SPEECH TO fʊnim: DEEP LEARNING AUTOMATIC PHONE RECOGNITION
MILESTONE 8: PROJECT REPORT

A PREPRINT

Kip McCharen
cam7cu@virginia.edu

Pavan Kumar Bondalapati
pb7ak@virginia.edu

Siddarth Surapaneni
sss2ea@virginia.edu

June 30, 2021

1 Introduction and Motivation

During client intake evaluation, many pediatric-focused Speech-Language Pathologists (SLPs) conduct phonological assessments to describe and analyze each child's phonological system. The clinician must assess the child's vowel production, phonemic inventory, clusters, and identify variations. Usually this process relies on the clinician manually transcribing phonemes for manual compilation and analysis. This process is time intensive and prone to human error during manual analysis, and takes time away from clinical work actively helping the child. In addition to the manual process, all humans habituate to stimulus over time, which in the context of speech therapy means that practitioners get used to clients' utterance patterns and begin to automatically infer meaning which would not be clear to others. Due to idiolect habituation, we cannot be sure that even experienced speech therapists are documenting speech objectively.

Due to the manual effort and format of recorded data, assessing long term progress for clients and across clients is prohibitively costly and not done without specific funding and time, and even then researchers are limited by the high cost of transcription and data accessibility. Due to this high cost, the fields of speech and language research are, and potentially have always been, suffering from a power-based replication crisis. In a 2020 statistical survey of 1,387 SLP and audiology peer-reviewed journal articles, Gaeta and Brydges found that "most study designs were not sufficiently powered to reliably detect a true effect" and furthermore that "most study designs in the field are underpowered and have not improved, on average, in the past 45 years" [11]. These findings are a staggering indictment of current SLP research, education, and practices, and match anecdotal experience of professional SLPs both in their education and in the field. A tool which could reliably and objectively transcribe phonemes across a variety of settings, dialects, and vocal ranges would transform the ability of both clinicians and researchers to collect, aggregate, and analyze speech both within and across studies. For all these reasons, clinicians and researchers would welcome automated generalizable tools to transcribe and analyze phonemes from spontaneous vocal sample recordings.

Clinicians have previously used the products Systemic Analysis of Language Transcripts (SALT) and its successor, Sampling Utterances and Grammatical Analysis Revised (SUGAR), as off-the-shelf tools to assist with phonological assessments. However, these tools do not actually complete any automated transcription, they are merely IDEs that assist practitioners in manually transcribing sessions. In that way there are no competitors to a successful phoneme transcription tool, and its value is immeasurable for practical daily and continuous implementation.

2 Data Collection

We are using the TIMIT corpus, which is a well established standard corpus against which new techniques are frequently tested for universal comparison. In 1986, Texas Instruments (TI) and the Massachusetts Institute of Technology (MIT) jointly developed a corpus of 630 speakers of eight major dialects of American English, each reading ten phonetically rich sentences [1]. The wikipedia page on TIMIT compares accuracy rates across different techniques of phoneme recognition, the leader of which was discussed in our last paper as controversial since Cao and Fan included only a small subset of phonemes in their analysis. The TIMIT dataset consists of three files for each sentence spoken by each person in the dataset: a space-separated file enumerating each word spoken with the word's start time and end time in milliseconds, a space-separate file enumerating each phoneme (written in a unique TIMIT phonetic representation, not the international phonetic alphabet) with the phoneme's start time and end time, and a .WAV audio file recording of the speaker's utterance.

3 Hypothesis

We hypothesize that using transfer learning in combination with self-attention over phoneme sequences will result in Phone Error Rates (PER) approaching state-of-the-art 8.3% using only TIMIT data. Error rates this low present the opportunity for a phoneme transcription tool which is accurate enough to use for practical daily transcription and analysis of fluid natural speech.

4 Literature Review

Significant work has been done on automatic speech recognition (ASR) techniques, notably including fairly successful implementations such as Siri and Alexa. However, ASR is a different task than automatic phone recognition (APR), which involves consistently identifying not words but the unique and irreducible sounds from which words may be formed. There are many reasons to detect phonemes, including transcriptions of unheard or poorly documented languages like Inuktitut [12], tracking children's exposure to word diversity [14], speech and voice disorder detection [6], and theoretical but untested methods like automated phonological assessment.

