Discourse on ASR Measurement: Introducing the ARPOCA Assessment Tool

Megan Merz
Rose-Hulman Institute of Technology
merzm@rose-hulman.edu

Olga Scrivner
Rose-Hulman Institute of Technology
scrivner@rose-hulman.edu

Abstract

Automatic speech recognition (ASR) has evolved from a pipeline architecture with pronunciation dictionaries, phonetic features and language models to the end-to-end systems performing a direct translation from a raw waveform into a word sequence. With the increase in accuracy and the availability of pre-trained models, the ASR systems are now omnipresent in our daily applications. On the other hand, the models' interpretability and their computational cost have become more challenging, particularly when dealing with less-common languages or identifying regional variations of speakers. This research proposal will follow a four-stage process: 1) Proving an overview of acoustic features and feature extraction algorithms; 2) Exploring current ASR models, tools, and performance assessment techniques; 3) Aligning features with interpretable phonetic transcripts; and 4) Designing a prototype ARPOCA to increase awareness of regional language variation and improve models feedback by developing a semi-automatic acoustic features extraction using PRAAT in conjunction with phonetic transcription.

1 Introduction

Automated speech recognition (ASR) is the process of automatically detecting and recognizing the words that have been said in a sample of speech. ASR has a wide variety of uses, such as voice assistants, automatic transcription, speech-to-text, and closed-caption generation. Many recent ASR models have been created using deep learning, with other methods including neural networks, hidden Markov models, and Gaussian mixture models (Papastratis, 2021).

ASR models are generally trained on a corpus, which is a collection of audio recordings. Corpora are widely available for more common languages, such as English. However, they are either small or nonexistent for less common languages and dialects. This is due to the resources needed to construct a corpus and lack of available speakers. Constructing a corpus involves gathering audio recordings from a variety of speakers and is a time-consuming and costly process. As a result, less common languages remain under-resourced in the ASR field. The performance accuracy will also vary with regional language variation and among different groups of users. ASR performs especially poorly when given the task of recognizing the speech of nonnative speakers of a language, leading to model biases in common AI-assisted speech technologies (DefinedCrowd, 2021).

Furthermore, there is a lot of variation in ASR systems. In the last decade, the ASR technology has evolved from probabilistic frameworks with hand-crafted features and pronunciation dictionaries to deep learning models in which features are extracted and learned in hidden layers (Georgescu et al., 2021). Speech signals also consist of various components, such as acoustic, phonetic, and language-dependent, which jointly provide a representation of word sequences. While some features are interpretable by humans (e.g., place of articulation, vowel formants, pitch), others are the results of transformations and cannot be directly associated with any specific phonetic sound.

Finally, various evaluation systems are put forth to measure speech model accuracy (Negri et al., 2014). Grapheme-based metrics (a written word) are commonly used to compare results, such as word error rate (WER). These measurement systems, however, are not able to diagnose whether phonetic errors resulted from a variation in pronunciation, speech boundary misalignment, noise, or the lack of sufficient data.

This research is focused on existing ASR evaluation systems and speech signal features used for training. We explore solutions for improving measuring performance metrics. Our goal is to 1) develop a semi-automatic phonetic classification be-

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Student Research Workshop, pages 366 - 372

tween vowels and consonants as these classes are traditionally associated with different salient features (e.g., vowel formants, consonant intensity, aspiration), 2) help ASR developers to identify improvement areas by focusing on specific feature engineering tasks, and 3) design an alternative evaluation system to encourage the ASR research for less-commonly used languages by incorporating development cost, corpus size, and phonetic transcript as compared to a traditional word error rate evaluation metric.

The paper is organized as follows. Section 2 presents the overview of ASR performance evaluation metrics, current ASR models and corpora. Section 3 describes the most common types of speech features and tools for their generation. In Section 4, we present our proposed evaluation system ARPOCA (Assessment of ASR using phonemes, originality, cost, and accent performance). Finally, we provide our preliminary results in Section 5, followed by conclusion and future direction.

