RESYNTHESIZING THE GECO SPEECH CORPUS WITH VOCALTRACTLAB

Konstantin Sering[1], *Niels Stehwien*[1], *Yingming Gao*[2], *Martin V. Butz*[1], *Harald Baayen*[1]

[1]*Eberhard Karls Universität Tübingen,* [2]*TU Dresden*
konstantin.sering@uni-tuebingen.de

Abstract: We are addressing the challenge of learning an inverse mapping between acoustic features and control parameters of a vocal tract simulator. As a first step, we synthesize an articulatory corpus consisting of control parameters and wave forms using VocalTractLab (VTL; [1]) as the vocal tract simulator. The basis for the synthesis is a concatenative approach that combines gestures of VTL according to a SAMPA transcription. SAMPA transcriptions are taken from the GECO corpus [2], a spontaneous speech corpus of southern German. The presented approach uses the duration of the phones and extracted pitch contours to create gesture files for the VTL. The resynthesis of the GECO corpus results in 53960 valid spliced out word samples totalling in 6 hours and 23 minutes of synthesized speech. The synthesis quality is mediocre. We believe that the synthesized samples resemble some of the natural variability found in natural human speech.

1 Motivation

Constructing an articulatory corpus benefits many research fields, including automatic speech recognition [3], speech synthesis [4], acoustic-to-articulatory inversion [5] et al. There exist some articulatory corpora, such as Wisconsin X-ray microbeam database (XRMB) [6], MOCHA-TIMIT [7], MRI-TIMIT [8], which were successfully employed in above research fields. In the present paper, we aim at constructing an articulatory corpus using a vocal tract simulator as well as corresponding synthesized speech signals upon a spontaneous German speech corpus. Compared to hardware-based recorded corpora, it is not labor intensive and noninvasive to speakers. Moreover, unlike articulatory information of recorded images or limited measure points, it provides with rich representation of articulation process quantified by 30 control parameters and at a resolution of 10 milliseconds. These parameters can in turn be used to control articulatory synthesis.

Coming up with the control parameters for the vocal tract simulator is not an easy task. The two most prominent approaches to approximate the parameters that control a vocal tract simulator is firstly, to give the articulators in the vocal tract simulation different targets at different points in time and interpolate between these targets cleverly or secondly, define a set of gestures that define the trajectory of a subset of the articulators for a time interval. Using a gestural approach and allowing for gesture overlaps demands a rule to mix gestures. We believe that both of these approaches capture some of the structures that we see in human articulations but cannot account for the wide range of different articulation that is present in everyday natural speech.

We therefore seek to replace a rule based target or gesture approach that composes a small number of targets or gestures in a smart concatenative way, by modeling the structure of the whole trajectory in a more direct data driven way. One approach to generate trajectories without defining targets or gestures is to find a mapping between acoustic features and control parameter

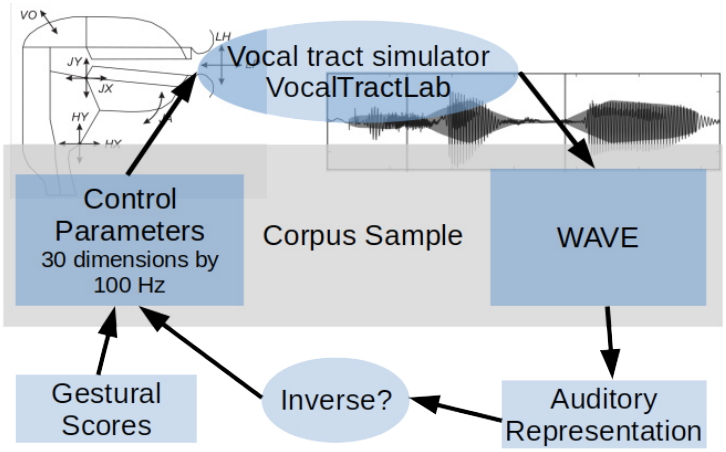

Figure 1 – The samples of the synthesized corpus are pairs of control parameters and wave files. The synthesized corpus will be used to train a mapping that inverts the vocal tract simulator.

trajectories of the vocal tract. Finding this mapping is not an easy task. One way is to estimate or learn the mapping from a corpus of training data.

