Zero-Shot Cross-lingual Phoneme Recognition from Yoruba to English

Aaron Bahr, Nikita L. Beklemishev, Haejin Cho, Kai Seidenspinner and **Ilinca Vandici**
Universität Tübingen

Abstract

In this paper, we pre-train an acoustic phoneme recognition model on the TIMIT dataset and evaluate its performance on the Yoruba portion of the Common voice data. We make use of the CTC architecture, allowing us to forego the need for time-aligned input data, and examine the models performance after transfer through a thorough and linguistically motivated feature weighting metric, reaching a 0.36 average Phoneme Error Rate on the Yoruba set. Leveraging the predictions produced by our model, we take an in-depth look at the effects of learning and transfer. All codes are available in our Github.

1 Introduction

Efficiently training an ASR system requires a rich, ideally time-aligned dataset. For low-resource languages, despite documentation efforts, exploiting the properties of transfer learning by pre-training on another, high-resource, language remains a sensible option. In our case, we focus solely on producing a consistent, generalized phonemic transcription, conditioned only on acoustic segments (no language model is included). For the purpose of zero-shot evaluation, picking a set of languages with similar phoneme inventories remains the practice yielding the best performance. We chose to work on transferring American English (< West Germanic < Indo-European) to Yoruba (< Volta-Niger < Atlantic-Congo), primarily spoken in Nigeria, whose phoneme inventories overlap to a large extent, despite different areas and *phyla*. Along with obtaining good performance on both languages, we aim to propose an efficient evaluation metric by employing a linguistically sound feature-weighted version of PER. We attempt to disentangle training from transfer errors by presenting a holistic view of our results, analyzing the confusion patterns in our predictions through metrics like cross-entropy and normalized PMI. In terms of the model architecture, using the *Connectionist Temporal Classification Loss* ensured that we could evaluate on non time-aligned data, with the additional benefit of enhancing adaptability to context (Reference to be added from Nikita)

1.1 Background

Successful transfer of knowledge in ASR relies on the fact that languages sound similar. However, acoustic sound is just the input part of the model. A common **function** from the sound to the target sequence is the necessary condition. However, the arbitrary variation between different orthographies makes for a poor generalization target. Luckily, for linguistics, the relevant representation of speech is phonemes, which tend to correspond to sound types—which are, by and large, commonly shared. Out of all the sounds that the vocal tract can produce, languages use only a small subset, and some sounds are much more common than others. (**?**, p. 6) The shared knowledge of sound types can be described using IPA symbols. (**?**) The IPA presents a well-defined framework through which this similarity is defined, and enables us to unify transcriptions across languages, ensuring that the learned function maps from the acoustic domain to a common representation. Furthermore, by analyzing **confusion and correspondence** patterns of the function from acoustics to phonemes inherited from English, and applied to Yoruba, we shed light on the variation and universals in these phonological systems. With architectures and compute power improving, models utilizing the property of transfer at the pre-training stage in order to achieve multilingual representations have become increasingly common. **?** is one example. However, this paper intends to enable transfer-learning without explicitly providing pre-defined feature mappings. We also believe that focusing on separate languages for training and evaluation allows us to gain a deeper insight of the mechanisms at work during transfer.

1.2 English to Yoruba transfer

The phonemic systems of Yoruba and English are not entirely similar., as can be observed in Appendix [ref], cf. (**???**). This is further complicated by several factors that might influence downstream performance. (1) Above all, the IPA does not aim to fully describe language's phonemic system, in a structuralist sence, but to represent "universal" phonetic features. (**?**, p. 40). Thus, for instance, we expect that the vowel space of both languages will have a different partition, even between the same IPA symbols. (2) Perceptually, one symbol may be interpreted as two in context of another phonological system. Yoruba has double-articulated sounds: /k͡p, g͡b/, /ĩ, ũ/, which could be mapped to sequences like /kp/ or /in/). (3) Phonological and phonotatic constraints are less universal (**?**) than inventories, which creates different sound sequence distributions. This hinders transfer-learning, since the RNN layer extracts temporal features during training. For example, English frequent diphthongs make the vowel–glide sequences much more probable than they are in Yoruba. It, in turn, has constraints on (C)V(N) syllables and vowel harmony. (**?**). (4) Ideally, the model should learn the contrasts of Yoruba, but it in fact is trained for distinctions of English. For example, unlike English, [n ∼l], [ɾ ∼ɹ] are allophonic in Yoruba. The shallow English transcription distinguishes allophones like [ɨ ∼i], and such cases will noise the correspondence between the predictions and Yoruba labels.

1.3 The inventories problem

Given the differences in inventories discussed above, relating English IPA predictions to the Yoruba IPA gold standard becomes a problem: What is a matching sequence, given that the targets only partly overlap? Phonological feature theory (**?**) provides us with a theoretical framework within which "similarity" exists on a gradient scale. Each sound has a feature representation indicating which natural classes it belongs and does not belong to. Although sounds vary in how many features from the overall set they specify, it is convenient to represent each sound as a vector over the entire feature inventory, with each feature coded as positive (+1), negative (1), or unspecified (0). The Hamming distance between two such vectors, restricted to nonzero dimensions and normalized by dimensionality, then serves as the dissimilarity measure (see Appendix C[ref]). This approach has been used for a long time in fields like cognate detection[] and sdialectometry.(**?**) . Among many notable databanks aggregating the knowledge of features, such as PHOIBLE (**?**) and SoundVector (**?**), we chose PanPhon (**?**) for its balance between the quality of representation and availability. The details of the implementation are discussed in Inference[ref].

2 Dataset

Training set For training we use the infamous TIMIT ASR corpus. **?** includes 6300 utterances recorded by 630 speakers from 8 major English dialects across the US. Annotations were done according to the customized IPA convention based on ARPAbet (**?**) (please refer to Appendix **??**). As both English and Yoruba are pluricentric languages, the learning benefits a lot from the variety in the input data. Even though TIMIT confines itself to the 1980s US language roof, it represents the existing dialectal variation well. This phonetic variation is necessary for learning to generalize over the variation in sounds, which is particularly relevant for transfer learning. Another reason for TIMIT is its ubiquity in ASR studies, which gives us confidence in yielding baseline results, comparable with other works in the area.

Preprocessing the training set For usage, we first concatenated the train, validation, and test splits before again randomly diving it into train and validation splits (with a 75 to 25 ratio). We perform a Fast Fourier Transform on the audio data, collecting 39 log-scale MFCC features, including first and second derivatives.

The TIMIT alphabet contained 63 unique labels in total. In order reduce prediction complexity and ensure the compatibility of the phonemic representation between both languages (and to make way for the IPA mapping process), we merged or split several labels. This included allophones not annotated in the Yoruba corpus: <ax-h> /ə̥/ and /ə/, syllabic sonorants, e.g. <eng> /ŋ̍/ and /ŋ/. Closures <dcl> /d̚ / and the following releases /d/ were joined into one label: we did not expect systematic unreleased closures in **open-syllable** Yoruba. (**?**) In the end, 15 label types were merged. As for the splitting the combinatorially large inventory of English diphthongs, we split them into vowel–glide sequences, keeping the vowels from the IPA convention: thus <oy> /ɔɪ/ became <ao y> /ɔ j/. This step was necessary since Yoruba does not have diphthongs. **?** We also concluded that splitting them will not perplex the prediction given that the CTC decoding does not need time alignment, and our evaluation ignores word boundaries.

