally (or at all) in that text. The reality is that only a negligible fraction of the 7000+ currently spoken languages (Eberhard et al., 2021) have sufficient text corpora to train state-of-the-art language models. This data scarcity results in systematic inequalities in the performance of NLP tasks across the world's languages (Blasi et al., 2021).

Local language communities that are working to develop and preserve their languages are producing diverse sets of data beyond pure text. The Bloom Library project,[2] for example, is being used by local language communities to create and translate "shell" or "template" books into many languages (426 languages at the time this paper is being written). However, Bloom allows users to do more than just translate text. Users are also recording audio tracks and sign language videos, which has resulted in 1600+ oral translations. Other examples showing the multi-modal nature of data in local languages include: (i) the creation of ChoCo: a multimodal corpus of the Choctaw language (Brixey and Artstein, 2021); (ii) SIL International's 50+ year effort to document endangered Austronesian languages via text, audio, and video (Quakenbush, 2007); (iii) the grassroots Masakhane effort catalyzing the creation and use of diverse sets of African language data (∀ et al., 2020); and (iv) work with the Me'phaa language of western Mexico that is producing digital recordings (video and audio) along with vocabulary, grammar and texts (Marlett and Weathers, 2018). These diverse data sources are effectively unusable by traditional text-based NLP techniques. In the light of data scarcity on these languages, they offer significant untapped potential to unlock improved NLP technology, if text data can be leveraged along with audio, image and video data. Furthermore, flexible multi-modal technology such as this will make it easier to include diverse people and communities such as those described above within the NLP technology development process - audio-based technology reducing the need for literacy, for example.

In this paper, we propose a multi-modal approach to train both language models and models for downstream NLP tasks using whatever text and/or audio data might be available in a language (or even in a related language). Our method uti-

[1] https://commoncrawl.org/

[2] https://bloomlibrary.org/

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 5306 - 5315
May 22-27, 2022 ©2022 Association for Computational Linguistics

lizes recent advances in phone recognition and text/grapheme-to-phone transliteration to convert input audio and text into a common phonetic representation (the IPA phone inventory). We then pre-train character-based language models in this phone-space. Finally, we fine-tune models for downstream tasks by mapping text-based training data into the phonetic representation. Thus, in addition to flexibility in pre-training, our method provides a way to reuse labeled text data for common NLP tasks, like Named Entity Recognition or Sentiment Analysis, in the context of audio inputs.

We demonstrate our phonetic approach by training Named Entity Recognition (NER) models for Swahili [swh][3] using various combinations of Swahili text data, Swahili audio data, Kinyarwanda [kin] text data, and Kinyarwanda audio data. These two languages both originate from from the same language family, Bantu, and are spoken by millions of people in Eastern Africa, often within the same country, resulting in some overlap in loan words, etc. [4] However, they are both considered low-resource languages. Kinyarwanda in particular, though spoken by approximately 13-22 million people[5], has very little text data available in that language, with fewer than 3,000 articles on the Kinyarwanda-language Wikipedia, and Swahili comparatively ahead but still poorly resourced at approximately 68,000 articles, far less than many European languages.[6], though some datasets have been created such as KINNEWS (Niyongabo et al., 2020). On the other hand, Kinyarwanda is uniquely placed as a language to leverage speech-based technologies, due to well-organized efforts[7] to collect voice data for that language. It is in fact one of the largest subsets available on the Common Voice Dataset (Ardila et al., 2019), with 1,183 hours of voice clips collected and validated. Choosing these two languages allowed us to test the use of the technique on legitimately low-resourced languages that could benefit from improved NLP technology, and which as part of the same family of languages might be similar enough in vocabulary, grammar, sound systems and so on, to benefit from cross-lingual training.

We find that simple NER models, which just look for the presence or absence of entities, can be trained on small amounts of data (around 2000 samples) in the phonetic representation. Models trained for complicated NER tasks in the phonetic representation, which look for entities and their locations within a sequence, are improved (by up to 6+% in F1 score) through pre-training a phonetic language model using a combination of text and audio data. We see this improvement when fine-tuning either a Swahili or Kinyarwanda language model for downstream Swahili tasks, which implies that one could make use of text and audio data in related languages to boost phonetic language model performance. The utility of the method in data scarce scenarios and importance of pre-training depends on the complexity of the downstream task.

2 Related Work

There have been a series of attempts to utilize phonetic representations of language to improve or extend automatic speech recognition (ASR) models. Some of these jointly model text and audio data using sequences of phonemes combined with sequences of text characters. Sundararaman et al. (2021), for example, uses a joint transformer architecture that encodes sequences of phonemes and sequences of text simultaneously. However, this joint model is utilized to learn representations that are more robust to transcription errors. The architecture still requires text inputs (from ASR transcriptions) and generates outputs in both text and phoneme representations. In contrast, our approach allows for text input, audio input, or text plus audio input to language models.