The BERTphone model trained only with CTC loss was able to reach a PER of as low as 11.5% on the VoxCeleb1 development set, but we do not know the PER of this construction on the TIMIT corpus [13]. Stadermann et al. combined Hidden Markov Models (HMM) with Recurrent Neural Nets (RNN) into a hybrid model for Automatic Speech Recognition (ASR) which resulted in a demonstrated a word error rate (WER) of 10.52% [2]. Prior research on APR has resulted in Phone Error Rates (PER) of as low as 8.7% using connectionist temporal classification (CTC) loss functions [8]. Baevski et al. combined multi-layer convolutional feature encoders with 24 transformer blocks to achieve a PER of 8.3% on the TIMIT dataset [10]. Our goal is to explore combining best practices from our research into a combined model in an attempt to achieve a lower PER on the benchmark TIMIT corpus dataset [1].

5 Problem Connecting

In order to apply a deep learning architecture to classify phonemes, the TIMIT dataset needs to be transformed into a TensorFlow Dataset object. First, the raw speech files in the TIMIT dataset only contain audio recordings of full sentences, instead of isolated phonemes. With segmentation of the audio by the noted intervals in the previous section, we can separate the individual phonemes from every spoken sentence in the dataset. These isolated phonemes have different lengths due to the audio segmentation. Implementing inputs of variable sizes are problematic in deep learning architectures written in TensorFlow. Given that the segmentation is performed on the time domain, the audio data is interpreted on the frequency domain. In other words, the utterance of a phoneme can be represented as an array of real numbers denoting frequency, where the indices refer to time in milliseconds (ms). To rectify the issue of variable input sizes, we can apply zero-padding to the left-hand and right-hand side of this array, so that the phonemes are all of a uniform length.

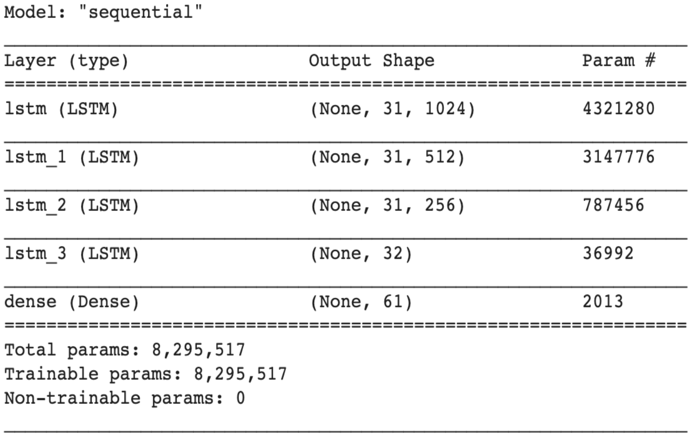
Given these transformations to the speech files in the TIMIT dataset, the spliced audio data containing the phonemes and the associated phoneme labels are written to the disk in a TFRecord format. By storing these variables in a protocol buffer, we can utilize serialization for efficient data storage. Using the TensorFlow API, we can readily load our TFRecords into a TensorFlow Dataset. From here, we can experiment with numerous deep learning architectures in order to classify phonemes.

6 Analysis and Interpretation

For this project, we initially started with a feedforward neural network consisting of 3 Dense layers with each layer followed by dropout and Batch Normalization. This simple model was used to get a sense of the performance when solely trying to learn from the isolated phonemes without converting them into Spectrograms or extracting MFCCs from them. The resulting phone error-rate(PER) for this model was 69.69%.

After establishing this baseline with the simple feedforward neural network, we encoded our audio tensors into Spectrograms utilizing a Fast Fourier Transform size of 512, a window size of 512, and a stride of 256. We then implemented a convolutional neural network model to learn from these Spectrograms as Spectrograms are a way to visualize the frequencies in an audio signal. This convolutional neural network architecture was adapted from VGG16 where convolutional layers were stacked on top of each other with these stacked convolutional layers(4 stacks, consisting of 2,3,3,3 convolutional layers respectively with each stack consisting of 64,128,256,512 filters resepectively) being followed by a max pooling layer(size of 2). We added 3 dense layers containing 128 units each on top of the convolution layers with each of these layers being followed by Batch Normalization and Dropout(0.2). A softmax output layer was used. This model improved performance with resulting Phone error rate being 31.99%.

The next approach extracted MFCCs was from the audio tensors and utilized LSTMs to take advantage of the sequential nature of the data. We utilized the first extracted first 30 MFCCs from the log-mel spectrograms giving a tensor of shape (31,30) for each audio tensor. The ideas was then to stack LSTM layers that would then create an encoding for each the given audio sequence. These stacked LSTM layers would then help create an encoding that would provide a higher level representation of the MFCC sequence data. Here is the model summary:



```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 31, 1024)          4321280
_________________________________________________________________
lstm_1 (LSTM)                (None, 31, 512)           3147776
_________________________________________________________________
lstm_2 (LSTM)                (None, 31, 256)           787456
_________________________________________________________________
lstm_3 (LSTM)                (None, 32)                36992
_________________________________________________________________
dense (Dense)                (None, 61)                2013
=================================================================
Total params: 8,295,517
Trainable params: 8,295,517
Non-trainable params: 0
_________________________________________________________________
```

The first three layers of the model created an encoding of the sequence. The model had a phone error rate of 32.99% PER.