Evaluation set Common Voice Yoruba data was used as a test dataset. Common Voice is a multilingual crowd-sourced corpus aimed for Speech Recognition purposes. (**?**) The audios were recorded by certified native speakers of each language. Annotations are suggested and later validated by other native speaker users via votes.There are 3.4k samples in total, with each sample including a MP3 file, speaker ID and audio transcription written in Yoruba orthography. The dataset also includes data from different dialects. To an extent, this accounts for the performance gaps that are shown to arise between standard Yoruba and other dialects in NLP tasks. **?**

Processing the evaluation set The sets of train (1.4k), validation (913) and test (1.1k) were again concatenated and used together for testing. We kept the original threshold of down votes for invalidated samples. As for the audio data, features were extracted under the same parameters as for the TIMIT dataset. Since Common Voice only provides an orthographic representation of the sentence, it was necessary to implement a grapheme-to-phoneme to obtain an IPA representation that could then be compared with the model output. For this purpose, we used Epitran Python module (**?**), and pre-processed the resulting strings. Word boundaries and pauses were removed. We also ignored the tone annotation (˧, ˥, ˨), although seeing the transfer abilities could be an interesting branch of research. We have removed the marginally phonemic /ɔ̃, ŋ/ from the inventory, merging with their allophones /ã/ and /n/. (**?**) Another marginal /ɛ̃/ remained. The data also turned out to contain an occurrence of dialectal <ụ> /ʊ/ (**?**), which is too small to generalize from, so we removed it as well from the evaluation for clarity. The resulting Yoruba IPA inventory is available in Appendix **??**.

3 Model and Traning

3.1 ResNet-Bi-LSTM model

Our model is based on that of (**?**), which used ResNet-BiLSTM model for Nepali Speech Recognition. The reason why we chose this model as our base reference was (1) to use pure neural-network-based model so that it is relatively easy to train and light-weight in terms of memory, (2) to include residual connection which is largely used in transformers as well as deep neural network models, and (3) to explore whether a model that is trained from scratch can also perform well on monolingual zero-shot cross-lingual speech recognition task unlike (**?**). We will first briefly overall architecture and setting of the original model and then list our adjustments.

The model starts with initial CNN layer and 5 consecutive ResNet blocks. These make up the 'ResNet Encoder', which aims to capture local dependencies. Each block consists of 2 unit blocks, with each unit block containing an initial convolution, Batch Normalization, using PrELU in lieu of the activation function. Each kernel has its size parameter set to 15, and performs 50 maps. The key idea behind residual connections is that adding the original input at deeper stages in the forward pass will result in grounding the representations (**?**) have attested that residual connections noticeably stabilize and optimize deep neural network training, which (**?**) confirms extends to audio data. A Bi-LSTM encoder part then follows the Residual Encoder. Bi-LSTM is designed to reflect distinctive temporally-related features in two opposite directions. (**?**) has two RNN layers with its dimension being both 170. As final layers, two dense layers and ReLU activation takes the output of bi-LSTM and projects the 170 dimensional output onto the phonemic embedding space.

We have made several crucial changes based on multiple experiments in order to prevent overfitting and to keep the depth of our model shallower under the zero-shot cross-lingual task setting. (1) The first major change is to reduce the depth of ResNet encoder block. Our model has 3 residual blocks while the original model has 5 blocks. Also, one residual block has only one set of unit block while the original residual block has 2 unit blocks. In other words, our ResNet encoder part is 0.3 times the depth of that of the base model. All other settings regarding ResNet encoder have fixed same as (**?**). (2) We have also diminished hidden dimensions of Bi-LSTM encoder and dense layer. From 170 to 128 as to hidden dimension of RNN layer and from 340 to 256 as to dense layer dimension. This also contributes to better computational efficiency as it corresponds to GPU's natural memory alignment and fetches. (3) Lastly, strong dropout rate was introduced to RNN layers. We set dropout

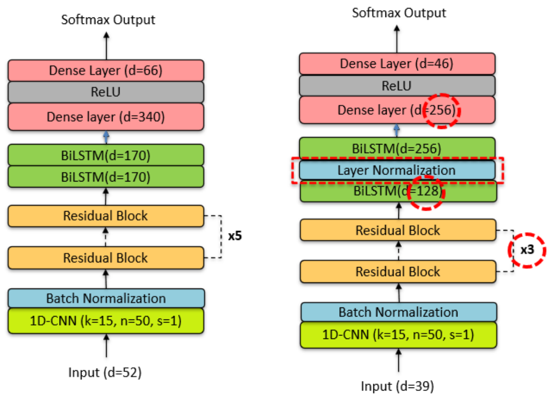


Figure 1: Left: model architecture of (?) , Right: Adjusted model architecture for English-Yoruba cross-lingual phoneme recognition. Changes highlighted with a red circle.

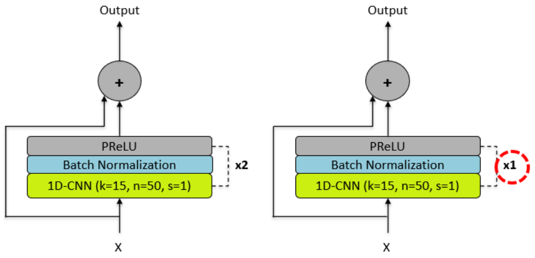


Figure 2: Left: residual block of (?), Right: Adjusted residual block for English-Yoruba cross-lingual phoneme recognition. Changes highlighted with a red circle.

as 0.4, which is two times higher than the original setting. This was also part of an effort to enable cross-lingual application. Refer to the following visualization that marks key differences between (?) and our model. Learning without an LM CTC.

We have made several crucial changes based on multiple experiments in order to prevent overfitting and to restrict the depth of the model under zero-shot cross-lingual task setting. (1) We downsize the number of ResNetEncoder layers from 5 to 3, and change the number of unit blocks from 2 to 1, meaning that we reduced the original model to a third of its depth. All other ResNet encoder settings remain fixed as in(?). We made changes to the dimensions of the Bi-LSTM encoder (reducing the hidden dimension of the RNN from 170 to 128) and the dense layer (from 340 to 256). We also diminished hidden dimensions of Bi-LSTM encoder and dense layer, From 170 to 128 as to hidden dimension of RNN layer and from 340 to 256 as to dense layer dimension. Finally, we introduced a stronger dropout rate to the dense layer, with the aim of facilitating cross-lingual application. Refer to the following visualization that marks key differences between (?) and our model.

We employed Connectionist Temporal Classification (CTC) loss as our training criterion. Unlike conventional frame-level cross-entropy, CTC does not require pre-aligned input-label pairs, which makes it especially suitable for low-resource languages such as Yoruba, where alignment information is unavailable. The key idea of CTC is to introduce a blank symbol and permit label repetitions so that multiple frame-level alignments can correspond to the same output sequence. During training, CTC computes the negative log-likelihood of the target sequence by summing the probabilities of all valid alignments:

$$L_{CTC} = \sum_{(X,Y)\in D} -\log P_{CTC}(Y \mid X)$$

where the probability of a target sequence Y given the input X is defined as

$$\begin{aligned} P_{CTC}(Y \mid X) &= \sum_{A\in B^{-1}(Y)} P(A \mid X) \\ &= \sum_{A\in B^{-1}(Y)} \prod_{t=1}^{T} p(a_t \mid h_t) \end{aligned}$$

with $A = (a_1, \ldots, a_T)$ being a frame-level alignment, B the collapse function that removes blanks and repeated labels, and h_t the hidden representation at time t. The blank symbol ϵ needs to be manually added to the alphabet so that the model can output blanks during training and inference.

At decoding time, greedy decoding selects the most probable label at each timestep, and the CTC collapse operation removes duplicates and blanks to produce the final prediction sequence $\hat{Y}$.

$$\hat{Y} = \arg\max_{Y} P_{CTC}(Y \mid X).$$

It is worth mentioning that, along with the purely practical lack of an aligned corpus, CTC can also in some cases present an advantage over architectures that rely on alignment. Indeed, **?** argues that aligned data might induce some biases, especially if the corpus is not diverse enough in regards to the kind of speech contexts available.