Similarly, in (Chaudhary et al., 2018) and (Bharadwaj et al., 2016) investigate the potential of phoneme-based or phoneme aware representations and models, showing gains in performance, language transfer, and flexibility across written scripts. These works conduct training on text-based data only, using Epitran to convert to phonemes.

Baevski et al. (2021) transforms unlabeled text (i.e., not aligned with corresponding audio files) into phonemes in a scheme to train speech recognition models without any labeled data. This scheme involves a generator model trained jointly with a discriminator model. The generator model converts

[3] Language codes formatted according to ISO 639-3 standard: https://iso639-3.sil.org/

[4] see for example (Kayigema and Mutasa, 2021), which describes English loan words entering Kinyarwanda "very often via Kiswahili"

[5] Sources vary: Ethnologue cites "Total users in all countries: 13,133,980", but there are 22 million according to *WorldData.info* (https://www.worlddata.info/languages/kinyarwanda.php).

[6] https://meta.wikimedia.org/wiki/List_of_Wikipedias

[7] https://foundation.mozilla.org/en/blog/how-rwanda-making-voice-tech-more-open/

audio, segmented into phonetic units into predicted phonemes, and the discriminator model attempts to discriminate between these predicted phonemes and the phonemes transliterated from unlabeled text. Although both text and audio are utilized in this work, they are not input to the same model and the primary output of the training scheme is a model that creates good phonetic speech representations from input audio.

Outside of speech recognition focused work, Shen et al. (2020) (and other researchers cited therein) attempt to "fuse" audio and text at the word level for emotion recognition. They introduce another architecture that internally represents both audio and text. However, the so-called WISE framework relies on speech recognition to generate the text corresponding to audio frames in real-time. The current work explicitly avoids reliance on speech recognition. The 2021 Multimodal Sentiment Analysis (MuSe) challenge continues this vein of research integrating audio, video, text, and physiology data in an emotion recognition task (Stappen et al., 2021). Contributions to this challenge, such as Vlasenko et al. (2021), introduce a variety of ways to "fuse" audio and text inputs. However, these contributions are squarely focused on emotion/sentiment analysis and do not propose methods for flexible, phonetic language models.

Lakhotia et al. (2021) introduced functionality for "textless" NLP. They explored the possibility of creating a dialogue system from only audio inputs (i.e., without text). As part of this system, language models are directly trained on audio units without any text. This advances the state-of-the-art with regard to self-supervised speech methods, but it does not provide the flexibility in audio and/or text language modeling introduced here.

3 Methodology

Our approach is inspired by the fact that many languages are primarily oral, with writing systems that represent spoken sounds. We convert both text and audio into single common representation of sounds, or "phones," represented using the International Phonetic Alphabet, or IPA. Then, we perform both language model pre-training and the training of models for downstream tasks in this phonetic representation. Well-tested architectures, such as BERT-style transformer models (Vaswani et al., 2017), are thus flexibly extended to either speech or audio data.

Regarding the conversion process of text and audio data, we leverage recent advances to transliterate this data into corresponding sounds represented by IPA phonetic symbols. This transliteration is possible for speech/audio data using tools such as the Allosaurus universal phone recognizer, which can be applied without additional training to any language (Li et al., 2020), though it can benefit from fine-tuning(Siminyu et al., 2021). To convert text data to phonemes we can use tools such as the Epitran grapheme-to-phoneme converter (Mortensen et al., 2018), which is specifically designed to provide precise phonetic transliterations in low-resource scenarios.

Fig. 1 shows how downstream models for certain NLP tasks, like Named Entity Recognition (NER), are performed in the phonetic representation. Labeled data sets for NLP tasks need to be mapped or encoded into the phonetic representation to train downstream models. However, once this mapping is accomplished, models trained in the phonetic representation can perform tasks with audio input that are typically restricted to processing text input.

3.1 Phonetic Language Modeling

One complication arising from direct speech-to-phone transcription is the loss of word boundaries in the transcription. This is expected, as natural speech does not put any pauses between the words in an utterance. This does, however, result in mixing text data sets containing clear word boundaries with speech data sets containing no clear word boundaries.

Borrowing from techniques used on languages that do not indicate word boundaries by the use of whitespace, we address the problem by removing all whitespace from our data sets after phone transliteration. We train *character-based* language models over the resulting data. Character-based models such as CharFormer (Tay et al., 2021) or ByT5 (Xue et al., 2021) have shown promise in recent years for language modeling, even if this approach is known to have some trade offs related to shorter context windows.