2 Literature Review

2.1 Measuring ASR Performance

One common way of measuring the performance of automatic speech recognition (ASR) models is word error rate (WER). WER is a way to measure the accuracy of ASR. The best possible value is 0% error, and higher percentages are considered worse. WER is counted by letting a model transcribe a section of audio, then comparing it to the correct transcription. Both transcriptions are normalized before comparing, which standardizes the transcripts by removing stop words, forming contractions, etc. The words that the model has inserted, deleted, or substituted are counted and used to calculate WER using the formula illustrated in Eq.1, where S is a word substitution, D is a deletion, and I is a word insertion:

$$WER = \frac{(S + D + I)}{TotalWords} \quad (1)$$

WER is a commonly used method to assess the performance of ASR models, and creating a model with a low WER is assumed to result in a model with better language understanding accuracy. However, a better WER may not actually result in a model with a better understanding of spoken language, meaning that even if a transcript is mostly accurate, it may not correctly represent the meaning of the spoken language (Wang et al., 2003).

This problem of accuracy is especially pertinent for models that are trained with small corpora, since these models often have a poor WER. The early study comparing different spoken language models (Wang et al., 2003) found that, while the Model developed using Hidden Markov and Context Free Grammar (HMM/CFG) had a worse WER than other language models (e.g. a trigram model) it achieved a better task classification error rate, which is a way to measure how well the model understands the spoken language. This result was even more pronounced for models trained with small amounts of data: the HMM/CFG model was able to use less training data and still generate a model with a better level of understanding than the trigram model. It is worth noting that the HMM/CFG model used domain knowledge and a grammar library, which helped it achieve good results without a large training dataset (Wang et al., 2003). So, while WER can be used as a way to measure performance, other metrics (e.g., task classification error rate) may be more useful, especially for models trained with smaller corpora.

In addition, WER does not provide much feedback for developers. While it measures the number of mistakes a model made, it does not help in revealing why the mistakes were made or whether similar mistakes were made repeatedly. Providing more feedback could aid developers in diagnosing problems with their models more quickly and in the end, creating better models. This project discusses the possibility of providing more feedback for ASR models by identifying commonly mistaken sounds and recognizing different pronunciations for words.

Another metric for the accuracy of ASR is phoneme error rate (PER), which is calculated similarly to WER. However, while WER is at the word level, PER counts the number of deleted, inserted, and substituted phonemes. Phonemes are smaller than words, which could potentially help pinpoint errors better.

2.2 Methods for ASR

Deep learning is commonly used for ASR. There are typically four steps in ASR: 1) pre-processing, 2) feature extraction, 3) classification, and 4) language modeling. Pre-processing is a process applied to recordings which reduces noise and filters the audio. Feature extraction converts the audio to features, which are then analyzed and converted to language in the classification step. Mel-frequency

Cepstral coefficients (MFCC) is commonly used for the feature extraction step. MFCC converts audio signals into a linear model of human auditory processing, which is non-linear.

Deep neural networks can be used for ASR, such as recurrent neural networks (RNN), convolutional neural networks, and transformer networks. One limitation of RNNs is that they process speech using only the previous input. However, speech depends on both what comes before and what comes after. This problem can be solved using bi-directional RNNs, which process speech forward and backward. Furthermore, Connectionist temporal classification (CTC), can be used to find the most probable alignment, which is the arrangement of speech and silence. Silence can be either not speaking or transitioning between words or sounds. CTC must be used in combination with a decoding step, such as the best-path decoding algorithm. The best-path decoding algorithm aims to find the most likely word for each sequence of sound. A method called RNN-transducer uses an RNN with CTC to analyze input and also a separate RNN to predict likely words in the sequence based on previous words (Papastratis, 2021).

Dialect detection uses similar methods as ASR, so dialect detection could be used to help improve ASR. There are several motivations for dialect identification, including determining the regional origin and ethnicity of a speaker in order to adapt content (Ismail, 2020). For example, deep neural networks have been used to distinguish between dialects of Arabic. A recent study by Lulu and Elnagar (2018) used an existing dialectal dataset called the AOC (Arabic Online Commentary), which has about 110 thousand labeled sentences. The motivation for the study was to improve dialect detection for Arabic as informal dialects of Arabic are widely used on the internet, especially for applications such as blogs, forums, social media, and more. The study showed that dialect detection is also useful for machine translation and sentiment analysis. Four different types of deep neural network were used: long-short term memory (LSTM), convolutional neural networks (CNN), bi-directional LSTM (BLSTM), and convolutional LSTM (CLSTM). Three different dialects were examined - Egyptian, Gulf (which included the similar Iraqi dialect) and Levantine. Of the neural networks, the LSTM was the most accurate overall, with approximately 80% accuracy on average, which is below the human accuracy of about 90% (Lulu and Elnagar, 2018).