3.2 Train

Train settings were set through multiple experiments. The number of epochs is 50 and batch size was 64. We have adopted Adam as our stochastic gradient optimizer and set weight decay as 1e-4. A plateau-based learning rate scheduler was used, with an initial learning rate was 1e-3. The total number of parameters are 0.8 million. With the (hyper)parameter settings given above, training was noticeably stable and robust to overfitting. Both training and validation loss and PER consistently decreased and we halted training at the 16th epoch where Train PER reached 0.0223 and Validation PER reached a similar figure, 0.0339.

4 Results

As seen from the final PER results, the model shows a promising performance on the English dataset. We now introduce the framework we used for evaluation, before delving deeper into the results produced by transfer.

4.1 Evaluation Method

To evaluate the output of our model on Yoruba, we have to adjust the Phoneme Error Rate according to the issues introduced by the IPA inventories we described earlier Introduction[ref]. The metric we employ is feature-weighted PER. Like the regular PER, it is a Levenshtein distance between predicted a golden sequences, normalized by the length of the golden sequence. Unlike the PER, the substitution cost depends on the Hamming feature distance, thus reflecting the expected dissimilarity per gold phone. are not relevant in describing the English and Yoruba inventories: [sg, velaric, long, hitone, hireg]. We built up and adapted the original PanPhone source code to suit our goals, and also offer an optimized version of the alignment calculation algorithm. The main assumption behind our evaluation scheme is that the alignments achieved with this method are not only minimally costly, but also intuitively correct. As we see in Discussion, it is not always the case.

4.2 Inference

During inference, we observed that our model had a tendency to over-generate, making the output consistently longer than the gold labels. A look at the Appendix [ref] reveals that this is a transfer defect of the model. Some of the worst performances of the model arise from this over-generation. Whenever this happens, the inserted sounds tend to be somewhat rare phones that repeat: a lot of /ʔ, ɨ, ʉ, k, t, n/. This mildly leads to hypothesis that it reflects some extra-linguistic noise, that was not present in the training TIMIT recordings. Indeed, we noticed that the Common Voice audio data often included pauses, especially preceding and following the utterance. While for future investigations, it might be relevant to include noise reduction techniques, we chose to tackle this issue at the decoding/ evaluation step, by lowering the deletion cost to 0.5 and insertion cost to 0.75.

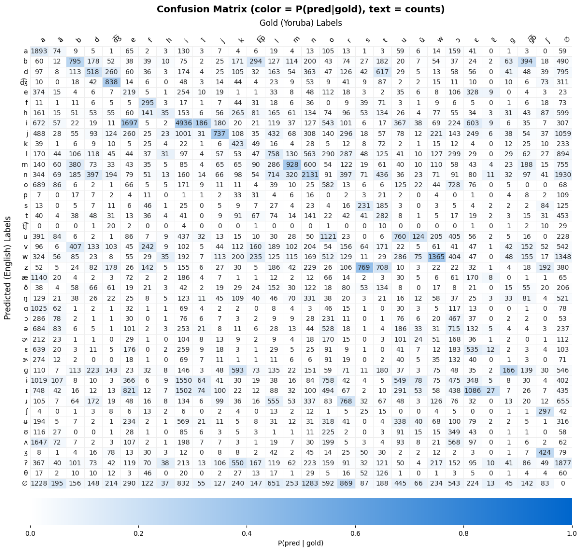


Figure 3: English-Yoruba Confusion Matrix with Posterior Probability

Results Default PER achieved on Yoruba was 0.34. It ranged from 0.18 to 1.11, with median 0.30. The best and worst predictions, as well as the alignment of the median prediction are in Appendix [ref]. With the original Levenshtein costs for deletion and insertion, median PER increases to 0.35.

4.3 Performance by phoneme

Alignment counts $weightedLevenshtein$ are depicted below, in the form of a confusion matrix **??**. Color encodes the posterior probabilities of the prediction labels given the gold labels.

Confusion entropy We used conditional entropy to find which Yoruba phonemes are well and poorly transferrable (Table **??**), assuming that a well-transferred sound will have a less uniform distribution of its alignments with predicted sounds: Most alignments would belong to a couple of phonetically close classes. For the formulas based on (**?**) refer to Appendix [ref]. For all of the phonemes the entropy was relatively high. For more accurately predicted phonemes the most common alignments are the phonetically similar sounds, in terms of feature distance, but there are still ambiguities, i.e. fricatives sometimes get predicted with their voiced pair—a contrast that Yoruba does not have. It seems, front vowels and consonants articulated in the front got accurate correspondences, whereas back vowels and consonants were more frequently confused. We found no possible phonetic explanation to this other than that it could be an artifact of the feature system.

Table 1: Confusion entropy, empirical and theoretical (feature-based) correspondences.

Gold	**Confusions**	**Similar**	**H**
s	[z, s, n, v]	[s, z, t, ʃ]	3.40
m	[m, n, l, w]	[m, b, n, p]	3.67
d͡ʒ	[d͡ʒ, d, n, z]	[d͡ʒ, t͡ʃ, ʒ, ʃ]	3.69
b	[b, v, m, n]	[b, p, m, v]	3.76
ʃ	[ʒ, ʃ, z, h]	[ʃ, ʒ, s, t͡ʃ]	3.80
i	[i, ɨ, ɪ, j]	[i, e, ɨ, ɪ]	3.81
w	[w, j, ʔ, u]	[w, ʍ, u, ɚ]	3.82
e	[i, ɪ, ɨ, j]	[e, i, æ, ɛ]	3.87
j	[j, i, w, ʔ]	[j, i, ɪ, w]	3.92
ĩ	[i, ɪ, ɨ, u]	[i, e, ɨ, ɪ]	3.93
ɛ̃	[ɪ, ɛ, n, ʔ]	[ɛ, e, ə, ɪ]	3.99
d	[d, n, g, b]	[d, t, n, z]	4.00
n	[n, m, l, d]	[n, d, l, m]	4.03
r	[ɹ, n, j, l]	[r, l, d, n]	4.07
ɛ	[ɪ, i, ɛ, ɨ]	[ɛ, e, ə, ɪ]	4.09
k	[g, ʔ, k, h]	[k, g, h, ŋ]	4.10
g͡b	[b, m, w, v]	[g, b, k, p]	4.13
f	[f, v, z, h]	[f, v, p, s]	4.14
u	[u, ɨ, i, ʉ]	[u, ʉ, o, ɨ]	4.16
t	[z, d, n, t]	[t, d, s, θ]	4.16
k͡p	[b, w, ʔ, v]	[k, p, g, b]	4.18
g	[g, b, w, v]	[g, k, ŋ, b]	4.18
ũ	[u, ɨ, w, ɪ]	[u, ʉ, o, ɨ]	4.21
l	[l, n, ɹ, j]	[l, d, n, z]	4.26
h	[ʔ, h, w, l]	[h, k, ʔ, j]	4.33
o	[u, ɨ, o, i]	[o, ɔ, ʌ, a]	4.41
a	[a, ʌ, æ, ɑ]	[a, æ, ɑ, ʌ]	4.52
ɔ	[o, ə, ʌ, ɨ]	[ɔ, o, ə, ʊ]	4.53
ã	[ɨ, o, u, ə]	[a, æ, ɑ, ʌ]	4.55

Mapping entropy A more interesting question in context of using the model transfer is whether we can restore the Yoruba phonemes reliably from the model predictions with TIMIT phonemes. To see how unambiguous this mapping is, how much signal our model contains after transfer, we use conditional entropy of the golden label probabilities given the predicted label. Appendix [ref]. The overall entropy was 3.5 bits, i.e. about 11 golden labels per prediction. However, for some of the labels it was much smaller, thus signifying almost one-to-one mapping from predicted labels to Yoruba.**??** The number would have likely been even smaller if not for deletions. Another thing raising the expected uncertainty is that the sounds that can map to multiple Yoruba labels, e.g. /m, l, w/, are among the most frequent ones.**??** All of the back vowels, whose high confusions we mentioned above, are now among the most certainly transferrable. This means that the high confusion entropy observed actually stems from the sheer amount of possible vowel segments in English. When English partitions the vowel space to /a, ɑ, æ, ʌ/, they all are included in Yoruba /a/. Ambiguous mappings (indicated by high entropy) also occur for sounds absent in Yoruba: IPA /ʔ, ɹ, ɚ, ð, θ/ have no clear correspondence in Yoruba to gravitate to. This mostly applies to consonants, which coincides with a commonly stated (**??**) intuition that vowel realizations distribute continuously in the acoustic space, while consonant allophones belong to one of some discrete targets.

