Data Constrained Phonetic-Based Automatic Speech Recognition for Low Resource Languages

Kyle Ng

Abstract

With the advances in Automatic Speech Recognition (ASR) for high resource languages, this paper aims to open the door to understanding the magnitude of data needed for speech recognition tasks. In this paper, we compare Wav2Vec2XLSR (56K hours of data) against Wav2Vec2IPA, a fine tuned Wav2Vec2-based architecture model trained on 5 hours of data from the TIMIT speech corpus (0.5% of baseline training data). From our evaluations, we find that the baseline model outperforms the smaller data model by 8.3 percentage points with respect to Phoneme Error Rate. In addition, to address a gap in the availability of the TIMIT speech corpus, we also introduce 2 new datasets on HuggingFace Hub, `timit_asr` and `timit_asr_ipa`. Additionally, this research on smaller models intends to address current issues in Natural Language Processing (NLP) surrounding environmental sustainability and democratizing access to AI with the advent of small machine learning models.

1. Introduction

While Automatic Speech Recognition (ASR) from audio to text is quite robust for high resource languages, lower resource languages (<10 hours of labeled speech data) make universal multilingual ASR difficult to scale.

Thus, this paper introduces a new model Wav2Vec2IPA fine-tuned on International Phonetic Alphabet (IPA) transcriptions of the TIMIT speech corpus aiming to understand if audio to IPA ASR pipelines can aid cross-lingual transfer in search for universal multilingual ASR.[1]

To evaluate model performance, we compare our new model against a multilingual Wav2Vec2 variant, Wav2Vec2XLSR. Since the TIMIT dataset uses a different phonetic transcription than IPA, we devised a preprocessing step which aligns TIMIT phonetic transcriptions to the IPA output of this Wav2Vec2 variant's vocabulary.

Rather than using the traditional Word Error Rate (WER) metric, we evaluate our model on a Phoneme Error Rate (PER) and minimum edit distance.

Our results show that on <5 hours of speech data, Wav2Vec2IPA achieves a PER of 0.373 compared to Wav2Vec2XLSR's PER of 0.290 which is trained on 56K hours of data (Ardila et al., 2019). Despite underperforming the baseline by 8.3 percentage points, this work helps to illustrate challenges when working with low resources languages and future approaches and strategies to utilize emerging techniques such as cross-lingual transfer.

Code: https://github.com/Ky-Ng/IPA-ASR

2. Background

2.1. Phonetics and Phonology

Linguistic theory (phonetics and phonology) proposes that the production and interpretation of sound units (phonemes) are universal across all languages. In other words, sounds in specific languages can be mapped to a language agnostic representation that are shared between many languages.

Thus, the goal of this project is to train the base Wav2Vec2 model to learn a universal transcription which could be robustly applied to a novel language the model has not seen.

Thus, we specifically avoided the use of a Language model on top of Wav2Vec2 such as implementations in Wispr or Wav2Vec2-BERT in order to avoid depending on the word co-occurence in a lanaguage which may not be consistent cross-linguistically.

3. Related Work

3.1. Wav2Vec2

The Wav2Vec2 Model(Baevski et al., 2020) released by Meta AI introduced learning latent speech representations directly from raw waveforms to robust latent speech representations. These representations can be fine-tuned for a variety of downstream tasks such as audio classification or

[1]For this paper, we use the TIMIT speech corpus which is an English-Only corpus with very limited speech data as a proof of concept for working with other low resource languages

ASR.

Wav2Vec2 feature extraction from raw waveforms is replaced the traditional Mel-Frequency Cepstral Coefficients (MFCCs) derived from Discrete Fourier Transforms (DFT).

Wav2Vec2 first encodes raw waveforms using a multi-layer CNN followed by a 12 deep transformer block trained on a contrastive loss task (differentiating between phonemes).

To handle phoneme alignment, the Connectionist Temporal Classification alignment scheme is used.

3.2. Wav2Vec2XLSR

Wav2Vec2XLSR (Cross Lingual Speech Recognition)(Conneau et al., 2020) builds upon Wav2Vec2 by simultaneously training latent speech representations with multilingual IPA transcriptions.