3.2 Potential Information Losses

The transliteration of text and audio data into phonetic representations presents several other challenges related to potential loss of information or injection of noise:

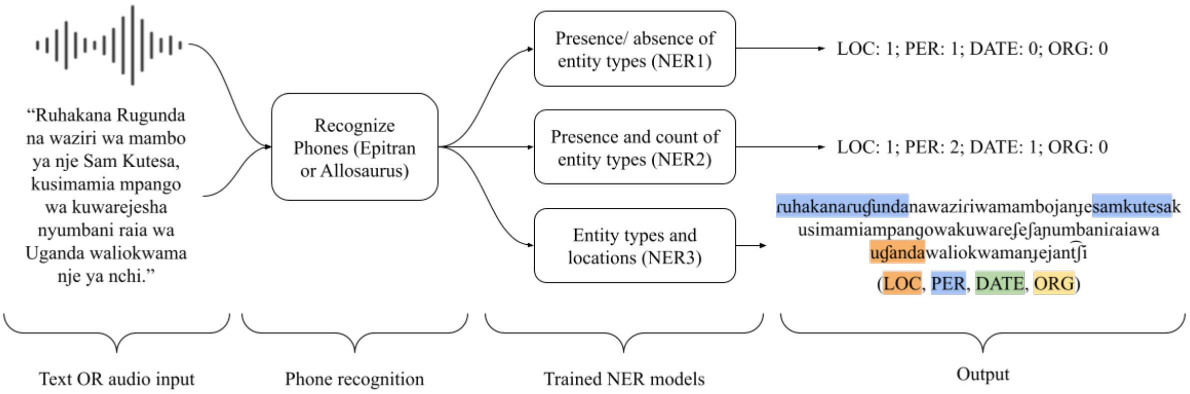
"Ruhakana Rugunda na waziri wa mambo ya nje Sam Kutesa, kusimamia mpango wa kuwarejesha nyumbani raia wa Uganda waliokwama nje ya nchi."
Recognize Phones (Epitran or Allosaurus)
Presence/ absence of entity types (NER1)
Presence and count of entity types (NER2)
Entity types and locations (NER3)
LOC: 1; PER: 1; DATE: 0; ORG: 0
LOC: 1; PER: 2; DATE: 1; ORG: 0
ɾuhakanaɾuɠundanawaziɾiwamambojaɲɟesamkutesak
usimamiampaŋgowakuwaɾeʄeʃaɲumbaniɾaiawa
uɠandawaliokwamaɲɟejant͡ʃi
(LOC, PER, DATE, ORG)
Text OR audio input
Phone recognition
Trained NER models
Output

Figure 1: Our approach: input from either modality can be converted by phone recognition, e.g. Epitran for text, Allosaurus for speech. Then we test on several downstream tasks which we designate NER1, NER2, NER3.

1. *Loss of suprasegmental information*: In some languages, meaning may be encoded through tones, or pitch changes *across* sounds (aka across segments, or "suprasegmental"). Particularly for tonal languages such as Mandarin Chinese [cmn], this loss can represent a significant informational loss particularly for homophones with different tones, as seen in (Amrhein and Sennrich, 2020). While IPA symbols can represent these intricacies, it adds complexity

2. *Phone/phoneme differences*: As noted in (Li et al., 2020), speech sounds which are physically different (different *phones*), may be *perceived* as the same (one *phoneme*) by speakers of one language, but these same sounds could perhaps be distinguished by speakers of another language. For example, the French words words *bouche*, and *bûche* contain phones (/u/ vs. /y/) which may sound "the same" to English speakers, but are semantically different to French speakers. In other words, in English, both *phones* map to the same *phoneme* perceptually. As the Allosaurus phone recognizer recognizes the actual phones/sounds, not their perceived phonemes, it would transcribe these two phones to different representations even for English speech. This can be mitigated to an extent by customizing the output of Allosaurus on a per-language basis, see Sec. 4.3.

3. *Simple errors in phone recognition*: As noted in (Siminyu et al., 2021), even the best-trained Allosaurus models, fine-tuned on language-specific data, have a non-trivial Phone Error Rate (PER).

An important question, therefore, is whether these added sources of noise/information losses are outweighed by the potential benefits in terms of flexibility. Does working in a phonetic representation cause a prohibitive amount of information loss? We constructed our experiments and data sets in order to answer this question.

4 Experiments

In order to evaluate the quality of learned phonetic representations, we transliterate several text and audio data sets in the Swahili [swh] language. We pre-train phonetic language models on various combinations of these data sets and evaluate downstream performance on NER tasks. See Fig. 2 for a detailed overview of these various combinations.

We refer to these combinations as denoted by downstream tasks (SNER for **S**wahili NER), and pre-training language ((**K** for **K**inyarwanda, **S** for **S**wahili) as well as data modality (**T** for text, **A** for audio). By way of example, the SNER+ST2 model results from pre-training using **2** swh **t**ext datasets (ST2) and fine-tuning on the swh **NER** (SNER) task, whereas the SNER+SAT model results from pre-training using swh **a**udio and **t**ext data (SAT).

Kinyarwanda [kin] data is used in our experiments as a language related to the target language (swh) with existing text and audio resources that, in some ways, surpasses those available in the target language. Thus, we pre-train some models on kin data while fine-tuning for the downstream NER task using swh data.