Given that the LSTM model had similar performance as the convolutional neural network model, Bidirectional LSTMs were leveraged to develop a better understanding of the audio signal. The implemented model adapted an approach by Li et. al, where a 6 layer stacked bidirectional LSTM with 1024 units each [12] was used to encode the audio signal. Similar to their approach, the first 40 MFCCs were extracted from the audio tensors.

The initial models were not able to get a lower PER than 31.99%. These models seemed to be struggling with learning enough information from the various types of inputs we passed in, whether it was spectrograms or MFCCs. Due to the dense high-dimensional information within human voice audio signal,the models we implemented were not able to adequately learn from the segmented phoneme data. Another issue was the sequence dependent nature of phonemes. Considering that phonemes are dependent on the sounds that come before and the sounds that come back after, using isolated phone sounds wasn't an adequate approach to detecting sounds that are sequence dependent. Since phonemes can sound different from individual to individual, there is a lot of a uncertainty associated with phoneme detection as which phoneme is correct can vary from one situation to another.

In trying to deal with these issues, we discovered the SpeechBrain toolset, an open-source all-in-one speech deep learning object-oriented toolkit built on top of the PyTorch package [17]. This toolset allows users to design recipes involving state-of-the art model architectures that can then be run on the CLI. SpeechBrain allows users to define a yaml file enumerating every parameter required for a given model with yaml-native meta referencing. Leveraging the SpeechBrain recipes, we were able to utilize transfer learning and train a seq2sec transformer model utilizing the embeddings from the Wav2vec2.0 model on our TIMIT dataset. This model utilizes pretrained embeddings, attention mechanisms, and takes a probabilistic approaches to identifying the phonemes in a given audio sequence. This model was able to achieve a PER of 8.69%.

The full architecture of our model which achieved 8.69% PER involved:

- **wav2vec2**: wav signal to wav2vec2 dimension reduction embedding of audio samples
- **augmentation**: only training wav embeddings undergo augmentation to improve generalizability, including probabilistic:
  - dropping of chunks of the audio (replaced with zero amplitude or white noise)

  - dropping of frequency bands (with band-drop filters)
  - speed perturbation (via resampling to slightly different rate)
- **enc**: wav embeddings to "Vanilla" neural net encoding with 2 layers, 1024 input and output neurons, and LeakyReLU activation (as shown above)
- **ctc_lin**: encodings to CTC linear probabilities passed through a log-softmax function
- **emb**: list sequence of encoded phones with beginning-of-sentence marker to neural net 128-node weight embedding with 42 outputs
- **dec**: sequence weight embedding and encodings to location-attentional GRU decoder layer with 256 attention dimensions, one hidden layer of 256 neurons, and 50% dropout
- **seq_lin**: GRU decodings to sequence-to-sequence output layer that is then passed through a log-softmax function to obtain output probabilities for the phonemes in the sequence
- **search**: for validation steps, a speedy heuristic Greedy Search algorithm is used to select the best prediction among probabilities from the seq_lin step. In test evaluation, a more complicated Beam Search algorithm with a beam size of 16, that selects the 16 words with the highest probability for each phoneme in the sequence and utilizes conditional probabilities to find the 16 most probable sequences at each point of the sequence. Then at the end of the sequence, the sequence with the highest probability was used as the prediction.

Loss is calculated within this structure with a mixed-model combination of 20% weighted CTC loss and 80% Seq2seq loss.

Mo̧del	**PER**
Feedforward Network	69.69
CNN	31.99
LSTM	32.99
Bidirectional LSTM	33.67
Seq2seq w/ Wav2Vec	8.62

Table 1: Performance of the Different Model Approaches.

7 Discussion

This study's PER of 8.6% is below the 2019 CTC-Acoustic Model benchmark of 8.7% and successfully approaches the current gold standard of 8.3% achieved by Baevski et al. in 2020 using a similar Wav2Vec-based transfer-transformer model [10]. However, the current model requires inputs to include time stamps for each phoneme which must be predicted, but a complete APR tool requires end-to-end transcription, including a phoneme edge detector.