4.4 Phone embeddings space

We attempt to plot phones as vectors in an easily interpretable manner, in order to investigate what the internal model representations for each language might look like. Ideally, the partition in the space would reflect similarities and differences for each target language. We run inference to produce phone embeddings from the last layer output (log probabilities in the phoneme classes space, so-called *PPGs*) and attempt to plot these using dimensionality reduction.

Phone posteriogram embeddings We obtain the PPG embeddings per phone by taking the log-probabilities for the first time stamp of each token after the CTC collapse. It runs against the intuition we have from spectrograms, but we consider it a suitable representation. Take a look at Fig. **??**, visualizing the conditional probability over the label inventory for each time step in a sentence. As a product of CTC decoding, posteriograms appear to be

Table 2: Mapping entropy and frequentmost correspondences.

Gold	Corr	H	Gold	Corr	H
a	[a, ã]	1.96	ɨ	[i, a]	3.41
ɑ	[a, ã]	2.01	ʉ	[u, ũ]	3.41
æ	[a, e]	2.32	ɪ	[ɪ, i]	3.48
ʃ	[ʃ, s]	2.33	ɚ	[ʊ, ɔ]	3.52
ʌ	[a, e]	2.49	n	[n, d]	3.59
t͡ʃ	[d͡ʒ, ʃ]	2.63	z	[s, d]	3.63
d͡ʒ	[d͡ʒ, ʃ]	2.80	ɹ	[j, l]	3.66
i	[i, e]	2.82	t	[t, d]	3.67
o	[o, ɔ]	3.05	θ	[s, t]	3.69
ʊ	[ʊ, ɔ]	3.06	ʔ	[h, k]	3.73
ɔ	[ɔ, o]	3.06	w	[w, u]	3.75
ɛ	[ɛ, e]	3.07	m	[m, b]	3.78
k	[k, g]	3.09	ŋ	[ŋ, g]	3.80
ʒ	[ʃ, d͡ʒ]	3.10	d	[d, t]	3.82
ə	[ɔ, ɛ]	3.28	l	[l, d]	3.91
ɝ	[ʊ, ɔ]	3.28	b	[b, m]	3.91
s	[s, t]	3.28	j	[j, i]	3.95
e	[e, i]	3.31	g	[g, k]	4.05
u	[u, ũ]	3.32	ð	[d, l]	4.16
f	[f, s]	3.33	h	[h, k]	4.25
p	[b, f]	3.36	v	[f, b]	4.26

different from spectrograms. Since the model is allowed to predict blank symbols, it tends to allocate blank symbols for majority of frames, while producing occasional spikes for non-blank phonemes. These spikes often appear at the time steps where phoneme predictions first become feasible.

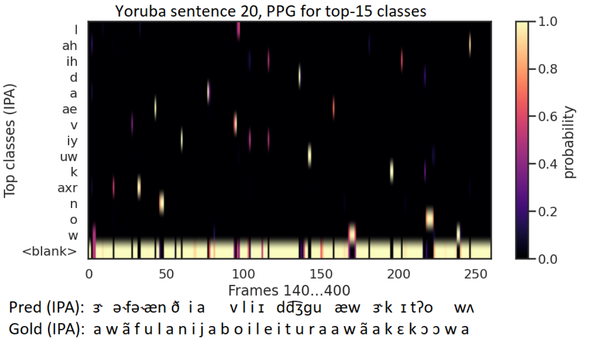


Figure 4: Phone posteriograms for one Yoruba sentence, after softmax, first 14 classes

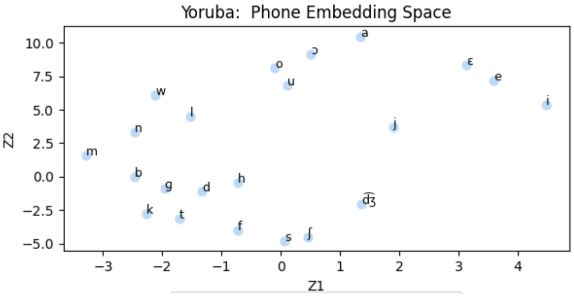


Figure 5: Yoruba Phone Embeddings (with English correspondences only)

Extraction We collected the PPGs for each correctly predicted phoneme (that is deemed a match by our alignment algorithm). Due to the overgeneration problem, it is entirely possible that some matching predictions are skipped in the process, but since we have not yet established a way to decode the noise that is generated, we limit ourselves to accurately aligned pairs. It also constraints us from analyzing the non-corresponding phonemes, like /ũ/ and /k͡p/. We employ *UMAP* to obtain 2-dimensional embeddings from 42-dimensional(CORRECT), for which we obtain the mean by phoneme. We settled on UMAP since it maintains the non-linear assumption of t-SNE regarding the data while being less dependent on initialization. While both plots seem to group similar sounds together to an extent, comparing inference simultaneously on both datasets remains tricky due to 1) the different probability distributions and the difference in dataset sizes 2) while UMAP remains an efficient dimensionality reduction method, it is unclear whether plotting always reflects similarity through clustering. We can see that in some cases, similar sounds do tend to cluster, but the extent to which these patterns can be identified different in both datasets.

Yoruba phones In the reduced Yoruba space **??**, cardinal vowels cluster in the same space in the graph, with their distance mostly corresponding to their feature weighted equivalent. We see how natural classes of phonemes group together in space: the sibilants /s/ and /ʃ/, plosives, sonorants, palatals. In that arrangement, the *y*-axis clearly corresponds to sonority hierarchy[quote]. It can also be argued that the *x*-axis loosely represents a gradual backness scale, when including acoustically related parameters like lip protrusion, in the case of labials.

English phones As for the TIMIT ?dimensional?CORRECT space **??**, these patterns are less clear. It is not easy to assign one linguistic feature to the axes, like sonority or backness. The natural groupings of sounds are still observable, but exact

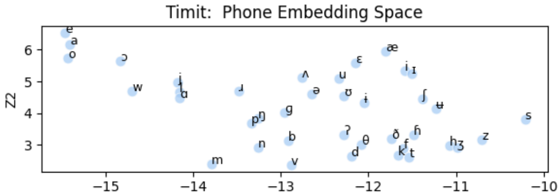


Figure 6: Timit Phone Embeddings

composition and number of clusters differs from Yoruba: nasal group separately from approximants and /l/; /p/ is separated from other voiceless stops. As for vowels, while /ə, ʌ/ and /i, ɪ/ (respectively positional allophones and a tense-lax pair vowel) are grouped together, it can be pointed out that /a, o, e/ appear as a detached cluster. Oddly, all of these labels belong only to nuclei of diphthongs, split after data processing, which has to be linked as to why they occupy their own space in the plot. It is thus probable that the differing annotations schemes for both datasets (with more fine-grained distinctions for TIMIT) partly contributes to this.