Three different formulations of the NER task, from more simple (NER1) to more compli-

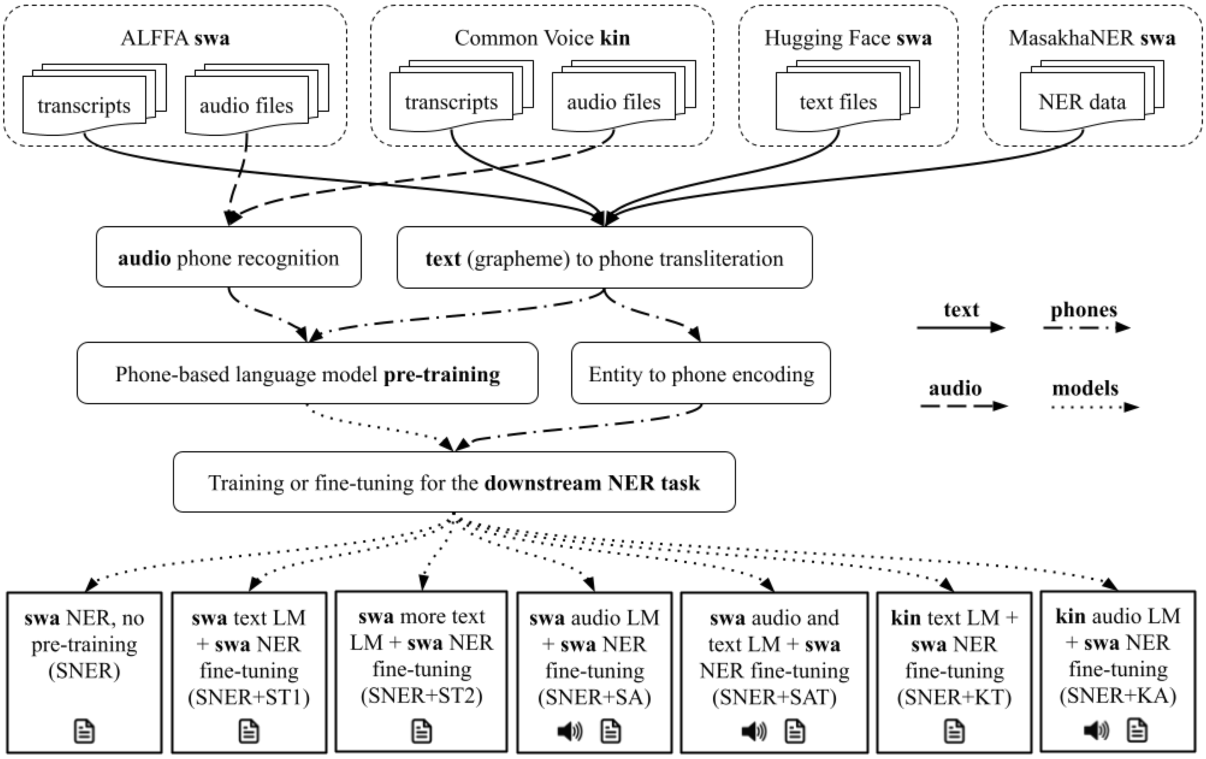
ALFFA swa
transcripts
audio files
Common Voice kin
transcripts
audio files
Hugging Face swa
text files
MasakhaNER swa
NER data
audio phone recognition
text (grapheme) to phone transliteration
text
phones
audio
models
Phone-based language model pre-training
Entity to phone encoding
Training or fine-tuning for the downstream NER task
swa NER, no pre-training (SNER)
swa text LM + swa NER fine-tuning (SNER+ST1)
swa more text LM + swa NER fine-tuning (SNER+ST2)
swa audio LM + swa NER fine-tuning (SNER+SA)
swa audio and text LM + swa NER fine-tuning (SNER+SAT)
kin text LM + swa NER fine-tuning (SNER+KT)
kin audio LM + swa NER fine-tuning (SNER+KA)

Figure 2: Training scenarios: we pre-train on various combinations of phonemized datasets, evaluating on the downstream NER task. SNER-ST denotes "**S**wahili **T**ext (ST) pre-training, **S**wahili **NER** (SNER) fine-tuning", SNER-SAT denotes Swahili NER with **S**wahili **A**udio and **T**ext (SAT) pre-training, SNER-**KA** uses **K**inyarwanda **A**udio (KA), etc.

cated/granular (NER3), are used (see Fig. 2) to help determine the applicability of our methods to less challenging (NER1) to more challenging (NER3) tasks. The NER1 task tries to determine the presence or absence of certain kinds of entities within an input. For our task we use PER, ORG, DATE, and LOC entities. The NER2 task additionally requires models to predict the correct numbers of these entities within an input. Finally, the NER3 task requires models to determine entities at the correct locations with an input sequence of phones.

For all of these tasks, we first convert text data to phones using Epitran and audio data to phones using Allosaurus. Then, we pre-train on various combinations of data, before fine-tuning on NER.