2.3 Data for ASR

There is a large amount of variability in the corpora used for ASR. Often, corpora are built at the word or phrase level. However, for some languages, such as Tibetan, a corpus at the syllable level can work better due to the lack of accuracy for word and phrase recognition (Dao et al., 2021). Many corpora use speech samples that have been recorded with minimal environmental noise and are of good quality, which results in models that work best in these ideal conditions. However, real life conditions can result in noisier speech, so models that have not been trained with noisy speech can struggle under such conditions (Borský, 2016).

Corpus creation can be a difficult and expensive process, which often results in smaller or nonexistent corpora for less spoken and under-resourced languages. Even if corpora exist for a language, they may not be suitable for certain applications, as was the case for an experiment conducted by Zissman et al. (1996). They found that while Spanish corpora existed, there was no corpus that had enough speakers of a variety of dialects. This led to the creation of the Miami corpus, which collected speech from Spanish speakers from Peru, Cuba, and other countries (Zissman et al., 1996). There are a number of steps involved in corpus creation. First, recordings must be obtained. This means researchers either have to find people to record their speech or find existing recordings. There are a variety of sources for existing recordings, such as audio books or YouTube videos (Ismail, 2020). If a transcript does not exist for the recording, then one must be created. Then, the transcript and audio must be aligned to ensure that the words shown in the transcript are placed where the same words are spoken in the recording (Panayotov et al., 2015). Recordings may also be cleaned of background noise and normalized. While there have been efforts to automate the corpus creation process, it is not guaranteed to be accurate. Therefore, much of this process is done manually.

3 Speech Signal Features

Feature extractions is a pre-processing task which transforms sound files into feature vectors that can be processed and analyzed by a computer. This tasks can be classified into two main groups: segment and suprasegmental prosodic features versus

speaker-dependent and speaker independent features (Georgescu et al., 2021; Shahnawazuddin et al., 2020). While most of acoustic phonetics utilize interpretable features (e.g. vowel formant, duration, voice onset time) to describe phonemes (mental representation of sound) and phones (actual sounds), the ASR field relies on transformed feature vectors optimized for Machine learning tasks (e.g. Linear Prediction and Mel-Frequency coefficients).

3.1 Acoustic Features

Formant is a common interpretable measurement that correspond to resonance frequencies in a vocal tract. The first formant (F1) is correlated with high-low dimension and inversely related to vowel height, where high values represent open vowels (e.g. /a/) as compared to low values for low vowels (e.g. /i/). The second formant (F2) is correlated with front-back dimension, namely the degree of backness for a vowel. For example, front vowels (e.g., /i/) will have higher F2 values than back vowels (e.g., /o/). The third formant (F3) indicates the round shape of a vowel (Ladefoged, 2006; Kent and Vorperian, 2018). These values can be seen in a spectogram as dark bands. It should be noted that these values are not uniform across speakers, speech style, morphological context, and language variation, as can be seen from Spanish acoustic data illustrated in Fig.1, where solid line represents a vowel space obtained in a controlled laboratory sampling of Peninsular Spanish and dotted lines demonstrate a much smaller vowel space from a spontaneous speech of Venezuelan Spanish (Scrivner, 2014).

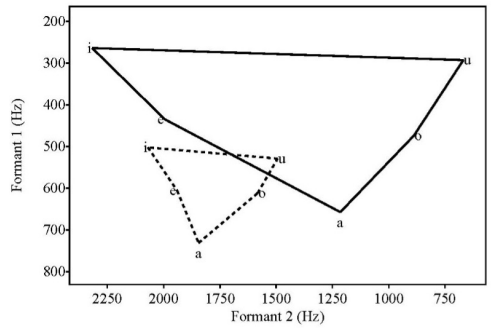

Figure 1: Comparison of Spanish vowel formants between controlled (solid line) and spontaneous speech (dotted line) and between two Spanish dialects (Venezuelan and Peninsular).

Similarly, consonants have three dimensions but related to 1) place of articulation (e.g. dental, glottal), 2) manner of articulation (e.g., nasal, fricative), and 3) voicing (Ladefoged, 2006).