Yoruba and English The fact that the clusters differ is somewhat trivial, considering that the UMAP reduces space differently for different subsets of labels, and Yoruba clustering did not have to account for relations with all TIMIT sounds. Hence or whence we also look into the simultaneous projection of Yoruba and English embeddings in one space, by first fitting the transform based on the English data, and then reducing the Yoruba data, collecting Euclidian distances between the common phonemes. After running the projection several times, we find that the distances we obtain are consistent across trials, but it is difficult to extract a common pattern (Figures **??**, **??**).

Distances To investigate this further, we decided to run a correlation test between the Euclidian distance between phones after dimensionality reduction and feature weight. Figures **??**, **??**. This shows that there is a significant correlation between the two variables, with the correlation coefficient being lower for Yoruba than it is for English. Figure **??**. This reflects the fact that the English label inventory represents more varied sounds than the Yoruba predicted values, helping establish the correlation pattern during fitting. While this possibly shows that our presentations are reliable and in line with the framework we establish, this remains obscured by the fact that the plot for English is not easily interpretable.CORRECT

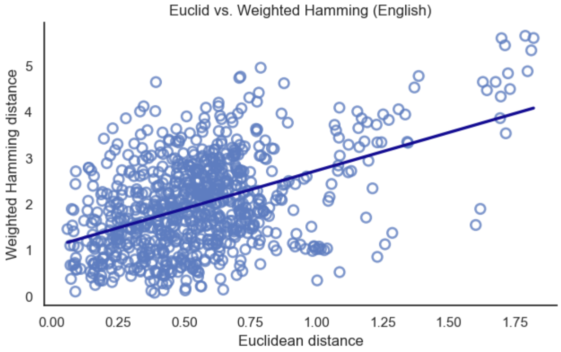


Figure 7: Correlation between Euclidian distance of embeddings and Weighted Hamming distance (English)

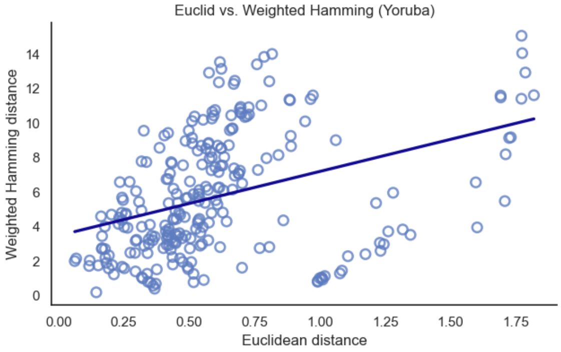


Figure 8: Correlation between Euclidian distance of embeddings and Weighted Hamming distance (Yoruba)

5 Discussion

5.1 Closer look on confusion and correspondence patterns

One of the pros of the conditional entropy metrics from Results[] for assessing transfer is that it does not need the predictions to be phonologically accurate—the only condition is that they are predicted unambiguously. The drawback of this is that low entropy could arise from systematic bias toward over-represented classes rather than accurate transfer. Let us consider in detail what phonologically non-trivial associations are found in alignments and how they reflect the expected differences between English and Yoruba discussed in Introduction[ref].

PMI To measure association we employ a symmetric measure related to the conditional entropies. Pointwise Mutual Information (PMI) computes how much it is not a coincidence for values of two variables to appear together. (**?**, p. 14) It is calculated by dividing the joint probability by the product of two marginal probabilities. Appendix[ref] In a cross-lingual transfer-learning setting, since two distributions do not have identical event space, we used the co-ocurrence matrix Results[ref] to find the joint and marginal probabilities. Phonemes vary greatly in their likelihood, therefore we normalize PMI by dividing it with negative log likelihood of the joint probability (the theoretical maximum of information content).(**?**, p. 61) NPMI shows what correspondences are statistically meaningful even when the raw counts are extremely low and is interpreted similarly to correlation coefficients. The values between -1 and 0 imply that the co-occurrence of the phonemes from two variables in alignments is more scarce than random, and the values between 0 and 1 suggest that two values co-occur more often than random, so a good transfer would have each class in predicted label show 1 with a certain Yoruba golden label, and in a trivial transfer the association would be with the closest by features.

Interpretation The first look reveals a strong grue diagonal in a lilac field: these are the correspondences from predicted phones to their logical Yoruba counterparts. No suspicious associations dwell in the off-diagonal areas, even among the least frequent phones: phonological closeness rather than noise accounts for the information-carrying phone pairs.

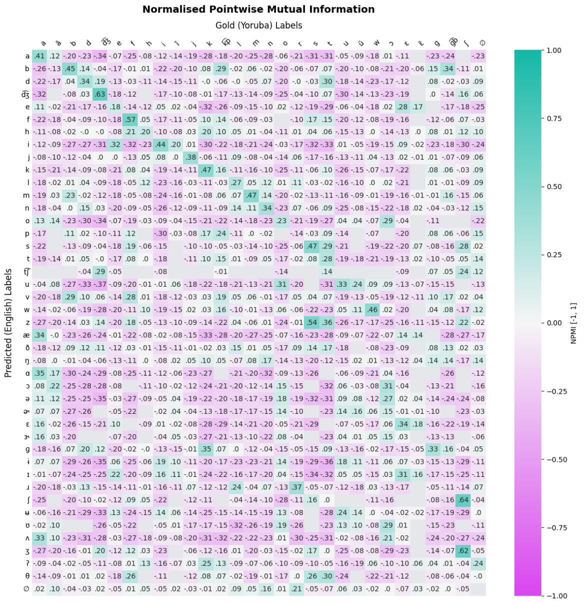


Figure 9: Normalized PMI. Turquoise denotes the encoding efficiency gain from aligning, magenta denotes the gain from not aligning. Grey stands for the alignments that have never occurred (we can assume negative values).

The rows and columns labelled Østand for alignments with deletion and insertion, correspondingly. All vowels strongly disassociate with being deleted. This must be due to the CV syllable structure in Yoruba (Transfer[ref] (3)). The model generates a lot of consonant clusters, in accordance with English phonotactics, which get deleted in order to align the vowels. The frequency of insertions, on the opposite, seems to be unrelated to the phoneme type. One strange exception is Yoruba /ɾ∼r∼ɹ/(see Trasfet[ref] (4)). The only *ad-hoc* explanation that comes to mind is that our model misses the Yoruba tap realisations, and they are short, or because it predicts a an English rhoticized vowel instead of it. English Phonemes similar to Yoruba's double-articulated stops, /p b k g/, are occasionally deleted, but not more than other consonants. This likely disproves the hypothesis derived from our cognition from Transfer[ref] (2), that double-articulated stops will be recognized as sequences. A frequent correspondence with labio-velar stops g͡b, k͡p unexpectedly, but logically turned out to be the labio-velar approximant w. Other labial consonants were also associated.

Generally all consonants without close place/manner counterparts show higher entropy. Thus, /θ/ is divided equally between /f,s,t/. The rather sonorant /ð,ŋ/ are pretty much random.

The glottal stop is split between /k, h/ and being deleted; /p/ is between /k͡p,k,f,b/. These are all examples of migrating to some sort of place/manner compromise. Contrast this with vowels: all vowels, regardless if are in Yoruba labels, have high entropy (see also ref Inference).

A clean tendency is that the VOT is transferred not one-to-one, but regularly: Yoruba lacks voiced fricatives, but when /z,ʒ,v/ appear in predictions, they consistently associate with their voiceless pairs, and sometimes with place pairs. Predicted voiced stops map equally to voiceless and voiced Yoruba stops, while voiceless always map to voiceless.

Liquids and nasals form a "connected cluster" of confusions; it is possibly but not necessarily related to their partial allophony in Yoruba, where nasality often spreads (see Yoruba to English [ref] (4)). In general, more sonorant phones in TIMIT have higher entropy here, and vowels are even more interrelated, slightly grouping by backness.