4.1 Data Sources

For `swh` pre-training data we use: (i) the "Language Modeling Data for Swahili" dataset (Shikali and Refuoe, 2019) hosted on Hugging Face (which we refer to as the "HF Swahili" data set); and (ii) the ALFFA speech dataset (Gelas et al., 2012). For ALFFA data we process both the audio files (using Allosaurus) and the original "gold" text transcriptions (using Epitran).

For Kinyarwanda pre-training data, we use the Common Voice (CV) Kinyarwanda 6.1 subset (Ardila et al., 2019). Again, we utilize both the audio files and transcriptions. Due to the large size of the CV 6.1 Kinyarwanda subset, we processed only about 80% of the audio files.

For fine-tuning the downstream NER task, we use the MasakhaNER data set (Adelani et al., 2021). As with other text-based data sets, we transform the NER sample with Epitran to map the samples into the phonetic representation.

4.2 Entity to Phone Encoding

For the downstream NER tasks we map or encode the NER annotations into the phonetic representation. We thus edited the labels (PER, ORG, DATE, and LOC) to convert them from word-level labels to phone-level labels as shown in Fig. 3. Unlike (Kuru et al., 2016), we leave in the B- and I- prefixes.

Our fork of the MasakhaNER data set, which implements our phonetic representations of the labels, is published on Github.[8].

[8] https://github.com/cdleong/masakhane-ner

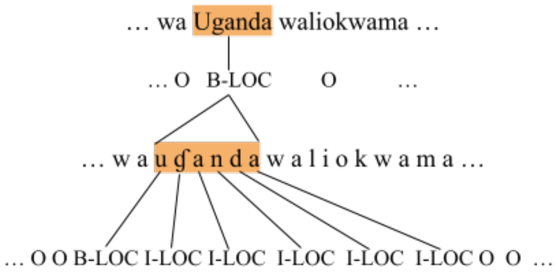
… wa Uganda waliokwama …
… O B-LOC O …
… w a u ɗ a n d a w a l i o k w a m a …
… O O B-LOC I-LOC I-LOC I-LOC I-LOC I-LOC O O …

Figure 3: Adaptation of word-level NER annotations to character-level annotations.

4.3 Phone Inventory Considerations

As mentioned already, we use Allosaurus for phone recognition with audio inputs. In order to ensure consistency with Epitran, we took advantage of Allosaurus's inventory customization feature, giving it the phone inventories specified by the same language in Epitran. The inventory used throughout this work (for `swh`) is the swa-Latn inventory from Epitran.[9] When this inventory is supplied as input, Allosaurus will only output symbols from the inventory. We followed similar practice when transliterating Kinyarwanda data.

We compare the output of Epitran and Allosaurus on the ALFFA dataset. Following the practice of (Li et al., 2020), we used the `editdistance`[10] library to calculate the Phone Error Rate (PER). Having no ground truth phone annotations, we instead take Epitran's outputs as "ground truth" for comparison. The mean PER between the outputs is 23.7%. This result is consistent with Siminyu et al. (2021), which finds PERs as high as 72.8% when testing on on the Bukusu (bxk), Saamia (lsm) and East Tusom languages (an endangered subdialect of the Tungkhulic language family). However, by training the phone recognizer on even minimal amounts of data in these languages, PERs were improved significantly.

A spreadsheet with detailed results for 10k samples from ALFFA can be found online.[11]

4.4 Model Architecture and Training

All models use the SHIBA implementation of CANINE (Tanner and Hagiwara, 2021). SHIBA was designed for use on the Japanese [`jpn`] language, which does not include spaces between its characters (similar to our phonetic representations without word boundaries). We used the default hyperparameter settings for SHIBA pre-training and fine-tuning, because we are primarily concerned with the relative impact of various combinations of pre-training data on the downstream NER tasks. We use the Hugging Face transformers library (Wolf et al., 2020) to train all models.

Because of the small size of the NER data set used during fine-tuning, we enabled Hugging Face's early stopping callback for all downstream training runs. We stopped these runs if they did not improve training loss after 20 evaluations. Nonetheless, we found after a number of trials that the models quickly overfit using this setting. We also experimented with modifying this on several trials to stop based on the evaluation loss instead, but this change did not significantly influence the evaluation results.

Following the example of Adelani et al. (2021), we do not run downstream model trainings once, but multiple times. We also pre-trained each phonetic language model multiple times with different random seeds. We report averages of these multiple trials in the following.

Scripts and code for our experiments will be uploaded to Github.[12]

5 Results and Discussion

Table 1 presents the F1 scores for our training scenarios in the downstream NER1 and NER2 tasks. The models that utilize pre-training on the `kin` audio and text data give the best results. However, pre-training does not appear to dramatically influence the level. F1 scores in the range of 74-85% suggests the minimum viability of these phonetic models for simple NLP tasks.

Table 2 presents the F1 scores for our various training scenarios in the downstream NER3 task, which should be the most challenging for our phonetic models. The influence of pre-training is more noticeable for this task. Further, the models pre-trained on the `kin` audio and text data have the best performance. This is likely due to the fact that the `kin` data is both large and higher quality (in terms of sound quality) as compared to the ALFFA Swahili data. This benefit of this data size and quality appears to outweigh any degradation due to the pre-training occurring in a different (although related) language.