In sum, three classes of distinct sound landmarks have been proposed: 1) abrupt discontinuity of consonants, 2) steady periods of vowels, 3) non-abrupt transition of glides (e.g. /w/) (Park, 2008).

3.2 Feature Vectors Extraction Algorithms

One of the preliminary operations to generate vector features is framing. Framing breaks the sound into small frames, typically 25ms long with 10ms overlap with neighboring frames. The overlap is important due to the dependence which speech has on preceding and following sounds. During framing, windowing is carried out, in which a Hamming or Han (sometimes referred to as Hanning) filter is performed. The window function decreases the amplitude at the beginning and end of the frame, which again, makes overlapping frames necessary to prevent anomalies (Georgescu et al., 2021).

Several feature extraction methods can be applied after framing, namely, Fast Fourier Transform (FFT), Linear Prediction Coefficients (LPC), Mel-Frequency Ceptrum Coefficients (MFCC), Mel-Filterbanks, Discrete Cosine Transform (DCT). FFT is a common technique used to transform speech signal from a time domain to a frequency domain. The FFT separates the air exhaled from the lungs and the time response of the vocal tract by converting from the time domain to the frequency domain, which allows these two features to be separated. When framing, windowing, and FFT are applied to an audio sample, a spectrogram can be created from the results. In contrast, LPC relies on linear prediction. It uses past samples to predict the current sample. However, this method has some drawbacks, such as inability to distinguish similar vowel sounds and its inaccurate analysis of speech signals due to the assumption that speech signals are stationary. Finally, MFCCs can be obtained by applying DCT to the log power spectrum of mel frequencies (Gupta et al., 2018).

4 ARPOCA Approach

In response to the problems previously identified in the field of speech recognition, this proposal aims to develop a more in-depth evaluation system called ARPOCA. ARPOCA is an acronym for Assessment of ASR using Phonemes, Originality, Cost, and Accent performance. The main goal is

to develop a phoneme recognition system using phoneme classification and transcription, independent from a grapheme representation used in WER.

First, we selected open source existing tool Praat, a software designed for sound processing (Boersma, 2001; Styler, 2011), to extract interpretable feature representations for each phoneme. Second, we identified the following salient features for phoneme classification: frequency formants, dispersion (also called standard deviation), center of gravity, and intensity. Standard formant ranges for F1, F2, F3 are used to identify vowels. Dispersion, center of gravity, and intensity are used to identify consonants. Center of gravity measures at what frequency a sound is most concentrated, while dispersion measures how widely the frequencies of a sound are spread. Intensity measures the loudness of a sound in decibels. For testing, we obtained a non-transcribed free sample Spanish audio corpus (Defined.ai, n.d.).

In our next stage, we will create a manual phonetic transcription of utterances from the corpus, in addition to segmenting and labeling the utterances for usage in PRAAT. We will collect information about expected values of acoustic features used for identifying phonemes and compare our manual phonetic transcription with the output from an available speech recognizer library in python. In addition, we will analyze several existing models to establish a baseline for originality and cost in these models, and use this to create a rating system. Furthermore, the phoneme recognition system will incorporate an accent performance analysis. That is, the phoneme recognition system will identify whether a model has a wide pronunciation gap and identify particular areas where a model struggles, which will help close the accent gap.

5 Preliminary Results

In the first stage of this proposal, we are exploring features extracted from spectogram and speechwave. Fig. 2 displays an example of Spanish word ‘necesito‘ (I need). The sound waves help distinguish between sound and silence, amplitude and intensity of sounds, while the spectrogram provides a view of formant frequencies, consonants obstruction and frication.

While PRAAT includes scripting, using Python in addition makes running the PRAAT script easier to automate, especially for large amounts of audio samples. Python code calls a PRAAT script, then

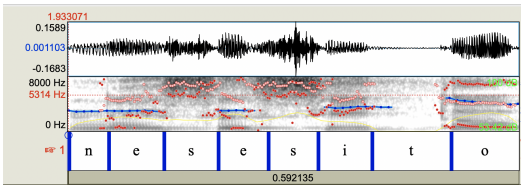

Figure 2: An example of using Praat to segment speech and label phonemes. Sound waves are shown at the top, followed by a spectrogram, then the segmentation. Red dots in the spectrogram show formants, while the blue line shows pitch.

performs additional operations on the results of the PRAAT script, such as matching the formants to the correct phoneme class. Table 1 demonstrates preliminary results from using PRAAT in conjunction with Python. The PRAAT script obtained formant values at the median time of each segment. Then, the results are matched to the phoneme based on formant range. For example, the /e/ phoneme typically has F1 values between 485 and 565 and F2 values between 2170 and 2430. While some of the vowels fell within the expected ranges of formants during testing, others did not. This could be for several reasons. One such reason is not normalizing speech prior to attempting to recognize phonemes. Normalization could help reduce variance between individual speakers. Dialectal variation may also be a factor, since vowel frequencies can vary between different dialects.