In the following sections we present how we created a corpus of pairs, each consisting of control parameters and synthesized speech files. The corpus is made up of around 50,000 words that are based on the transcription and recording of southern German natural speech conversations in the GECO corpus [2]. We therefore hope to get a natural variability in the durations of the sounds and the corresponding control parameter trajectories, natural variability in word durations, shortend forms and an approximated natural word distribution.

We used a concatenative, dominance rule based approach to come up with the control parameters and the resulting synthesized speech files. The quality of the synthesized speech is mediocre. Our final goal is to replace the rule based approach with a finer grained, direct approach. The mediocre quality is acceptable to us, as the synthesized samples should be good enough to extract the above mentioned mapping between the synthesized speech and the control parameters.

1.1 Vocal tract simulator

We choose to use VocalTractLab 2.2 (VTL; [9] [1]) as the vocal tract simulator, as it provides us with a good repertoire of gestures for German phones.

VTL is a geometrical vocal tract simulator and an acoustic synthesis model. VTL makes it possible to synthesize speech from control parameters that define the geometrical shape of the vocal tract, as well as the properties of the glottis model and lung pressure for each time step. In VTL there are around 30 different control parameters that need to be specified at least once every 10 milliseconds. Up until now, control parameters have been derived mostly by a dominance model through composition of gestural scores. However, the creation of gestural scores and their composition involves a considerable amount of handcrafting. VTL is not a biomechanical model, therefore no physiological constraints other than shape constraints are present.

The VTL can be seen as a mapping between a space of control parameters and wave forms. The aim of our research project is to automatically infer control parameters from speech samples, which amounts to inverting the VTL (Figure 1). One of the challenges in learning an approximate inverse model is that there are around 30 different control parameters, which need to be adjusted every 10 milliseconds. As a result, the control parameter input space (a fine grained time series) grows quickly. A further complication is due to motor equivalence phenomena in speech production [10], in our case there are several different sets of control parameters

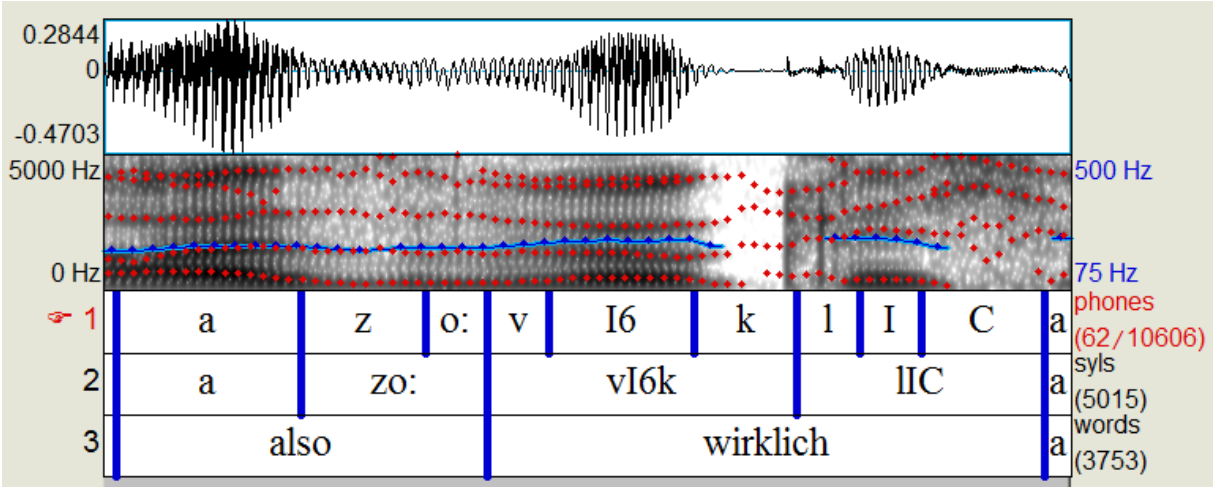

Figure 2 – GECO corpus annotates the studio-recorded German spontaneous speech on three tiers: word tier, syllable tier, and phone tier (from bottom to top).

that lead to the same synthesized speech, therefore a unique inverse function to the VTL does not exist. Nevertheless, an inverse function can most likely be approximated by training on a large and phonetically diverse corpus in combination with additional physical and physiological constraints, such as boundaries for postural values as well as for speed and acceleration. The targeted approximation of the inverse function will be part of an adapted dynamical artificial neuronal network, which has recently shown to be able to control various simulated dynamical systems (cf. [11]).