[9] https://bit.ly/30f8YCI

[10] https://github.com/roy-ht/editdistance

[11] https://bit.ly/3F0is3t

[12] https://github.com/sil-ai/phone-it-in

Model	F1 NER1	F1 NER2
SNER	0.829	0.753
SNER+ST1	0.827	0.770
SNER+ST2	0.824	0.747
SNER+SA	0.817	0.751
SNER+SAT	0.818	0.763
SNER+KT	0.823	**0.771**
SNER+KA	**0.846**	0.763

Table 1: Mean results for presence/absence of entity types (NER1) and presence and *count* of entity types (NER2). Average of at least three trials per experiment, calculated with the scikit-learn library. (Pedregosa et al., 2011)

Model	F1	F1 (strict)
SNER	0.357	0.161
SNER+ST1	0.401	0.213
SNER+ST2	0.394	0.166
SNER+SA	0.363	0.163
SNER+SAT	0.405	0.203
SNER+KT	**0.408**	**0.217**
SNER+KA	0.397	0.197

Table 2: Prediction of entity types and precise locations (NER3). Average of at least three trials per experiment, scores calculated with seqeval library. (Nakayama, 2018)

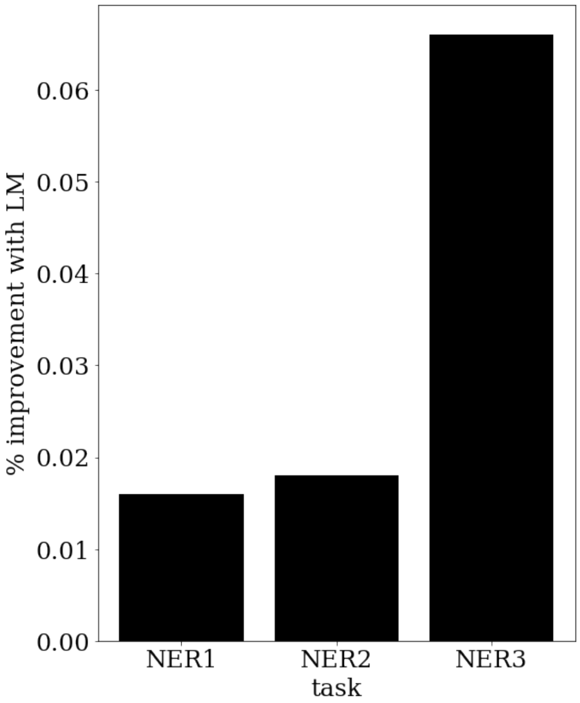

Figure 4: The max percentage improvement with fine-tuning for each kind of NER task that was explored. Presence/absence of entity types (NER1), presence and *count* of entity types (NER2), and prediction of entity types and precise locations (NER3).

The importance (or relative impact) of pre-training phonetic language models increases with the complexity of the NER task. Fig. 4 shows the maximum percentage improvement due to pre-training for each of our NER tasks. This suggests that simple NLP tasks with a small number of output classes are much easier to port to phonetic representations, even without pre-training, while more complicated NLP tasks may require a more significant amount of text and/or audio data for pre-training. We expect this trend to carry through to tasks like sentiment analysis, which could be formulated as a simple classification task with NEG, NEU, and POS sentiment labels or a more complicated aspect based sentiment analysis task.

6 Conclusions and Further Work

The proposed method for multi-modal training using phonetic representations of data has minimum viability for simple NER tasks. For more complicated NER tasks, pre-training phonetic language models boosts downstream model performance by up to 6% in F1 scores. This pre-training can be performed in the target language or in a related language using text and/or audio data. Thus, the method provides flexibility in the data needed to train language models, while also allowing for audio and/or text inputs to models trained on downstream NLP tasks.

We anticipate exploring various extensions to and validations of this method in the future. Specifically, we would like to explore methods that might mitigate performance degradation due to a lack of word boundaries in our method. Subword tokenization techniques, such as Byte-Pair Encodings (BPE) (Sennrich et al., 2016; Gage, 1994), or character-based word segmentation techniques might help in detecting and exploiting repeating patterns within the phonetic representation. Furthermore, the word embedding techniques used by (Chaudhary et al., 2018) or (Bharadwaj et al., 2016) have been shown to work well, and would be worth investigating how the removal of space-delimited word boundaries would affect this.

We would also like to validate our methods on a variety of other data sets and tasks. We selected the MasakhaNER dataset for evaluation because we specifically wished to evaluate results on actual low-resource languages supported by both Allosaurus and Epitran. While there are still, we argue, detectable improvements in downstream results with our method, further work would benefit from additional evaluations on other data sets or tasks. In particular, the Swahili News Classification corpus (David, 2020) corpus may provide a useful evaluation.