Phoneme	Duration	F1	F2	F3
n	0.0522	-	-	-
e	0.0726	572	2438	2960
s	0.0713	-	-	-
e	0.0657	484	2086	2964
s	0.0654	-	-	-
i	0.0708	489	2575	3439
t	0.1040	-	-	-
o	0.0932	6034	1274	2862

Table 1: Preliminary results from formant analysis using Praat and Python to identify formants in the audio segment ’necesito’ (I neeed). F1, F2, and F3 are values for formants 1, 2, and 3 respectively.

Since consonants cannot be identified using formants only, we use different measurements, including center of gravity, intensity, and standard deviation. Currently, using these measurements is only precise enough to differentiate between fricative and non-fricative consonants. More work must be done to refine the expected ranges for consonants

to be able to identify individual consonants.

Originality is another aspect of ARPOCA. We have determined that a scoring rubric would likely be best to assess originality, since there is little research on this topic. Thus, research that is an addition or improvement on an existing model will receive a lower score than more novel research. Cost is also an important element of ARPOCA. Preliminary research suggests that a budget of $500,000 would be attainable for many researchers (NIH, n.d.). An overall cost, including corpus cost and compute cost, which does not exceed $500,000 would score the highest, with score decreasing as total cost increases. The reason for this is twofold. Firstly, there are many applications that require smaller, less costly models. For instance, such models could be used to assist people with hearing loss by providing real-time transcriptions. Secondly, many costly models with large corpora already exist and are prioritized under the prevalent measurement system of WER. Therefore, in order to encourage innovation in the field of ASR, smaller, less costly models will be encouraged.

There are several important outcomes from the preliminary results. In its current state, the phoneme recognizer is unlikely to work with English, due to the presence of a large number of vowels which are not easily distinguishable. The phoneme identifier has been tested using Spanish, which is better suited to this purpose due to the smaller number of vowels, which are relatively easy to distinguish. An additional flaw in the phoneme identifier is its difficulty distinguishing between vowels and voiced consonants. Table 2 shows that the /n/ phoneme is identified as a vowel, but should be identified as a voiced consonant. The speech segments used were relatively noiseless; the phoneme recognizer is likely to be less accurate in a more noisy environment.

Time	Phoneme ID	SR
1.935	vowel	n
1.987	e	e
2.059	voiceless fricative	s
2.129	e	e
2.184	voiceless fricative	s
2.275	e	i
2.331	voiceless non-fricative	t
2.441	o	o

Table 2: A comparison of preliminary results from the phoneme identifier and a transcript created by the speech recognizer. Phoneme ID represents the results from the phoneme identifier, while SR represents the results from the python speech recognition.

6 Conclusion and Future Work

The objective of this work is to supplement ASR models and developers with an additional tool providing not only a feedback but also more interpretable representation of sound models via phonetic transcription. Such feedback could include highlighting phonemes that have been consistently misidentified and/or measuring performance of the model when given audio samples produced by non-native speakers, which is an area in which ASR models typically struggle. This feedback could improve the accuracy of ASR models and lessen the accent gap. Accuracy of models could also be improved by providing developers more feedback on their models than just using standard performance metrics. For instance, commonly mistaken sounds (phonemes) could be used as a form of feedback to help improve models and augment existing corpora. Furthermore, a phonetic approach could help create dictionaries with dialectal variation (regional alternative pronunciation) that can be added to training corpora. Finally, language transfer (using the resources from one language to develop resources in another similar language or dialect) could help provide resources for underrepresented spoken languages.