1.2 Source corpus

As the source corpus, which functions with its phonetic transcription and its pitch contours as the basis to synthesize data, we use the GECO corpus [2]. The GECO corpus is a spontaneous speech corpus containing roughly 20 hours of studio recordings of spontaneous dialog with phonetic annotation on the word, syllable and phone level of 20 female speakers.

Spontaneous speech is a reliable source of an adequate range of sounds used in speech. Using a spontaneous speech corpus, we compromise between precise resynthesis of reliably intelligible speech and an encompassing body of sounds used in spoken language. In the task of finding the acoustic features to control parameter mappings we are interested in a lot of natural variation and not so much in high quality synthesis in an artificial setting.

2 Method

For the creation of the training corpus, we currently rely on handcrafted gestural scores [1] and some basic heuristics to compose them in a reasonable way. Currently, the main goal is not to create high quality human-like, valid samples, but to obtain a diverse range of samples with different speech qualities as output. The present training corpus contains many transitions between most German phones.

Figure 3 gives an overview of the logic that is behind the resynthesis of the GECO corpus with VTL. The code that implements the logic is written in Python and is freely available. It can be downloaded from [12].

In order to resynthesize the human speech recordings we started with the TextGrid annotation of the GECO corpus. The annotation has three tiers as shown in Figure 2: a word level tier, which segments the recorded audio into accordingly labeled words and pauses, a syllable tier and a phones tier. The latter two tiers are annotated with the SAMPA transcription.

As the GECO corpus contains natural speech there are some very short words. Unfortunately, the rule based approach we implemented had the limitation that it could not handle words shorter than 150 milliseconds and words with only one syllable. Therefore we exclude these words for now.

The rule based compositor we use here is based on the work of our colleague Yingming Gao and was extended by us to obey the durations of the phones. The compositor implements a concatenative approach that subsequently selects different gestural scores of VTL, depending on the phone transcription and the duration of each phone. If the last phone of a word was a stop, i. e. ending with one of the following phones /p/, /t/, /k/, /b/, /d/ or /g/, we appended another empty gesture of 30 milliseconds. We do this in order to make the stop release happen and acoustically visible. As a result of that, the synthesized words that end with a stop have a 30 millisecond longer duration, those that do not end with a stop have the same duration as the original spliced out word. After running the compositor on a SAMPA transcription of one word it creates a plain VTL gesture file, "plain" denoting its arbitrary, flat f0-contour.

The syllabic tier of the GECO TextGrid is taken into account in two ways. On the one hand, syllable boundries are used as input for the PhonesToGes tool. On the other hand, pitch contours have been extracted with Praat [13] and then fitted by TargetOptimizer [14] with one target pitch for each syllable and varying transition speeds between the target pitch depending on what fits the extracted pitch contour of the recorded data best. The slope of the target pitch within each syllable was limited to $[-1, +1]$ from the default of $[-10, +10]$. Limiting the slope this way leads to degraded fitting of the extracted pitch data, but to better quality of synthesis, as the extrapolation of pitch in a long syllable to the end cannot become very large or very small anymore. Fitting the curves with TargetOptimizers default slope value led to extremely high pitched sounding synthesis or to "sweeps".

In the process of synthesis, we generated the following intermediate data: wave files (.wav), spliced out from the original recordings in the GECO; an utterance file (.txt) with SAMPA transcription and durations of the phones for each word in the GECO; VTL gesture files (.ges), generated heuristically and rule-based, that are "plain" in the sense that they have an arbitrary, constant f0-contour (pitch); pitch contours (.PitchTier), extracted from spliced out wav recordings by Praat, f0-gestures (.ges), fitted by TargetOptimizer.

We now combine the fitted f0-gestures and the VTL gesture files to obtain VTL gestures files with an approximated f0-contour. These merged gesture files are used to derive control parameters for VTL and synthesize a wave form from the control parameters.