The architecture is almost identical to Wav2Vec2 and uses a shared CNN and transformer encoder across audio inputs from 53 languages across 10 language families (Common-Voice, BABEL, Multilingual LibriSpeech).

In total, Wav2Vec2XLSR uses approximately 56K hours, of which 1,087 hours are English[2].

4. Dataset and Evaluation

4.1. Dataset

We fine-tuned the Wav2Vec2-Base model on the TIMIT[3] database which contains <5 hours of read speech with phoneme level transcriptions sampled at 16kHz. We used a 80-10-10 Train-Validation-Test split.

Wav2Vec2-IPA uses 0.5% of the volume of data relative to the baseline model to simulate the training on an endangered or low resource language.

4.1.1. Data Preprocessing Overview

Since the database was created in 1993 at a time where IPA symbols could not be digitalized, this paper introduces a phonemic transcription of the 1993 TIMIT phonetic symbols to current 2024 IPA transcription.

The data preprocessing is broken down into 3 main steps consisting of (1) Creating a TIMIT to IPA Lookup Table, (2) Applying the Lookup to the Train/Test datasets, and (3) Stratifying the Test Dataset on the Speaker Sex and Dialect Region attributes to create a balanced Validation dataset.

[2]780 hours of the 1,087 (approximately 70%) of the transcription are verified

[3]Texas Instrument Massachusets Institute of Technology TIMIT on Hugging Face

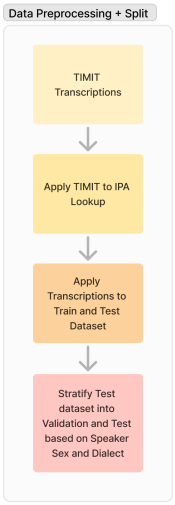

Figure 1. Phoneme Error Rate vs. Training Step

4.1.2. 1. Creating the TIMIT to IPA Lookup

As the TIMIT database was made to capture maximal variance accross American Dialect Regions, the TIMIT database contains phonetic detail on allophones (phonemes in free variation[4]). Thus, the main goal for the lookup table is to map allophones to their corresponding phonemes[5].

Although these allophones may be phonemic (meaning differentiating) in other languages, our focus on the constrained English speech corpus makes it possible for us to ignore these differences.

Below is an example table of a mapping between the TIMIT phonetic detail and the IPA transcription we chose[6].

Table 1. Sample of the proposed translation system between the 1993 TIMIT and 2024 IPA transcription

TIMIT	IPA
tcl	*t*
tcl t	t^h
ng	ŋ
aw	*aʊ*

[4]When a phoneme or sound is in "free variation", the term refers to the idea that speakers may produce a sound from a distribution of interchangeable sounds based on the speaking environment without changing the meaning of the word

[5]these phonemes can be thought of as a base sound whose distribution during speech production are the allophones

[6]The full lookup can be found in `timit_ipa_translation.py`

4.1.3. 2. Applying Lookup Table

After applying the lookup table to eliminate allophones, the main issue comes from the TIMIT transcription of "aspiration"[7]. In the table above, the tcl / t is "unaspirated" whereas a tcl t / t^h is aspirated.

While aspiration does not express phonemic difference in English, we could not immediately replace all tcl with t and tcl t as t^h because there were cases where a only t was present without a tcl.

To understand whether or not a standalone t should be treated as a tcl or tcl t, we viewed these specific cases using a waveform and spectrogram in Praat which revealed that these standalone t are unaspirated.

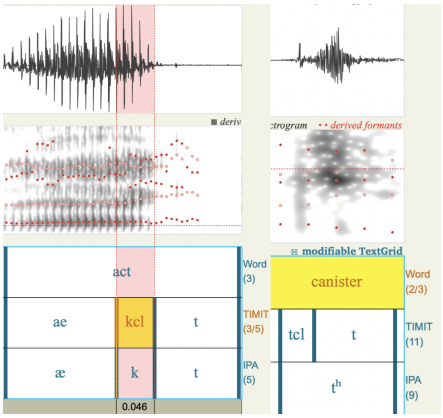

Figure 2. Unaspirated (Left) vs. Aspirated (Right)

However, after viewing the transcriptions from Wav2Vec2XLSR, the baseline model does not apply aspiration differences in its transcriptions. Thus, although we transcribe the TIMIT speech corpus with aspiration differences, a sanitization step later removes the allophone t^h and replaces it with t.