Yoruba nasal a is associated with o-like vowels, that's because those are allophones in Yoruba [see Datasets], and probably this means that it more often surfaces higher. Other nasal vowels associate with their oral bases (/ĩ/ with i-like), etc. but with weaker certainty. They get aligned with approximants more often: /ũ/ with /w/, and /ĩ/ with /j/.

5.2 How to benchmark our results

A challenge in evaluating our model and drawing conclusions about phonology lies in circular nature of alignments. We first use theoretical assumptions about phoneme closeness for our alignment algorithm. Then we treat these alignments like the actual correspondence (e.g., as if the model assigned mentioned the predicted label exactly during the time span of the golden Yoruba label) to speculate which sounds the model was not able to recognise correctly. These alignments are not in fact related the traceable causal/computational link between in- and output during inference. In a way, the alignment being theoretically sound (ref PMI) is not surprising, because they are obtained via theoretical edit distance.

Refer to Appendix[ref] for examples of alignments. Impressionistically, although the model output is clearly linked to the input, the alignments are not always intuitively correct. (show) Indeed, they are proven minimal given the feature model, but it does not guarantee to correspond to a linguists intuition. This is a common issue in the pure feature-based *Lev* alignments (Dellert, p.c.). More traceable options include alignment with some kind of supervised training on a human-aligned data. **?** reports good alignment results with log-transformed feature weights. Another option is to embed some pre-knowledge of phonotactics in the cost functions by assigning weights to different features: say, to improve resolution of consonant clusters, knowing the Yoruba **CVN** syllable structure, we can make deleting consonants cheap, while changing consonants to vowels costly.

Combining feature model with minimal distance leads to the overall fwPER being hard to interpret. For example, we do not know the expected cost for two random phonemes, so, unlike in the regular PER, there is no intuition such as "every 3rd letter is wrong". It is also not clear how much minimizing the alignment lowers the fwPER on average.

One theory-light way we considered is to benchmark our model's performance was to compare it with the fwPER from evaluating random strings. We evaluated 3 different randomly generated "predictions" with the same cost settings as in Results(ref).

First, we took the exact lengths of the sentences from our predictions, to simulate some understanding of time, then we tested: (1) generating random sequences of TIMIT labels of that length, using equal probabilities ("nulligrams") (2) generating that using the unigram probabilities of the TIMIT labels in our predictions. This way we had the same predictions, but filled with random letters, unrelated to the audio input whatsoever. (3) making a classic bigram model based on actual predictions, generating strings of random length until the EOS symbol.

The PER with the first two random sequences is surprisingly close to ours: `0.4` and `0.39`. The difference from the model's PER the distribution and median (Figure **??**) is worth noticing: in predictions the positively-skewed bell curve has longer tails on both sides; the random alignments gravitate more towards thee average. (Figures **??**) The intuition allows us to say that the model predictions are not random: it seems, the random simulations never achieve low PER values like `0.2`to `0.3`. A qualitative look at the alignments in appendix [ref] suggests that the prediction outcomes are better and linked to the input sound.

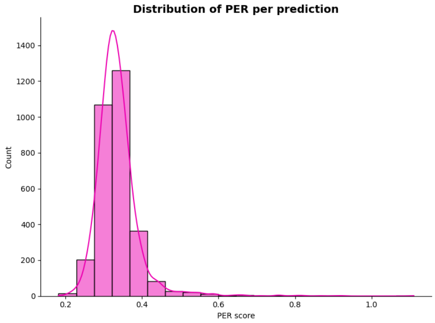


Figure 10: PER Distribution

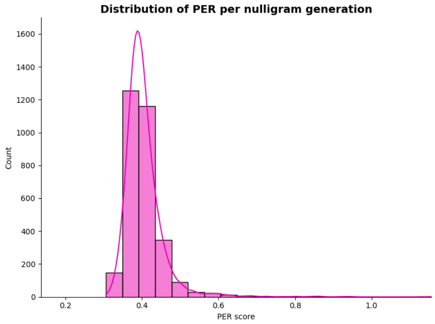


Figure 11: Equal probability PER Distribution

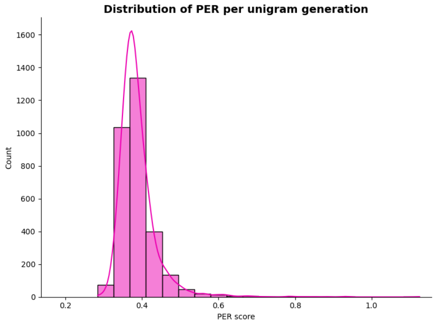


Figure 12: Unigram PER Distribution

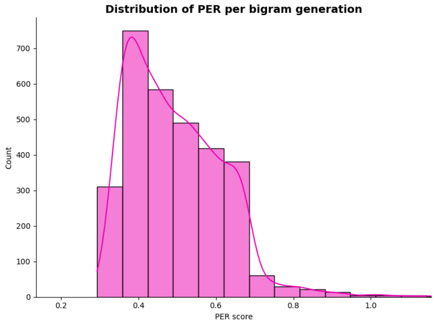


Figure 13: Bigram PER Distribution

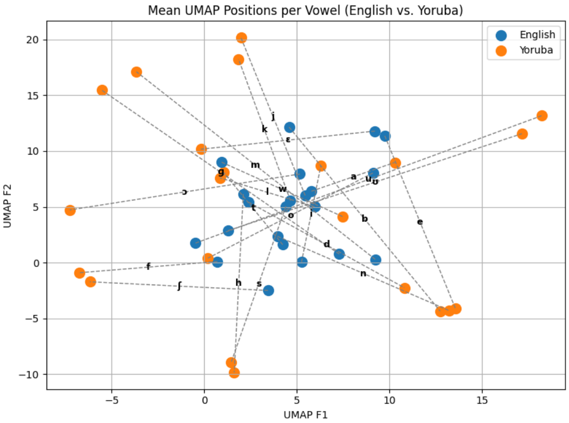


Figure 14: Vowel distances between English and Yoruba

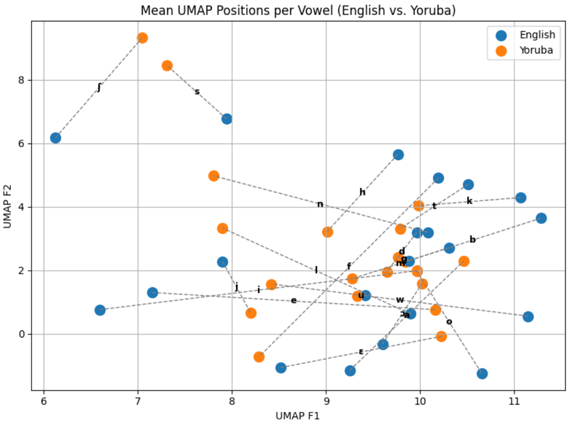


Figure 15: Yoruba and English consonant embeddings plotted in the same space

Table 3: Pearson correlation between vowel distance and another variable

Language	Correlation Coefficient (r)	p-value
English	0.47	0.0002
Yoruba	0.49	0.004

Table 4: Vowel distances between English and Yoruba (sorted by similarity)

Vowel	Distance
u	0.095
g	0.170
k	1.119
d	1.280
m	1.424
t	1.571
j	1.633
s	1.800
ɔ	1.947
ɛ	1.971
b	1.993
h	2.549
n	2.887
w	2.907
e	3.059
ʃ	3.278
o	3.291
l	3.363
i	3.590
a	3.643
f	5.945

Limitations

For future reference, we believe that our model could be further improved by dedicating more attention to the over-generation problem, and possibly address it at the pre-training stage with a modified pre-training objective (or simply a denoising procedure at the data preprocessing stage), or by investigating whether there are some repeating patterns linked to over-generation. This would then in turn help improve the quality of alignment for the evaluation step. Moreover, having datasets that agree on the granularity on the description would be helpful, as this would facilitate the mapping and reduce the risk of inducing biases. Although the pre-training stage should in principle be as general as possible, introducing phonotactic constraints through a LM model would likely also contribute to reduce noise during generation. In regards with the Common Voice dataset specifically, we can also mention the size of the corpus in comparison to the English one, since the 0.25 split we used for English for evaluation is almost twice the size of all the Yoruba splits combined. Moreover, although we deem Epitran to be suitable for the G2P step (its paper mentions it being competitive with the FST-based baselines for ASR), it would be interesting to see whether a more-linguistically informed model, perhaps a WFST would yield a different performance. Another direction we could was to use the embeddings obtained to see whether they latently contain some information about Yoruba which was left out from the annotations we used. For instance, it would be interesting to see whether tone can be accurately predicted by feeding the final representations into a classifier, investigating whether the architecture was sufficient for the model to internalize this additional information.