ARPOCA needs more development in order to become more accurate. This could include additional data for improving the cost baseline and grading in addition to more research into expected values of formants, center of gravity, intensity, and dispersion. In its current state of research, ARPOCA serves as a proof of concept for the development of a more robust assessment tool for ASR models. We envision ARPOCA being used in settings such as peer reviews and conferences to promote discussion and improvement of ASR models. ARPOCA can aid in supporting different research goals than WER. For instance, a model with a smaller corpus typically costs less to produce and would therefore score better in the cost section of WER. This could encourage the production of models for under-resourced and less widely spoken languages, even if such models do not immediately have a good enough WER score to compete with models for languages such as English. Another possible benefit of using ARPOCA is closing the

accent gap. Although the accent performance analysis system has not been developed yet, the existing phoneme identification could help developers determine if there are specific groups of formants that a model has misidentified. On the other hand, ARPOCA must be carefully revised to ensure that the scoring system is fair and accurate. If there are inaccuracies in ARPOCA or top scores are unattainable, this could result in a variety of unwanted outcomes, such as giving models the wrong scores or discouraging developers. In addition, while ARPOCA has been developed with collaboration and discussion in mind, it has the possibility to fuel competition as well, due to its role as a tool for assessment. Therefore, ARPOCA must be used with care and consideration as to whether its use is appropriate for a given situation.

Acknowledgements

We would like to thank Dr. Michael Wollowski and anonymous reviewers for their valuable feedback.

References

Paul Boersma. 2001. Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10).

Michal Borský. 2016. *Robust recognition of strongly distorted speech*. Ph.D. thesis.

Jizhaxi Dao, Zhijie Cai, Rangzhuoma Cai, Maocuo San, and Mabao Ban. 2021. A method of constructing syllable level Tibetan text classification corpus. *MATEC Web of Conferences*, 336:06013.

Defined.ai. n.d. Inclusive Speech Recognition Technology.

DefinedCrowd. 2021. Preventing Bias in Speech Technologies. Technical Report September.

Alexandru Lucian Georgescu, Alessandro Pappalardo, Horia Cucu, and Michaela Blott. 2021. Performance vs. hardware requirements in state-of-the-art automatic speech recognition. *Eurasip Journal on Audio, Speech, and Music Processing*, 2021(1):1–30.

Divya Gupta, Poonam Bansal, and Kavita Choudhary. 2018. The State of the Art of Feature Extraction Techniques in Speech Recognition. *Advances in Intelligent Systems and Computing*, 664:195–207.

Tanvira Ismail. 2020. A Survey of Language and Dialect Identification Systems. *Adalya*, 6(1).

Raymond D Kent and Houri K Vorperian. 2018. Static measurements of vowel formant frequencies and bandwidths: A review. *Journal off Communication Disorders*, 74:74–97.

Peter Ladefoged. 2006. *A course in phonetics*. Thomson, Wadsworth.

Leena Lulu and Ashraf Elnagar. 2018. Automatic Arabic Dialect Classification Using Deep Learning Models. In *The 4th International Conference on Arabic Computational Linguistics*, volume 142.

Matteo Negri, Marco Turchi, José G C De Souza, Daniele Falavigna,) Fbk -Fondazione, and Bruno Kessler. 2014. Quality Estimation for Automatic Speech Recognition. *Proceedings of COLING*, pages 1813–1823.

NIH. n.d. Data Book.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpu. 2015. Librispeech: An ASR corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE.

Ilias Papastratis. 2021. Speech Recognition: a review of the different deep learning approaches.

Chiyoun Park. 2008. *Consonant Landmark Detection for Speech Recognition by*. Ph.D. thesis, Massachusetts Institute of Technology.

Olga Scrivner. 2014. Vowel Variation in the Context of /s/: A Study of a Caracas Corpus. In Rafael Orozco, editor, *New Directions in Hispanic Linguistics*, chapter Vowel Vari. Cambridge Scholars Publishing.

S. Shahnawazuddin, Nagaraj Adiga, Hemant Kumar Kathania, and B. Tarun Sai. 2020. Creating speaker independent ASR system through prosody modification based data augmentation. *Pattern Recognition Letters*, 131:213–218.

Will Styler. 2011. Using Praat for Linguistic Research.

Ye Yi Wang, Alex Acero, and Ciprian Chelba. 2003. Is word error rate a good indicator for spoken language understanding accuracy. *2003 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2003*, pages 577–582.

Marc A. Zissman, Terry P. Gleason, Deborah M. Rekart, and Beth L. Losiewicz. 1996. Automatic dialect identification of extemporaneous, conversational, Latin American Spanish speech. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2.