The GECO corpus contains 74575 words of duration longer than 150 milliseconds and at least two syllables. Our compositor could create plain gesture files for 74418 words due to some matching or parsing errors. Praat extracted the pitch for 74548 words. The TargetOptimizer successfully fitted 66775 pitches. During the pitch optimization we lost around 7773 words due to lack of data or failure of fitting the pitch. Combining the gesture files resulted in 57079 fixed gesture files. In the last step of synthesizing the fixed gesture files another 3119 files were lost due to errors in the simulation process. Finally, 53960 samples of control parameters and their respective synthezised wave files were created. Most of the synthesized words ended up being intelligible.

3 Corpus description

The resynthesized corpus consists of 53960 samples, which are paris of control parameters and wave forms. As byproducts the area function, the gestural score files, the pitch contours and the phone transcription are stored alongside. The original wave recordings can easily be generated from the GECO corpus, if needed for comparison, but cannot be part of the published corpus

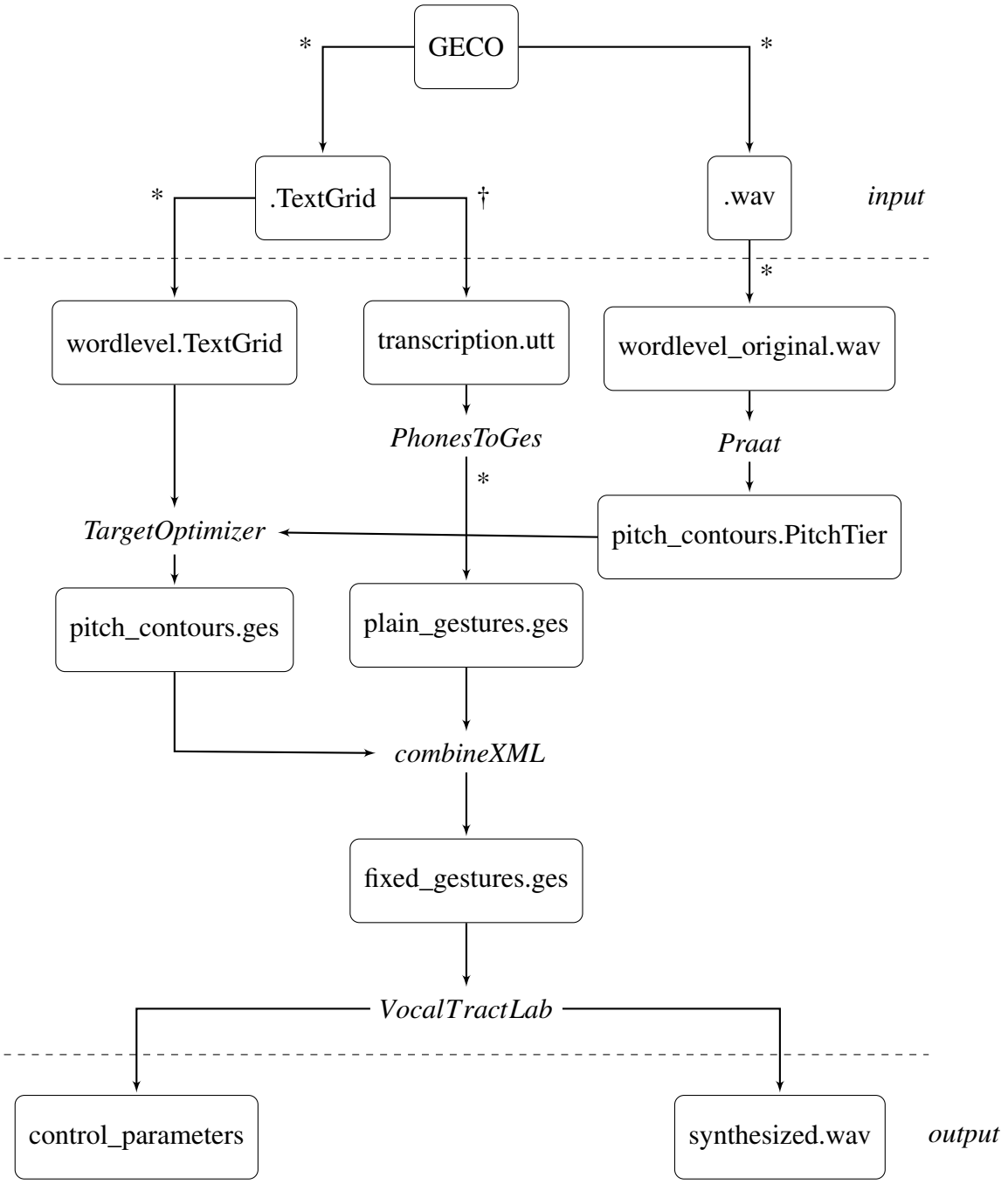

Figure 3 – *Flow of data in the corpus creation. As input, GECO corpus data is used to derive control parameters and synthesize speech. Asterisks denote file splitting, the dagger denotes file merging. All other steps operate on word-level data. As output, the final corpus contains control parameters and synthesized words as wave files.*
Praat [13], VocalTractLab [1] and TargetOptimizer [14] are separate programs, the rest of the logic is implemented in Python. The Python scripts are freely available from [12]