We did not investigate going from audio to phones, then phones to words/characters, judging that information losses and errors would likely compound in multiple stages of processing. Instead, we focused on what could be achieved with the Allosaurus "universal phone transcriber" without any language-specific finetuning. A truly universal transcriber would increase flexibility when training for truly low-resource scenarios.

Nevertheless, it has been shown by Siminyu et al. (2021) that it is possible to improve phone recognition with even small amounts (approximately 100 sentences) of annotation. It may be possible to improve phonetic language modeling results by performing this fine-tuning in the target language.

Experiments involving other languages with, e.g. languages that are *not* related would help to isolate the role of relatedness, lexical overlap, or related sound systems/phonology.

While we do not claim that conversion to phones provides better performance generally, we believe that our experiments show that the fundamental idea of converting *either* text *or* audio data to the common phone representation provides a viable path to more flexible approach to certain downstream NLP tasks, worthy of further development.

Acknowledgements

The authors wish to thank Dr. Vijayan Asari, Dr. Steven Rogers, Dr. Julia Kreutzer, Dr. Graham Neubig, Dr. David Mortenson, Andre Niyongabo Rubungo, and Joshua Turner for advice, helpful discussions, assistance in debugging, and time spent in proofreading. In addition, David Adelani and the Masakhane community provided invaluable help, encouragement and assistance with the MasakhaNER dataset.

We used GNU Parallel for much of the dataset processing (Tange, 2011). In combination with Lhoest et al. (2021) from Hugging Face, GNU Parallel significantly accelerated pre-processing and phone transcription.

ClearML (AI, 2019) provided experiment tracking, model and dataset management, and (when needed) prompt and helpful technical support. As our project involved the creation of over 20 distinct dataset variations and training many models on some of them, these management tools significantly eased the entire research process.

Ethics Statement

This research project uses open datasets and models, which are used in accordance with corresponding licenses to the best of our knowledge. For the downstream task in question (NER), we used the MasakhaNER dataset, which is constructed from newspaper data. Where this newspaper data includes mentions of individuals, the individuals are public figures. The domain of this NER data is limited to the newspaper/news domain, which should be kept in mind while considering the applicability of the methods presented.

In terms of compute, the work presented here required approximately 200 pre-training or finetuning jobs tracked via ClearML. Each run lasted no more than 1-2 hoursfor finetuning, but generally much longer for pretraining (on the order of a day), and only consumed one GPU resource at a time (either an A100 or P100). This computation sums up to around 5-6 GPU-weeks on the A100, about one gpu-week on the Titan RTX, and several compute-days each for the other GPUs. Additional exploratory work and debugging consumed another few GPU-days on Google Colab.

References

D. Adelani, Jade Z. Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, Stephen Mayhew, Israel Abebe Azime, Shamsuddeen Hassan Muhammad, Chris C. Emezue, Joyce Nakatumba-Nabende, Perez Ogayo, Anuoluwapo Aremu, Catherine Gitau, Derguene Mbaye, J. Alabi, Seid Muhie Yimam, Tajuddeen R. Gwadabe, Ignatius Ezeani, Rubungo Andre Niyongabo, Jonathan Mukiibi, Verrah A Otiende, Iroro Orife, Davis David, Samba Ngom, Tosin P. Adewumi, Paul Rayson, Mofetoluwa Adeyemi, Gerald Muriuki, Emmanuel Anebi, Chiamaka Ijeoma Chukwuneke, Nkiruka Bridget Odu, Eric Peter Wairagala, S. Ajiboye Oyerinde, Clemencia Siro, Tobius Saul Bateesa, Temilola Oloyede, Yvonne Wambui, Victor Akinode, Deborah Nabagereka, Maurice Katusiime, Ayodele Awokoya, Mouhamadane Mboup, Dibora Gebreyohannes, Henok Tilaye, Kelechi Nwaike, Degaga Wolde, Abdoulaye N Faye, Blessing Sibanda, Orevaoghene Ahia, Bonaventure F. P. Dossou, Kelechi

Ogueji, Thierno Ibrahima Diop, Abdoulaye Diallo, Adewale Akinfaderin, Tendai Munyaradzi Marengereke, and Salomey Osei. 2021. MasakhaNER: Named Entity Recognition for African Languages. *Transactions of the Association for Computational Linguistics*, 9:1116–1131.

Allegro AI. 2019. Clearml - your entire mlops stack in one open-source tool. Software available from http://github.com/allegroai/clearml.

Chantal Amrhein and Rico Sennrich. 2020. On Romanization for model transfer between scripts in neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2461–2469, Online. Association for Computational Linguistics.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. 2019. Common voice: A massively-multilingual speech corpus. *CoRR*, abs/1912.06670.

Alexei Baevski, Wei-Ning Hsu, Alexis Conneau, and Michael Auli. 2021. Unsupervised speech recognition. *ArXiv*, abs/2105.11084.

Akash Bharadwaj, David Mortensen, Chris Dyer, and Jaime Carbonell. 2016. Phonologically aware neural model for named entity recognition in low resource transfer settings. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1462–1472, Austin, Texas. Association for Computational Linguistics.

