Optimization of the data processing pipeline for pseudo-words in phonetics research

Sin Yu Bonnie Ho

The University of Münster, Germany

zbonnieho@uni-muenster.de

Abstract

In phonetics and phonology research, especially experimental phonetics, pseudo-words are often designed and employed to elicit target production. However, when it comes to data processing, many state-of-the-art pipelines for automatic phone segmentation favour real or natural speech. Therefore, one of the aims of my PhD project is to optimize the data processing pipeline for speech data containing pseudo-words using WebMAUS. In addition, acoustic feature extraction and annotation of linguistic factors are also attempted to be included in the pipeline.

Index Terms: phonetics and phonology, data processing pipeline, pseudo-words, automatic phone segmentation, WebMAUS

1. Motivation

The procedures of data processing in phonetics and phonology research often involve segmenting words into phones and labelling each segmented phone. Manual segmentation and labelling can be tedious and consume a lot of time and manpower. Although there are many state-of-the-art data processing pipelines (e.g. (Web)MAUS [1], [2]), they are mostly trained on real corpus data. In experimental phonetics, pseudo-words are often used for specific purposes. Many existing data pipelines were not designed for processing pseudo-words. Therefore, one of the aims of my PhD project is to optimize the data processing pipeline for pseudo-words using WebMAUS and to add more features such as acoustic characteristic extraction and linguistic factor annotation to the pipeline.

2. Data

A list of 256 syllables were designed with an onset singleton consonant, a monophthong as the nucleus, and a coda singleton consonant (CVC). The consonant was either a plosive or a fricative in English, whereas the vowel was one of the four vowels /i, e, u, a/. The syllables were then combined to form a disyllabic pseudo-word with the stress on the first syllable. An example of a pseudo-word is “zutfug” (/ˈzut.fug/). Each pseudo-word was embedded in the carrier phrase “Say _____ again”. A total of 27,136 target phrases were collected. They were recorded by 106 speakers in a sound-proof booth inside a phonetics laboratory.

3. Data processing pipeline

A pipeline was developed to process speech data containing pseudo-words (see Figure 1). Ideally, the pipeline should be able to automate and incorporate the common procedures of conducting acoustic analysis, including phone segmentation, phone labelling, and acoustic feature extraction. In reality,

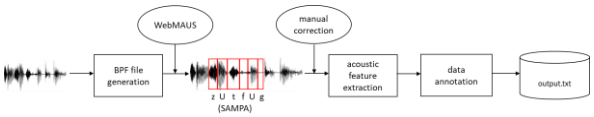

Figure 1: *Data processing pipeline.*

manual correction of the segmented phone boundaries is inevitable, especially when it comes to pseudo-words.

The pipeline was designed to run in the terminal; other tools such as Python 3 [3], Praat [4] and WebMAUS were also required.

3.1. BPF file generation

Due to the nature of pseudo-words, most existing open-source pre-trained language models cannot be applied to the data of this study directly. Each phone to speech signal path of the pseudo-word needs to be specified by the researcher for automatic phone segmentation. To this end, WebMAUS was adopted and the required BPF file was generated for each audio file. The format used in the BPF file was created by Schiel and colleagues [5] and it uses Speech Assessment Methods Phonetic Alphabet (SAMPA) symbols for canonical sound representations. Each audio file was transcribed phonemically using SAMPA symbols on a word level in the BPF file (see Table 1).

Table 1: *Example of a BPF file in this study.*

LHD:	Partitur 1.2
SAM:	48000
LBD:	
KAN:	0 seI
KAN:	1 zUtfUg
KAN:	2 @gen

3.2. WebMAUS (British English model vs. language independent model)

WebMAUS is the web service of “Munich AUtomatic Segmentation” (MAUS), which automatically segments and maps a speech signal into its phonetic segments [1, 2]. Given a selected language model, a set of probabilistic phonological (or pronunciation) rules is applied, and a-priori statistically weighted variants of the given phoneme are generated. Viterbi algorithm is adopted to select the most probable path using the acoustic probabilities from the acoustic model of the selected language [1]. For the purpose of this study, a phonemic segmentation and labelling was preferred instead of a phonetic one. Moreover, due to the fact that the input was not orthography but SAMPA transcript, forced alignment to input SAMPA was selected.

Two models, namely the British English model and the language independent model, were chosen for comparison. Although English vowels, fricatives, and plosives were used to construct the pseudo-words, certain combinations might violate the phonotactics in English. It means that some of the phone bigrams may not exist in the English model. Therefore, apart from the British English model, the language independent model was also chosen for comparing the segmentation performances.