We of course encourage the reproduction of this study on different datasets, be it for Yoruba or other languages.

Conclusion

We trained an audio to IPA ResNetBiLSTM CTC recognizer on TIMIT and evaluated it zero-shot on Yoruba with a feature-weighted PER, reaching near-perfect figure of 0.0339 for English validation set. (KIT: add more) Tackling the problem of different inventories, we utilized a modified version of feature-weighted Levenshtein algorithm for evaluation. The system reaches `0.34` PER on Yoruba (median 0.30), among which over-generation error dominates. Limiting model capacity by decreasing hidden-dimension and making model to be shallower greatly improved cross-lingual generalization from English to Yoruba. The depth of our model, which is 19, is less than half of that of the original model after reducing the number of residual blocks from 5 to 3, convolution computation within residual unit from 2 to 1, and dense layers in the end. We also dramatically diminished the model size from approximately 1.55 million to 0.8 million. This constraint prevented over-fitting to the TIMIT phoneme distribution and forced the encoder to extract lower-entropy, articulation-level representations. The resulting model preserved contrasts in manner and place of articulation while smoothing over language-specific refinements in vowel space, consistent with the (N)PMI and entropy analyses provided in previous sections. Another important part of the training was CTC loss function. CTC marginalizes over all monotonic alignments between input frames and label sequences, using a dedicated blank symbol to enable many-to-one and skip alignments. Considering over-generation of rare phonemes, which we suspect to reflect extra-linguistic noise, however, the way of processing pauses deemed to be important. We have deleted pause labels from both English and Yoruba based on the fact that our task does not spot word boundaries, so that assuming pause removal will not undermine the performance. Additional noise reduction step as preprocessing for Yoruba would have yielded better achievements, since noise appearing before and after the utterance was unique to Common Voice Yoruba not TIMIT.

Analysis of confusion, mapping entropy, and (N)PMI shows that transfer is phonetic rather than random: NPMI table revealed that phonetic diplomacy is captured. The network aligns by feature similarity, negotiating between two phonological systems. The model successfully transferred cues of acoustic similarity, but over-differentiates distinctions that Yoruba neutralizes, where numerous English classes cut through the acoustic spaces of Yoruba. To note, American English has 7 vowel contrasts while the transcription distinguishes 15 acoustically similar allophones. We indeed merged redundant distinction in TIMIT labels, however, coalescing English label classes to a greater extent, making it more aligned with Yoruba labels is expected to enhance the transfer. Correspondences track place/manner similarity, voicelessness transfers more reliably than voicing (due to VOT differences), labio-velar stops often surface as /w/, and Englishs finely split vowel space and overly narrow transcription tradition collapses toward Yorubas coarser categories. In short, the transfer is smart about sounds, but tone-deaf. Transfer keeps the tendency for consonant clusters, systematically breaking the syllable structure of Yoruba. PER results with different n-gram settings indicates a circularity: alignments are theory-driven (based on feature phonology) and not causal chain-driven, which biases the further conclusions about phonology into confirm pre-existing notions(thus, in turn, fixing model's tendency to over-generate in noisy circumstances is the most relevant way to reduce the error). This nature of the alignments remains a problem, as long as the alignments are not checked for the link through the time-aligned test set or for intuitiveness by supervised training or curation. The fwPER also lacks the easy interpretation of PER. Nonetheless, random baselines neither match the low-PER tail nor the skew of model outputs, and bigram generations without length control explode in PERevidence that predictions are signal-bearing and that length control is critical.

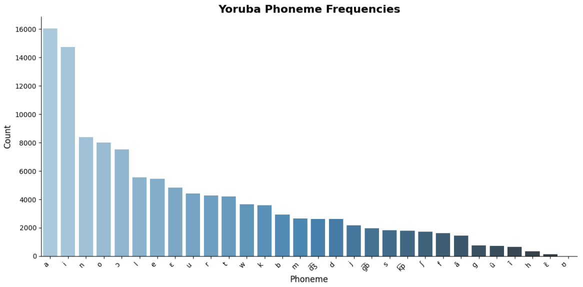


Figure 16: Original Yoruba Dataset Phoneme Frequency

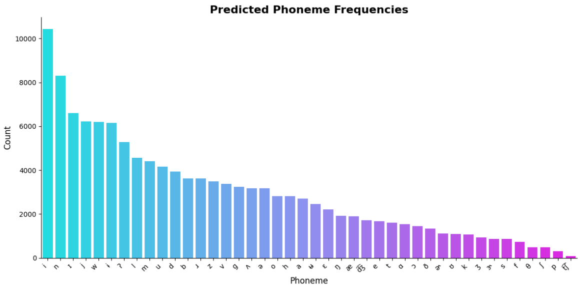


Figure 17: After Evaluation Phoneme Frequency

Moving forward, the clearest gains lie in (i) controlling insertions (denoising, decoding constraints, or a light phonotactic LM), (ii) inventory simplification on the source side (merge the gradient vowel/voicing contrasts, redundant in English, keep the discrete consonant contrasts) to sharpen correspondences, and (iii) better alignments (supervised or phonotactic-aware costs) and harmonized annotation granularity. Finally, the learned embeddings invite probing for tone and other Yoruba-specific information absent from the labels.

A Appendix

A.1 Appendix. A

A.2 Appendix. B

A.3 Appendix. C

Formula for the substitution cost of two IPA vectors from PanPhon inventory: English $\vec{e}$ and Yoruba $\vec{y}$.

$$\mathrm{dist}(\vec{e}, \vec{y}) = \frac{\left|\{\, i \mid (e_i \neq y_i) \wedge (e_i \neq 0 \vee y_i \neq 0) \,\}\right|}{\left|\{\, i \mid e_i \neq 0 \vee y_i \neq 0 \,\}\right|}$$

Table 5: TIMIT to IPA mapping (folded into two columns)

TIMIT	IPA	Our	TIM	IPA	Our
aa	ɑ	ɑ	ch	t͡ʃ	t͡ʃ
ae	æ	æ	d	d	d
ah	ʌ	ʌ	dh	ð	ð
ao	ɔ	ɔ	dx	ɾ	r
aw	aw	a + w	el	l̩	l
ay	aj	a + j	em	m̩	m
ax	ə	ə	en	n̩	n
axr	ɚ	ə	f	f	f
eh	ɛ	ɛ	g	ɡ	ɡ
er	ɝ	ɝ	hh	h	h
ey	ej	e + j	h	h	h
ih	ɪ	ɪ	jh	d͡ʒ	d͡ʒ
ix	ɨ	ɨ	k	k	k
iy	i	i	l	l	l
ow	ow	o + w	m	m	m
oy	ɔj	ɔ + j	n	n	n
uh	ʊ	ʊ	nx	ɾ̃	n
uw	u	u	ng	ŋ	ŋ
ux	ʉ	ʉ	p	p	p
ax-h	ə̥	ə	q	ʔ	ʔ
bcl	b̚	b	r	ɹ	r
dcl	d̚	d	s	s	s
eng	ŋ̍	ŋ	sh	ʃ	ʃ
gcl	ɡ̚	ɡ	t	t	t
hv	ɦ	h	th	θ	θ
kcl	k̚	k	v	v	v
pcl	p̚	p	w	w	w
tcl	t̚	t	wh	ʍ	ʍ
pau	|	–	y	j	j
epi	||	–	z	z	z
h#	/	–	zh	ʒ	ʒ
b	b	b			