This study trained and evaluated our model exclusively on the TIMIT dataset. However, there are additional datasets which could be used to further train and evaluate the model with more diverse utterances and speakers. Li et al. hypothesized in 2020 that a more diverse and deep collection of multilingual phonemes, such as those transcribed and aligned in the PHOIBLE [4] database, should allow an APR model to better differentiate between subtle phoneme differences. However the Allosaurus model lacked newly available transfer-transformer techniques. Diverse multilingual phonemic datasets could ideally be concatenated together into a superset with consistent inputs and outputs, potentially including the Buckeye dataset [3], Tusom2021 [16], Librispeech [5], and Persian Consonant-Vowel Combination (PCVC) dataset [7].

Unfortunately the ARPABET phoneme texts used in TIMIT are not useful to an SLP clinician, who instead uses the International Phonetic Alphabet (IPA). Aligning phoneme transcription texts is a complicated procedure which requires linguistics expertise, such as demonstrated and provided in the PHOIBLE 2.0 dataset [9]. Consistent alignment across a diverse set of phoneme datasets as above would be a daunting but theoretically possible task.

Even if a tool is able to accurately transcribe phonemes directly from recordings, an SLP practitioner must also compare produced utterances against target utterances. Presumably a child who can accurately produce all phonemes on demand has no reason to visit an SLP, a new client must be compared against what they *meant* to say. Next generation tools may be able to use extant transcriptions and identify the most probable intended utterance, against which pronunciation errors can be inferred and documented. Until such a tool is designed, a useful tool would need a human-in-the-loop interface to allow practitioners to correct utterances such as "buddah" to potentially "puddle" or "butter", and thus identify either trouble pronouncing an "r" and a "t", or trouble pronouncing a "p" and an "l".

Finally a useful tool must aggregate and store phonetic utterance recordings, transcriptions, and annotations in an easily accessible and PII-safe manner. Data engineering techniques must account for on-demand streaming, security protections, and possibly cloud-based aggregate analysis techniques.

8 Conclusion

The product of this study was an exciting next step towards an end-to-end audio to phoneme transcription tool which has the potential to revolutionize speech, language, and audiology research and practice. We have achieved the desired goal of this study to approach state-of-the-art PER on the TIMIT dataset. Significant work remains to succeed at developing a complete Automatic Phone Recognition (APR) tool for practical use, but this study demonstrates that it is time to confidently approach the next steps and develop a proof of concept APR tool.

Once a successful APR tool is accomplished, the next goal is to refine this tool to consistently identify speech and voice disorders which are currently difficult for experts to differentiate with trained ears. Ongoing research is helping identify more disorders such as Parkinsons's Disease purely from smartphone voice audio [15], which suggests deeper potential applications in other medical contexts where phonological production is affected, even in small ways imperceptible to the casual human ear. Machine learning has unlocked many secrets of sound in our world and there are many more secrets to find.

9 Team Contribution

The team worked well together and contributed equally to writing up sections of the reports across the project. Each week the team had a set of tasks to work on and these task were equally distributed across the team members. The team met at least twice each week to share progress with each other and collectively decide the next steps based on the progress made.

References

[1] *TIMIT: Acoustic-phonetic continuous speech corpus.* English, OCLC: 53222255, Philadelphia, Pa., 1993.

[2] J. Stadermann and G. Rigoll, "Comparing NN paradigms in hybrid NN/HMM speech recognition using tied posteriors," in *2003 IEEE Workshop on Automatic Speech Recognition and Understanding (IEEE Cat. No.03EX721)*, Nov. 2003, pp. 89–93. DOI: 10.1109/ASRU.2003.1318409.

[3] M. A. Pitt, K. Johnson, E. Hume, S. Kiesling, and W. Raymond, "The Buckeye corpus of conversational speech: Labeling conventions and a test of transcriber reliability," en, *Speech Communication*, vol. 45, no. 1, pp. 89–95, Jan. 2005, ISSN: 01676393. DOI: 10.1016/j.specom.2004.09.001. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0167639304000974 (visited on 02/22/2021).

[4] S. P. Moran, "Phonetics Information Base and Lexicon (PHOIBLE)," English, PhD thesis, University of Washington, 2012. [Online]. Available: http://hdl.handle.net/1773/22452 (visited on 03/07/2021).

[5] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210. DOI: 10.1109/ICASSP.2015.7178964.

[6] T. B. Ijitona, J. J. Soraghan, A. Lowit, G. Di-Caterina, and H. Yue, "Automatic detection of speech disorder in dysarthria using extended speech feature extraction and neural networks classification," in *IET 3rd International Conference on Intelligent Signal Processing (ISP 2017)*, Journal Abbreviation: IET 3rd International Conference on Intelligent Signal Processing (ISP 2017), Dec. 2017, pp. 1–6. DOI: 10.1049/cp.2017.0360.