4.1.4. Stratification for Test and Validation Dataset

Originally, we used the HuggingFace `timit_asr` implementation of the TIMIT dataset. However, the HuggingFace implementation of the speech corpus (1) does not have the audio files built-in (requires downloading the audio files from a 3rd-party zip file), (2) does not utilize all of the data available in the corpus (50 minutes of training data and 17 minutes of testing data with only 4 of 8 dialect regions represented), and (3) does not include a Validation dataset (the original TIMIT speech corpus contains only train/test).

To address each problem, this paper contributes two open source implementation of the TIMIT ASR on HuggingFace Hub `timit_asr` and `timit_asr_ipa`.

The `timit_asr` follows addresses problems (1) and (2), making the HuggingFace audio downloadable directly from the Hub and expands the corpus to include all speakers and dialect regions.

The `timit_asr_ipa` builds on top of the new `timit_asr`, adding the IPA transcription mentioned in the previous step as part of the dataset out of the box. In addition, this dataset also splits the original Test dataset in half to create a Test and Validation dataset. However, the TIMIT speech corpus is recommended to follow specific demographic distributions. To preserve this data distribution split, we stratify on the speaker sex and dialect region attributes to create an 80-10-10 train-test-validation split for a total of 187 minutes (3629 examples), 35 minutes (670 examples), and 34 minutes (670 examples) respectively.[8]

4.2. Evaluation

This paper uses 2 main methods for evaluation: (1) Minimum Edit Distance[9] and (2) Normalized Phoneme Error Rate

We use Phoneme Error Rate rather than the traditional ASR Word Error Rate since the models we are looking to evaluate are on a character level rather than a word level[10].

4.2.1. Edit Distance

We use edit distance as a naive[11] metric to understand how far away two transcriptions are.

$$\textbf{Edit Distance} = \textbf{substitutions} + \textbf{deletions} + \textbf{insertions}$$

4.2.2. Phoneme Error Rate

Secondly, we consider phoneme error rate or edit distance normalized by number of phonemes.

$$\textbf{PER} = \frac{\textbf{Edit Distance}}{\text{len(ground truth transcription)}}$$

However, PER over-penalizes model transcriptions which are much longer than the ground truth prediction:

[7] Aspiration is the small puff of air or "release" from a stop consonant such as the "t" or "p" sound in "talk" "poll"

[8] Note: as per TIMIT recommendation, the Training, Test, and new Validation dataset do not contain any overlap in speakers

[9] Using the Levenshtein Distance

[10] In fact, the notion of a word becomes quite difficult to quantify when trying to generalize the model cross-linguistically

[11] We consider edit distance as naive since we are not able to generalize what type of mistakes the model is making and is not normalized to a specific length. For example, a model which produces longer sequences are more likely to have increased edit distances.

Case 1) Truth = ”ABC” and Prediction = ”ABC12345”

$$PER = \frac{\textbf{Edit Distance}}{\text{len(”ABC”)}} = \frac{5}{3} = 1.667$$

Case 2) Truth = ”ABC12345” and Prediction = ”ABC”

$$PER = \frac{\textbf{Edit Distance}}{\text{len(”ABC12345”)}} = \frac{5}{5} = 1.0$$

In order to make our PER metric symmetric with respect to truth and prediction, we use normalized PER[12]:

$$\textbf{PER'} = \frac{\textbf{Edit Distance}}{\textbf{max}(\text{len(truth transcription), len(pred. transcription)})}$$

4.2.3. Loss Function: Connectionist Temporal Classification

For the loss function we used to train the transformer layers in Wav2Vec2 we use the Average Connectionist Temporal Classification Loss which uses a Recurrent Neural Network (RNN) to handle sequence alignment between the prediction and truth prediction.

5. Methods

5.1. Model Architecture

Both Wav2Vec2XLSR and Wav2Vec2IPA are based on the Wav2Vec2 architecture. Rather than relying on MFCCs like traditional signal processing modells, Wav2Vec2 uses a CNN trained on a contrastive learning task to transform raw speech signal into a feature embedding. This embedding could carry the phonological properties as the deep underlying structure that we desire to learn in order to perform ASR.

