Automatic measurement of vowel duration via structured prediction

Yossi Adi[a)], Joseph Keshet[b)]

Department of Computer Science, Bar-Ilan University, Ramat-Gan, Israel, 52900

Emily Cibelli[c)], Erin Gustafson[d)]

Department of Linguistics, Northwestern University, Evanston, IL, 60208

Cynthia Clopper[e)]

Department of Linguistics, Ohio State University, Columbus, OH, 43210

Matthew Goldrick[f)]

Department of Linguistics, Northwestern University, Evanston, IL, 60208

[a)]e-mail: adiyoss@cs.biu.ac.il

[b)]e-mail: joseph.keshet@biu.ac.il

[c)]e-mail: emily.cibelli@northwestern.edu

[d)]e-mail:egustafson@u.northwestern.edu

[e)]e-mail:clopper.1@osu.edu

[f)]e-mail: matt-goldrick@northwestern.edu

Abstract

A key barrier to making phonetic studies scalable and replicable is the need to rely on subjective, manual annotation. To help meet this challenge, a machine learning algorithm is developed for automatic measurement of a widely used phonetic measure: vowel duration. Manually-annotated data is used to train a model that takes as input an arbitrary length segment of the acoustic signal containing a single vowel that is preceded and followed by consonants and outputs the duration of the vowel. The model is based on the structured prediction framework. The input signal and a hypothesized set of a vowel's onset and offset are mapped to an abstract vector space by a set of acoustic feature functions. The learning algorithm is trained in this space to minimize the difference in expectations between predicted and manually-measured vowel durations. The trained model can then automatically estimate vowel durations without phonetic or orthographic transcription. Results comparing the model to three sets of manually annotated data suggest it out-performs the current gold standard for duration measurement, an HMM-based forced aligner (which requires phonetic transcription as an input).

PACS Numbers: 43.70.Jt, 43.72.Ar, 43.72.Lc, 43.70.Fq

1 INTRODUCTION

Understanding the factors that modulate vowel duration has been a long-standing focus of laboratory research in acoustic phonetics (Peterson and Lehiste, 1960). The vast majority of such research–from the mid-20th century to the start of the 21st–has relied on manual annotation to determine vowel duration. There are two key issues with such an approach. Given the labor intensive nature of such annotation, there are substantial limitations on the amount of data that can be practically analyzed; this limits the statistical power and types of issues that phonetic studies can address. Furthermore, given the fundamental reliance on subjective annotator judgments, analyses cannot be directly replicated by other researchers.

An alternative to this approach are algorithms for automatic alignment of vowel boundaries. The current standard approach for vowels is to utilize forced alignment algorithms (Yuan et al., 2013). However, this approach suffers from two important limitations: it requires a phonetic transcription, and frequently requires substantial preprocessing of the data to insure adequate performance (Evanini, 2009). Another approach is to use a binary classifier to detect for each frame whether it contains a vowel of not. On the one hand the advantage of this method is that the algorithm doesn't need the phonetic transcription and preprocessing is not required. On the other hand, since the algorithm process one frame at a time, this method does not take into account the relation between the start and end times of

the vowel. Moreover, in order to get the vowel duration, an additional algorithm is required (Adi et al., 2015).

In this paper we propose a method for automatic measurement of vowel duration using structured prediction techniques which have provided excellent results in analyzing other phonetic measures (Keshet et al., 2007; Sonderegger and Keshet, 2012). The algorithm was trained at the segment level on manually annotated data to extract the vowel start and end times; this provides a straightforward way to compute the duration of the vowel. Following the structure of vast majority of laboratory studies of vowel duration, we assume the input signal contains a single vowel, proceeded and followed by a consonant (CVC)–no additional detailed information about the phonetic transcription is required to process the speech.

We evaluated our method on data from three phonetic studies: one focusing on vowel duration (Heller and Goldrick, 2014), the second using vowel segmentation to automatically determine points for formant analysis (Clopper and Tamati, 2014), and the third a standard set of vowel production norms (Hillenbrand et al., 1995). We compared results with our model to the state-of-the-art in vowel duration measurement, HMM-based forced alignment (Rosenfelder et al., 2014). We also assessed whether inferential statistical models fit to data from our model and HMM-based forced alignment replicated the patterns obtained from manual data. The results suggest that our algorithm is superior to the current gold standard at matching the manual measurements of vowel duration, both in terms of deviation and in

replicating inferential statistical results.

The paper is organized as follows. In Section 2 we state the problem definition formally. We then present the learning framework (Section 3), algorithm (Section 4), and the acoustic features and feature functions (Section 5). In Section 6 we describe the datasets we use to train the models and evaluate the performance of the algorithm. In Section 7 we detail the particular methods used to implement our algorithm here, along with the standard approach to vowel duration measurement. Experimental results are detailed in two sections: the first focusing on measurement deviation (Section 8) and the second on the reproduction of results from inferential statistical models (Section 9). We conclude the paper with possible applications and extensions in Section 10.