[7] S. Malekzadeh, M. H. Gholizadeh, and S. N. Razavi, "Persian Vowel recognition with MFCC and ANN on PCVC speech dataset," *arXiv:1812.06953 [cs, eess]*, 2018, arXiv: 1812.06953. DOI: 10.13140/RG.2.2.12187.72486. [Online]. Available: http://arxiv.org/abs/1812.06953 (visited on 05/12/2021).

[8] D. He, X. Yang, B. P. Lim, Y. Liang, M. Hasegawa-Johnson, and D. Chen, "When CTC Training Meets Acoustic Landmarks," *arXiv:1811.02063 [cs, eess]*, Feb. 2019, arXiv: 1811.02063. [Online]. Available: http://arxiv.org/abs/1811.02063 (visited on 03/24/2021).

[9] S. Moran and D. McCloy, Eds., *PHOIBLE 2.0*. Jena: Max Planck Institute for the Science of Human History, 2019. [Online]. Available: https://phoible.org/.

[10] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," *arXiv:2006.11477 [cs, eess]*, Oct. 2020, arXiv: 2006.11477. [Online]. Available: http://arxiv.org/abs/2006.11477 (visited on 03/19/2021).

[11] L. Gaeta and C. R. Brydges, "An Examination of Effect Sizes and Statistical Power in Speech, Language, and Hearing Research," en, *Journal of Speech, Language, and Hearing Research*, vol. 63, no. 5, pp. 1572–1580, May 2020, ISSN: 1092-4388, 1558-9102. DOI: 10.1044/2020_JSLHR-19-00299. [Online]. Available: http://pubs.asha.org/doi/10.1044/2020_JSLHR-19-00299 (visited on 05/12/2021).

[12] X. Li, S. Dalmia, J. Li, M. Lee, P. Littell, J. Yao, A. Anastasopoulos, D. R. Mortensen, G. Neubig, A. W. Black, and F. Metze, "Universal Phone Recognition with a Multilingual Allophone System," *arXiv:2002.11800 [cs, eess]*, Feb. 2020, arXiv: 2002.11800. [Online]. Available: http://arxiv.org/abs/2002.11800 (visited on 03/07/2021).

[13] S. Ling, J. Salazar, Y. Liu, and K. Kirchhoff, "BERTphone: Phonetically-aware Encoder Representations for Utterance-level Speaker and Language Recognition," en, in *Odyssey 2020 The Speaker and Language Recognition Workshop*, ISCA, Nov. 2020, pp. 9–16. DOI: 10.21437/Odyssey.2020-2. [Online]. Available: http://www.isca-speech.org/archive/Odyssey_2020/abstracts/93.html (visited on 03/19/2021).

[14] O. Räsänen, S. Seshadri, M. Lavechin, A. Cristia, and M. Casillas, "ALICE: An open-source tool for automatic measurement of phoneme, syllable, and word counts from child-centered daylong recordings," en, *Behavior Research Methods*, Sep. 2020, ISSN: 1554-3528. DOI: 10.3758/s13428-020-01460-x. [Online]. Available: http://link.springer.com/10.3758/s13428-020-01460-x (visited on 03/08/2021).

[15] S. Singh and W. Xu, "Robust Detection of Parkinson's Disease Using Harvested Smartphone Voice Data: A Telemedicine Approach," en, *Telemedicine and e-Health*, vol. 26, no. 3, pp. 327–334, Mar. 2020, ISSN: 1530-5627, 1556-3669. DOI: 10.1089/tmj.2018.0271. [Online]. Available: https://www.liebertpub.com/doi/10.1089/tmj.2018.0271 (visited on 05/12/2021).

[16] D. R. Mortensen, J. Picone, X. Li, and K. Siminyu, "Tusom2021: A Phonetically Transcribed Speech Dataset from an Endangered Language for Universal Phone Recognition Experiments," *arXiv:2104.00824 [cs, eess]*, Apr. 2021, arXiv: 2104.00824. [Online]. Available: http://arxiv.org/abs/2104.00824 (visited on 05/12/2021).

[17] M. Ravanelli, T. Parcollet, A. Rouhe, P. Plantinga, E. Rastorgueva, L. Lugosch, N. Dawalatabad, C. Ju-Chieh, A. Heba, F. Grondin, W. Aris, C.-F. Liao, S. Cornell, S.-L. Yeh, H. Na, Y. Gao, S.-W. Fu, C. Subakan, R. De Mori, and Y. Bengio, *Speechbrain*, https://github.com/speechbrain/speechbrain, 2021.