After the audio wave files are embedded by the CNN, the representations are transformed again using the transformer architecture which is then returns logits. These logits are then decoded into their most likely token index.

Lastly, these tokens are compared against the ground truth token transcription using the CTC Loss (RNN based) architecture.

[12]A nice property is that normalized PER is normalized between [0, 1]

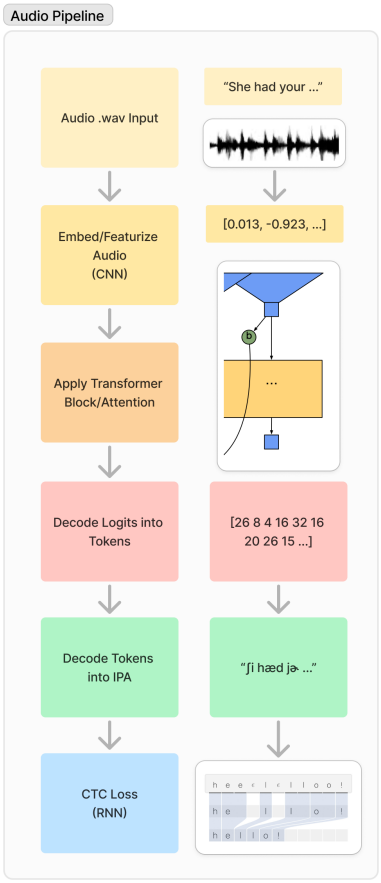

Figure 3. Nerual Network Architecture: Audio to Decoded Output

5.2. Hyperaparmeters

In this paper, the main hyperparameters considered are (1)

1. Train-Test-Validation Split Percentage
   - After increasing the amount of data from 1 hour to 5 hours, we could decrease the percentage of the Validation and Test dataset while still having representative(stratified) data to compare against during training
2. Learning Rate
   - Since the model seemed to overfit at 5 epochs, we were motivated to decrease the learning rate from 10^{-2} to 10^{-4}
3. L2 Regularization
   - Since the model tended to overfit quite early, we applied L2 regularization to make encourage the weights to be much smaller
4. Number of Epochs

- Originally, we had the epochs set to 100 which was extremely time intensive and prone to overfitting.
- Thus decreasing the epochs to be 20 helped maintain a low validation-train loss gap

The improvement we can see from the previous methods of Wav2Vec2IPA come from choosing hyperparameters which help to decrease overfitting on the training dataset and increasing the amount of training data to help generalization.

6. Experiments

6.1. Baseline Results

For Wav2Vec2XLSR, we report an average edit distance of 10.2 and an average PER of 0.285.

To inspect cases of especially high Edit Distance or PER, we visualize errors using a Sequence Alignment. This helps us understand what kind of phonemes the model is most prone to.

```
ʃihɛd-dʒɛ˞--darksut?ɪŋgr---isɪ-warʃwaɪɚʔaljɪɚ
|||   |     || |||| |      ||  || || || ||||
ʃih--æd---jɚda-ksut-ɪ---ŋgɹis-iwa-ʃw-ɹɚ-aljɪ-
  Score=23
```

Figure 4. Sequence alignment for "She had your dark suit in greasy wash water all year". The sequence alignment score represents the maximum number of aligned phonemes given the minimum edits to each sequence

For this specific example, we can see how the vowels which differentiate "ae" vs "eh" and the extra consonant "r" in the truth transcription is not picked up by the baseline models. By listening to the audio for this specific example, we can start to find a generalization that the baseline model does not perform well on Dialect Region "New England".

6.2. Wav2Vec2IPA

6.2.1. 1 Hour Dataset

For our trained model, we used Wandb to track the PER and Connectionist Temporal Classification error over training epochs.

First, we start training off at 100 epochs at a learning rate of 10^{-4}

The plot above shows extreme overfitting at training step 1500 which translates to epoch 4.