3.2.1. Results

The automatic segmentation accuracy rate was calculated using a 20ms range compared with the manually labelled boundaries as the benchmark [6]. Preliminary results show that both the British English model and the language independent model had a relatively low accuracy, while the British English model performed slightly better than the language independent model (see Table 2).

Table 2: *Segmentation accuracy with different models*

Model	Accuracy (%)
British English	19.02
Language Independent	15.75

Figure 2 illustrates the wave form, the spectrogram, and automatic phone segmentation result of an audio file. There seems to be a tendency for the boundary to be shifted to the right.

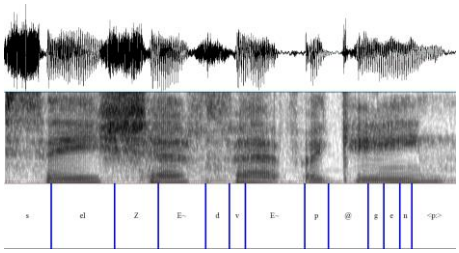

Figure 2: *Snapshot of the segmentation result.*

In the pipeline, the British English model was adopted for the automatic phone segmentation. All phone boundaries were then manually checked and corrected using Praat.

3.3. Acoustic feature extraction

In phonetics research, it is common to examine the acoustic features of certain phones, depending on the study. For now, the pipeline mainly focuses on extracting consonant features including the zero-crossing rate, the intensity, the four spectral moments (i.e. centre of gravity, standard deviation, skewness, kurtosis), and the harmonics-to-noise ratio.

3.4. Data annotation

The final step in the pipeline is annotating each segmented phone. Linguistic factors including syllable position (onset, coda), word position (initial, medial, final), stress pattern (stress, unstressed), as well as preceding and following phonetic environment were annotated automatically.

The output of the pipeline was a text file with all the time-aligned segmentation information, acoustic measurements of each segmented phone, and the annotated linguistic factors of each segmented phone.

4. Major contributions

Processing speech data for phonetics and phonology research studies is indeed time-consuming. While the technology of Natural Language Processing and Automatic Speech Recognition has been advancing, it seems that academia cannot benefit much from it, especially in the filed of experimental phonetics and acoustic phonetics. Phone segmentation and annotation remains a daily chore for many linguistics researchers. Therefore, while working on the collected speech data for my PhD project, I attempted to optimize the existing phone segmentation pipeline by adding features which can handle English data with pseudo-words, extract acoustic characteristics, and annotate linguistic factors to the segmented phone.

5. Future work

As can be seen in the results section, the phone segmentation accuracy rate is low and manual correction of the boundaries is necessary. One possible reason is that phonemic transcription was adopted instead of phonetic transcription as the input, which means the mapped phone to speech signal path could be false. However, assuming that the pseudo-words from the experiment do not exist in the English lexicon of the language model of WebMaus, forced alignment to the input phonemic transcripts would be the best choice. To improve the performance, a small set of the data can first be auditorily transcribed to explore what variants can occur. A set of pronunciation rules with possible paths can then be added to WebMAUS.

6. Acknowledgements

I would like to express my gratitude to my supervisor, Prof. Dr. Ulrike Gut for her guidance and support. I would also like to thank the developers and maintainers of WebMAUS for their help. Last but not least, many thanks to the Language Development Laboratory of the University of Hong Kong and the Speech and Language Sciences Laboratory of the Hong Kong Polytechnic University for assisting the data collection.

7. References

[1] F. Schiel, "Automatic Phonetic Transcription of Non-Prompted Speech," *In International Congress of Phonetic Sciences (ICPhS) Proceedings*, 1999, pp. 607-610.

[2] T. Kisler, U. Reichel, and F. Schiel, "Multilingual processing of speech via web services," *Computer Speech & Language*, vol. 45, pp. 326–347, 2017.

[3] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*, Scotts Valley, CA: CreateSpace, 2009.

[4] P. Boersma and D. Weenink, "Praat: doing phonetics by computer [Computer program]," version 6.1.16, retrieved 6 June 2020 from http://www.praat.org/.

[5] F. Schiel, S. Burger, A. Geumann, and K. Weilhammer, "The Partitur format at BAS," *Proceedings of the First International Conference on Language Resources and Evaluation*, 1998.

[6] A. Stolcke, N. Ryant, V. Mitra, J. Yuan, W. Wang, and M. Liberman, "Highly accurate phonetic segmentation using boundary correction models and system fusion," *In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Proceedings*, 2014, pp. 5552-5556.