Damián E. Blasi, Antonios Anastasopoulos, and Graham Neubig. 2021. Systematic inequalities in language technology performance across the world's languages. *ArXiv*, abs/2110.06733.

Jacqueline Brixey and Ron Artstein. 2021. Choco: a multimodal corpus of the choctaw language. *Language Resources and Evaluation*, 55:241–257.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Aditi Chaudhary, Chunting Zhou, Lori Levin, Graham Neubig, David R. Mortensen, and Jaime Carbonell. 2018. Adapting word embeddings to new languages with morphological and phonological subword representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3285–3295, Brussels, Belgium. Association for Computational Linguistics.

Davis David. 2020. Swahili : News classification dataset. The news version contains both train and test sets.

David M. Eberhard, Gary F. Simons, and Charles D. Fennig. 2021. *Ethnologue: Languages of the World*, twenty-fourth edition. SIL International, Dallas, Texas.

∀, Wilhelmina Nekoto, Vukosi Marivate, Tshinondiwa Matsila, Timi Fasubaa, Tajudeen Kolawole, Taiwo Fagbohungbe, Solomon Oluwole Akinola, Shamsuddee Hassan Muhammad, Salomon Kabongo, Salomey Osei, and others. 2020. Participatory research for low-resourced machine translation: A case study in african languages. *Findings of EMNLP*.

Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal archive*, 12:23–38.

Leo Gao, Stella Rose Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling. *ArXiv*, abs/2101.00027.

Hadrien Gelas, Laurent Besacier, and Francois Pellegrino. 2012. Developments of Swahili resources for an automatic speech recognition system. In *SLTU - Workshop on Spoken Language Technologies for Under-Resourced Languages*, Cape-Town, Afrique Du Sud.

Jacques Lwaboshi Kayigema and Davie Elias Mutasa. 2021. Aspects of deceptive cognate derived loanwords in kinyarwanda. *South African Journal of African Languages*, 41(2):113–122.

Onur Kuru, Ozan Arkan Can, and Deniz Yuret. 2016. CharNER: Character-level named entity recognition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 911–921, Osaka, Japan. The COLING 2016 Organizing Committee.

Kushal Lakhotia, Evgeny Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu Nguyen, Jade Copet, Alexei Baevski, Adel Ben Mohamed, and Emmanuel Dupoux. 2021. Generative spoken language modeling from raw audio. *ArXiv*, abs/2102.01192.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. 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, and Metze Florian. 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.

Stephen A. Marlett and Mark L. Weathers. 2018. The sounds of me'phaa (tlapanec): A new assessment. *SIL-Mexico Electronic Working Papers*, 25.

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).

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

Rubungo Andre Niyongabo, Qu Hong, Julia Kreutzer, and Li Huang. 2020. KINNEWS and KIRNEWS: Benchmarking cross-lingual text classification for Kinyarwanda and Kirundi. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5507–5521, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830.

J. S. Quakenbush. 2007. Chapter 4. sil international and endangered austronesian languages. In *LD&C Special Publication No. 1: Documenting and Revitalizing Austronesian Languages*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Guanghu Shen, Riwei Lai, Rui Chen, Yu Zhang, Kejia Zhang, Qilong Han, and Hongtao Song. 2020. Wise: Word-level interaction-based multimodal fusion for speech emotion recognition. In *INTERSPEECH*.

Shivachi Casper Shikali and Mokhosi Refuoe. 2019. Language modeling data for Swahili. Type: dataset.

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

Lukas Stappen, Alice Baird, Lea Schumann, and Björn W. Schuller. 2021. The multimodal sentiment analysis in car reviews (muse-car) dataset: Collection, insights and improvements. *ArXiv*, abs/2101.06053.

Mukuntha Narayanan Sundararaman, Ayush Kumar, and Jithendra Vepa. 2021. Phoneme-bert: Joint language modelling of phoneme sequence and asr transcript. *ArXiv*, abs/2102.00804.

O. Tange. 2011. Gnu parallel - the command-line power tool. *;login: The USENIX Magazine*, 36(1):42–47.

Joshua Tanner and Masato Hagiwara. 2021. SHIBA: Japanese CANINE model. Publication Title: GitHub repository.

Yi Tay, Vinh Tran, Sebastian Ruder, Jai Gupta, Hyung Won Chung, Dara Bahri, Zhen Qin, Simon Baumgartner, Cong Yu, and Donald Metzler. 2021. Charformer: Fast character transformers via gradient-based subword tokenization. *ArXiv*, abs/2106.12672.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Bogdan Vlasenko, RaviShankar Prasad, and Mathew Magimai.-Doss. 2021. Fusion of acoustic and linguistic information using supervised autoencoder for improved emotion recognition. *Proceedings of the 2nd on Multimodal Sentiment Analysis Challenge*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2021. Byt5: Towards a token-free future with pre-trained byte-to-byte models. *CoRR*, abs/2105.13626.