To combat this overfitting, we tried to increase the learning rate from 10^{-4} to 10^{-2} to possibly "jump" out of potential local minima. However, with such a drastic increase in learning rate, the model is underfitting and unable to converge for training loss. Above we can further see how the Phoneme Error Rate stays relatively high (close to 1) throughout all training.

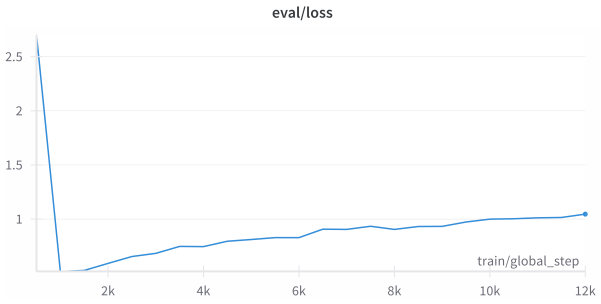

Figure 5. Validation loss vs. Training Steps (1 Hour Dataset)

6.2.2. 5 Hour Dataset

Before tuning different hyperparameters, we first tried keeping the same setup as in the ¡1 Hour Dataset.

Using the new dataset, we found that we would overfit at step 12,000 rather than step 1,500 with a validation loss at around 0.4. Unlike the 1 Hour dataset which seemed to increase in loss just from pure chance, the larger dataset does not experience as sharp of an upward trend of validation loss but does start to overfit much more at the 20,000 step mark.

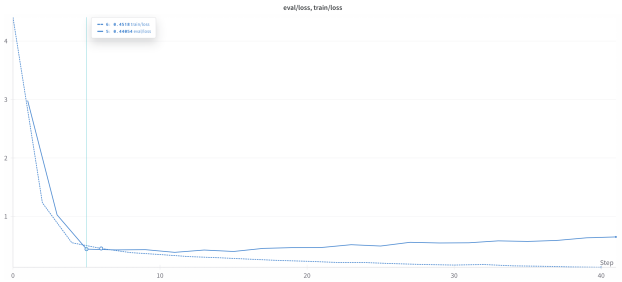

Figure 6. Validation loss vs. Training Steps (5 Hour Dataset)

However, even after tuning the L2 Loss, Learning Rate, Epochs, and Dataset split, the Character Error Rate maintained to be consistent indicating that the gating factor on improvement is the amount of data the model has to learn from. Below are hyperparameter selections of changing a specifc hyperparameter in isolation[13].

[13]Note that the 5 epochs is compared to 15 epochs which were used in the Learning Rate = 10^{-4} and No L2 Regularization

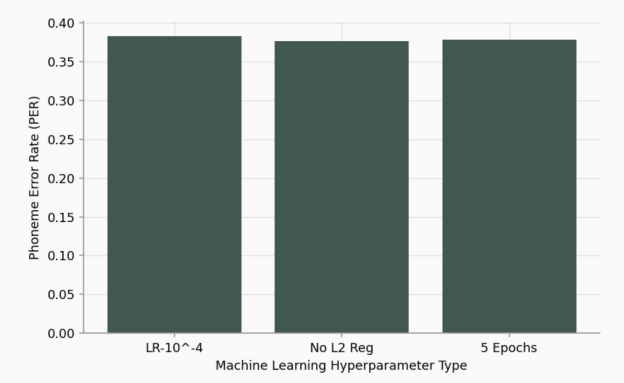

Figure 7. Phoneme Error Rate vs. Hyperparameter Selections

7. Discussion

While evaluating the Wav2Vec2XLSR vs. Wav2Vec2IPA model, it is clear that the baseline (PER of 0.285) outperforms the much smaller dataset model (0.377). However, the specific errors the Wav2Vec2IPA model makes are quite robust and promising.