Table 6: Yoruba IPA inventory

Yoruba	IPA	IPA (adjusted)
m	m	m
i	i	i
k	k	k
y	j	j
u	u	u
a	a	a
w	w	w
n	n	n
t	t	t
l	l	l
s	s	s
b	b	b
e	e	e
o	o	o
g	ɡ	ɡ
h	h	h
d	d	d
r	ɾ	r
f	f	f
ẹ	ɛ	ɛ
ṣ	ʃ	ʃ
ọ	ɔ	ɔ
j	d͡ʒ	d͡ʒ
´	˦	–
`	˨	–
in	ĩ	ĩ
un	ũ	ũ
gb	ɡ͡b	ɡ͡b
p	k͡p	k͡p
ọn	ɔ̃	ã
ẹn	ɛ̃	ɛ̃
an	ã	ã
–	˧	–
n	ŋ	–
ụ	ʊ	–
ị	ɪ	–

(?)

Formulae for confusions. Formally, let Y denote the gold Yoruba phoneme and $\hat{Y}$ the predicted TIMIT phoneme var. Each row of the confusion matrix is a probability distribution over predicted labels, conditioned on a given Yoruba label.

$$P(\hat{Y}=j \mid Y=i)=\frac{|\{(x : Y(x)=i, \hat{Y}(x)=j)\}|}{|\{(x : Y(x)=i)\}|},$$

Then the confusion entropy is $H(\hat{Y} \mid Y=i) = -\sum_j P(\hat{Y}=j \mid Y=i) \log_2 P(\hat{Y}=j \mid Y=i)$.

The metric that we call mapping entropy is the opposite. With the posterior probabilities of gold labels given a certain prediction

$$P(Y=i \mid \hat{Y}=j)=\frac{|\{(x : Y(x)=i, \hat{Y}(x)=j)\}|}{|\{(x : \hat{Y}(x)=j)\}|}$$

the mapping entropy is $H(Y \mid \hat{Y}=j) = -\sum_i P(Y=i \mid \hat{Y}=j) \log_2 P(Y=i \mid \hat{Y}=j)$. With the expectation $H(Y \mid \hat{Y}) = \sum_j P(\hat{Y}=j) H(Y \mid \hat{Y}=j)$, and perplexity $P(Y \mid \hat{Y}) = 2^{H(Y|\hat{Y})}$. (?)

Table 7: Some of the worst predictions (Pred vs. Gold).

	Pred	Gold
W1	ʔ ɪ t ð ə ʔ ɪ ə b i p ɨ ɪ n d u ə b ɨ ɹ ɛ n ə d ɨ ɹ ɪ k ɨ ɨ t ɪ b ʉ p l ɪ n ɨ n t l ʊ d ʉ ɨ ɹ ɹ ɪ t ɨ ŋ b ʉ d b l ɨ b ʔ e j ɹ t ʉ z d i d d n g ɨ n p ɨ p	g͡b o g͡b o a w ã d ɔ k i t a t i k͡p ĩ u l a t i w a ɔ n a a b a j ɔ s i a a r u n
W2	ɔ w a w v ə e ʔ j z ʉ w ɪ m ʉ ɹ u u ʔ j i ɹ i ɪ z z ɪ ɪ ə z ʉ ɪ ɨ g u d l ʌ ʌ ʔ ɪ n ɹ i ɪ ɪ z ʉ ɹ h j o	i k u n i e r e ɛ ʃ ɛ f u n ɛ̃ i t o b a k͡p a e e j a n
W3	n ɛ ə ʔ ə t æ ʔ ɛ ʔ ə ɹ n p ʊ ɪ t ɪ m p ɝ m ɨ ɹ ɨ z b ɹ a j t o ɨ n t a t o w t ə ɹ ɨ p ɨ n ɛ ɹ o w u t ɛ t ɹ ɨ ɹ n ə ʔ æ t k w h i p ɨ n b ə p ɹ ʉ z k ʉ n z ɨ w ɨ d ɨ ŋ ɨ p n æ t ɪ n k ɨ t ə ə n	o l u ɔ m ɔ l o f a r a g͡b a ɔ b ɛ f u n a a r ɛ n i b i i k͡p o l o n g o i b o

Table 8: Two best predictions

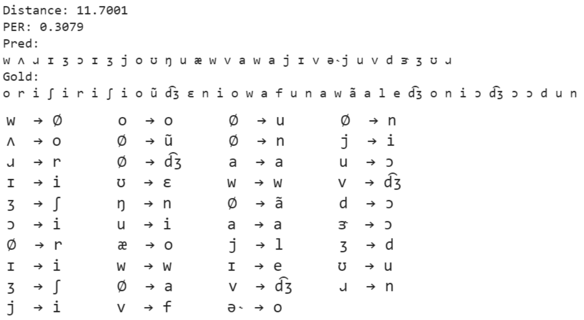
Distance	6.9
Pred	u i l e j d i ɑ d u l a w ə m æ d ɑ l æ l i ɹ a j ɨ z s ɔ i ɹ u b ɑ m
Gold	o r i l ɛ e d e a d u l a w ɔ m ɛ t a l a l o m a a n s ɔ e d e j o r u b a
Distance	6.0
Pred	o l m i ɑ j ɹ i b i ʌ l w k ʌ b ʌ d͡ʒ u l ɪ b ʊ s i i b ɔ w u
Gold	o m i j a l e n i ɔ l ɔ k͡p a a r ɔ d͡ʒ ɔ i b ã s i i g͡b o h o

Formula for normalized PMI for a gold sound y and predicted sound e in the co-occurrence counts matrix obtained from alignments.

$$\text{nPMI}(y, e) = log_2(y, e) - log_2(y) - log_2(e)$$

(?)

A.4 Appendix D



```
Distance: 11.7001
PER: 0.3079
Pred:
w ʌ ɹ ɪ ʒ ɔ ɪ ʒ j o ʊ ŋ u æ w v a w a j ɪ v ɚ j u v d ɝ ʒ ʊ ɹ
Gold:
o r i ʃ i r i ʃ i o ũ d͡ʒ ɛ n i o w a f u n a w ã a l e d͡ʒ o n i ɔ d͡ʒ ɔ ɔ d u n

w  → ∅      o  → o      ∅  → u      ∅  → n
ʌ  → o      ∅  → ũ      ∅  → n      j  → i
ɹ  → r      ∅  → d͡ʒ     a  → a      u  → ɔ
ɪ  → i      ʊ  → ɛ      w  → w      v  → d͡ʒ
ʒ  → ʃ      ŋ  → n      ∅  → ã      d  → ɔ
ɔ  → i      u  → i      a  → a      ɝ  → ɔ
∅  → r      æ  → o      j  → l      ʒ  → d
ɪ  → i      w  → w      ɪ  → e      ʊ  → u
ʒ  → ʃ      ∅  → a      v  → d͡ʒ     ɹ  → n
j  → i      v  → f      ɚ  → o
```

Figure 18: Automatic alignment of the median prediction