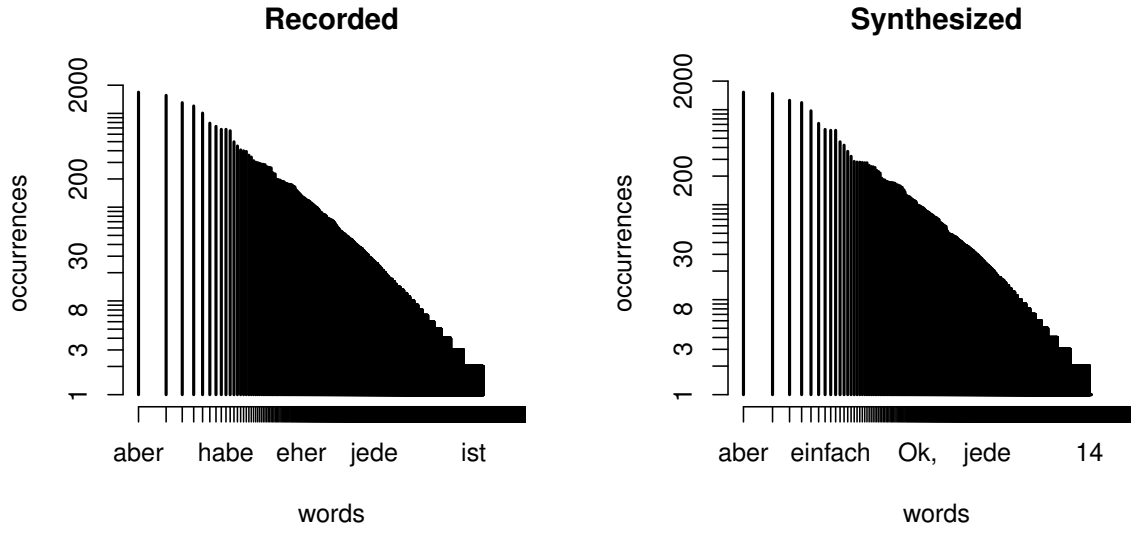

Figure 4 – Word frequency distributions for the 74575 spliced out words from the GECO corpus and the 53960 synthesized words. The log-log plot shows a roughly linear relationship between rank and word frequency, therefore both distributions seem to follow Zipf-law.

due to licensing issues.

The samples have a varying duration between 190 milliseconds and 2.1 seconds and a summed duration of 6 hours and 23 minutes. The corpus consists of 10028 different words. The frequency distribution of the different words can be seen in Figure 4. The frequency distribution resembles the distribution in the recorded data and matches a Zipf-distribution closely.

The intelligibility of the synthesized speech is mediocre to poor but the sound variation and the coocurences of different sounds with different durations seem to mimic some aspects of human language like having similarity with the prosody of the original recordings and similar sound patterns. Some words are not intelligible. We believe that there are some parameters we can further adjust to achieve better synthesis. For instance, phones as transcribed in the GECO can be very short. VTL gestures have a lower bound for how fast a transition to the target value can occur. This inertia in the model prevents very short sounds from being articulated in synthesis. Another Issue is that the synthesized speech often sounds too high pitched. we assume that this is due to errors in the pitch detection, for instance when frictatives are falsely identified as having a f0-contour with a high value.