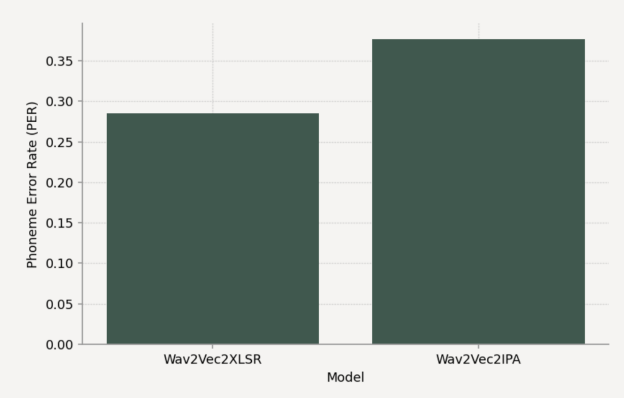

Figure 8. Wav2Vec2XLSR vs. Wav2Vec2IPA for (PER)

Below, we can see that for the phonemes which the model outputs, the transcriptions are largely correct. Rather, the underscores, indicating an *UNK* or unknown token is what contributes to the higher Character Error Rate.

```
Ground Truth: doʊnæskmiɹɪkɛrɪɪnəʔɔɪlɪræglaɪkðæt
Prediction:   doʊ_æ_k_iɹɪkɪ_iɪ_əʔɔɪ_i_æg_aɪkðæt
```

Figure 9. Wav2Vec2IPA Transcription with UNK tokens

Although the 5 Hour data model performs better than its 1 hour counterpart, it seems that blocker for PER improvement is having more data.

Future endeavors should seek to understand more about specifically how much data is needed to improve a base model like Wav2Vec2Base to perform on par with a baseline model trained on thousands of hours of data[14].

8. Conclusion

In conclusion, in this paper, we created a pipeline for translating the traditional TIMIT transcription from 1993 into the modern 2024 IPA transcriptions. From there we evaluated two versions of Wav2Vec2, the baseline model Wav2Vec2XLSR and a fine-tuned Wav2Vec2 model. During this processes we fine-tuned the base model, Wav2Vec2IPA, on 5 Hours of speech (0.5% of the data used to train the baseline model).

Although the baseline outperformed Wav2Vec2IPA, this paper hopes to open the door to understanding exactly how much more data is needed to gain on par state of the art performance for a model trained on magnitudes more data[15].

Lastly, the implications of having much smaller models not only increase the energy efficiency and sustainability of AI models, but also present a unique opportunity for increasing accessibility and democratizing access to high-quality AI models for lower resource and endangered languages.

9. Appendix

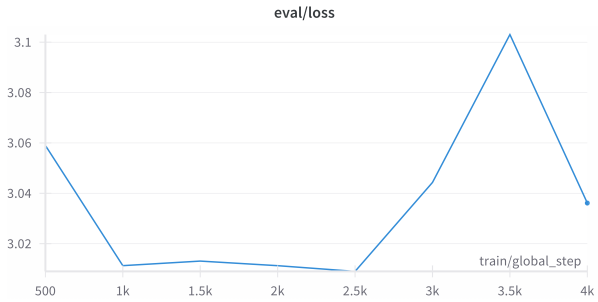

Figure 10. Evaluation Loss vs. Training Step for 10^{-2} Learning rate

[14]Overall, although we were hoping to see that the model performance of Wav2Vec2IPA could rival the baseline model, it is quite interesting to see how using such little data can still give us quite robust results. With respect to linguistic analysis of the data, it seems interesting that the missing phonemes in transcriptions tend to be voiced glides and nasals which are more complicated and varying with respect to articulatory production.

[15]An interesting result of this machine learning research is how much model performance is dependent on data quantity and quality which is why the Dataset and open source contribution to the TIMIT variants are central to this paper's evaluation.

References

Ardila, R., Branson, M., Davis, K., Henretty, M., Kohler, M., Meyer, J., Morais, R., Saunders, L., Tyers, F. M., and Weber, G. Common voice: A massively-multilingual speech corpus. *CoRR*, abs/1912.06670, 2019. URL http://arxiv.org/abs/1912.06670.

Baevski, A., Zhou, H., Mohamed, A., and Auli, M. wav2vec 2.0: A framework for self-supervised learning of speech representations. *CoRR*, abs/2006.11477, 2020. URL https://arxiv.org/abs/2006.11477.

Conneau, A., Baevski, A., Collobert, R., Mohamed, A., and Auli, M. Unsupervised cross-lingual representation learning for speech recognition. *CoRR*, abs/2006.13979, 2020. URL https://arxiv.org/abs/2006.13979.