4 Conclusion and Outlook

We hope this corpus is a first building block to create an automated, flexible speech synthesis system that does not use symbolic building blocks and rules, but rather creates audio by driving a vocal tract simulator in a fine grained way. As a first step in this direction we will learn a mapping between acoustic features similar to the ones presented in [15] that are extracted from the wave forms and the control parameters that belong to these wave forms.

The corpus itself can be improved in several ways. On the one hand synthesizing samples that capture a specific structure found in German language like cooarticulations between German phones. This could be achieved by using a model proposed and validated in [9], and some random variation in gestural score compositions. Another direction of enhancement is adding additional information to the samples like tongue tracker movements. This is especially interesting to us as we are planning to resynthesizing the Karls-Eberhard-Corpus which comes with articulatory measurements. At the moment we are working on extracting the tongue positions

over time out of VTL in an automated way. In the end we hope that we can enhance the corpus presented here with EMA-like trajectories.

References

[1] BIRKHOLZ, P.: 2018. URL http://www.vocaltractlab.de/index.php?page=vocaltractlab-about.

[2] SCHWEITZER, A. and N. LEWANDOWSKI: *Convergence of articulation rate in spontaneous speech.* In *INTERSPEECH*, pp. 525–529. 2013.

[3] WRENCH, A. A.: *A multichannel articulatory database and its application for automatic speech recognition.* In *In Proceedings 5 th Seminar of Speech Production*, pp. 305–308. 2000.

[4] HUEBER, T., G. CHOLLET, B. DENBY, M. STONE, and L. ZOUARI: *Ouisper: corpus based synthesis driven by articulatory data.* In *16th International Congress of Phonetic Sciences*, pp. 2193–2196. 2007.

[5] RICHMOND, K.: *Preliminary inversion mapping results with a new ema corpus.* 2009.

[6] WESTBURY, J.: *X-ray microbeam speech production database user's handbook: Madison. WI: Waisman Center, University of Wisconsin*, 1994.

[7] WRENCH, A.: *The mocha-timit articulatory database.* 1999.

[8] NARAYANAN, S., A. TOUTIOS, V. RAMANARAYANAN, A. LAMMERT, J. KIM, S. LEE, K. NAYAK, Y.-C. KIM, Y. ZHU, L. GOLDSTEIN ET AL.: *Real-time magnetic resonance imaging and electromagnetic articulography database for speech production research (tc). The Journal of the Acoustical Society of America*, 136(3), pp. 1307–1311, 2014.

[9] BIRKHOLZ, P.: *Modeling consonant-vowel coarticulation for articulatory speech synthesis. PLOS ONE*, 8(4), pp. 1–17, 2013. doi:10.1371/journal.pone.0060603. URL https://doi.org/10.1371/journal.pone.0060603.

[10] PERRIER, P. and S. FUCHS: *11 motor equivalence in speech production. The handbook of speech production*, p. 225, 2015.

[11] BUTZ, M. V., D. BILKEY, D. HUMAIDAN, A. KNOTT, and S. OTTE: *Learning, planning, and control in a monolithic neural event inference architecture. arXiv preprint arXiv:1809.07412*, 2018.

[12] SERING, K., N. STEHWIEN, and Y. GAO: *create_vtl_corpus: Synthesizing a speech corpus with VocalTractLab.* 2019. doi:10.5281/zenodo.2548895. URL https://doi.org/10.5281/zenodo.2548895.

[13] BOERSMA, P. and D. WEENINK: *Praat: doing phonetics by computer (version 6.0.46).* 2019. URL http://www.praat.org.

[14] SCHMAGER, P.: 2017. URL http://www.vocaltractlab.de/index.php?page=targetoptimizer-about.

[15] ARNOLD, D., F. TOMASCHEK, K. SERING, F. LOPEZ, and R. H. BAAYEN: *Words from spontaneous conversational speech can be recognized with human-like accuracy by an error-driven learning algorithm that discriminates between meanings straight from smart acoustic features, bypassing the phoneme as recognition unit*. *PloS one*, 12(4), p. e0174623, 2017.