2 PROBLEM SETTING

In the context of a typical laboratory study of speech, the goal of automatic vowel duration measurement is to accurately predict the time difference between the vowel onset and offset, given a segment of the acoustic signal in which a vowel preceded and followed by consonants. The acoustic sample can be of any length, but should include only one vowel[g)]. We assume

[g)] When more than one vowel is presented in the input utterance, we first apply a force aligner to roughly find the location of the desired vowel and then provide the portion of the utterance that include the vowel with its preceding and following consonants.

there is a small portion of silence before and after the uttered speech, but do not require the beginning of the speech signal or the vowel onset are synchronized with the onset or offset of the acoustic sample.

We turn to describing the problem formally. Throughout the paper we write scalars using lower case Latin letters, e.g., x, and vectors using bold face letters, e.g., $\boldsymbol{x}$. A sequence of elements is denoted with a bar $\bar{x}$ and its length is written as $|\bar{x}|$. Similarly a sequence of vectors is denoted as $\bar{\boldsymbol{x}}$ and its length by $|\bar{\boldsymbol{x}}|$.

The acoustic sample is represented by a sequence of acoustic feature vectors denoted by $\bar{\boldsymbol{x}} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T)$, where each $\boldsymbol{x}_t$ $(1 \leq t \leq T)$ is a d-dimensional vector that represents the acoustic content of the t-th frame. The domain of the feature vectors is denoted as $\mathcal{X} \subset \mathbb{R}^d$. The input acoustic sample can be of an arbitrary length, thus T is not fixed. We denote by $\mathcal{X}^*$ the set of all finite length segments of acoustic signal over $\mathcal{X}$. In addition, we denote by $t_b \in \mathcal{T}$ and $t_e \in \mathcal{T}$ the vowel onset and offset times in frame units, respectively, where $\mathcal{T} = \{1, ..., T\}$, and the total duration of the input acoustic sample is T frames. For brevity we denote this pair by $\boldsymbol{t} = (t_b, t_e)$, and call it *onset-offset pair*. Practically there are constraints on t_b and t_e and they cannot take any value in $\mathcal{T}$, e.g., the vowel onset t_b cannot be T or $T - 1$. Our notation is depicted in Figure 1.

Our goal is to find a function f from the domain of segments of acoustic signal, $\mathcal{X}^*$, to the domain of all onset-offset pairs, $\mathcal{T}^2$. Given a segment of the acoustic signal $\bar{\boldsymbol{x}}$, let

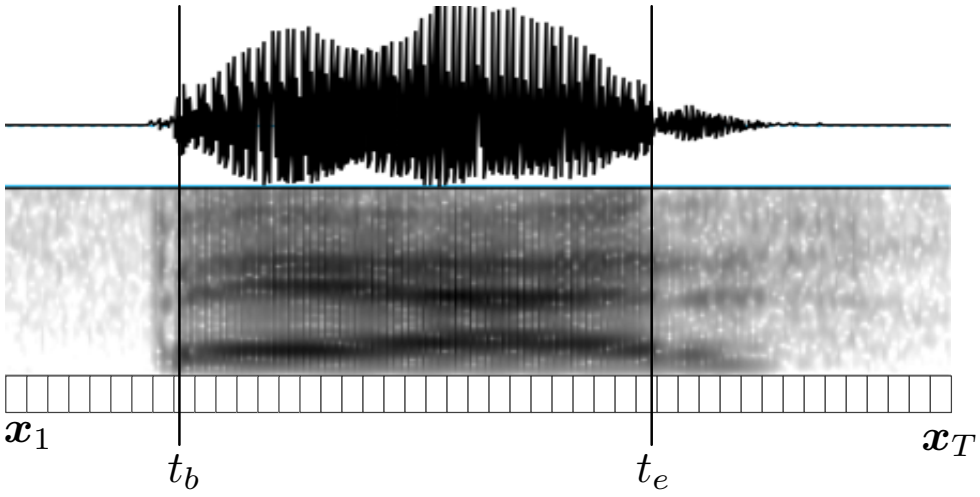

Figure 1: An example of our notation. The top panel presents the signal in the time domain, and the middle panel presents the spectrogram of the signal. The vertical solid lines present the annotated vowel's onset t_b and offset t_e. The speech signal is represented by a sequence of acoustic feature vectors as depicted in the lower panel. For the t-th frame, the acoustic feature vector is denoted by $\boldsymbol{x}_t$.

$\hat{\boldsymbol{t}} = f(\bar{\boldsymbol{x}})$ be the predicted onset-offset pair, where $\hat{\boldsymbol{t}} = (\hat{t}_b, \hat{t}_e)$. The quality of the prediction is assessed using a *cost function*, denote by $\gamma(\boldsymbol{t}, \hat{\boldsymbol{t}})$, that measures the magnitude of the penalty when predicting the pair $\hat{\boldsymbol{t}}$ rather than the target pair $\boldsymbol{t}$. Formally, $\gamma : \mathcal{T}^2 \times \mathcal{T}^2 \to \mathbb{R}_+$ is a function that receives as input two ordered pairs, and returns a positive scalar. We assume that if both the predicted and the target pairs are the same, then the cost is zero, $\gamma(\boldsymbol{t}, \boldsymbol{t}) = 0$.

3 LEARNING FRAMEWORK

We assume that an input acoustic sample and a target onset-offset pair are drawn from a fixed but unknown distribution ρ over the domain of the segments of acoustic signal and the vowel onset-offset pairs, $\mathcal{X}^* \times \mathcal{T}^2$. We define the *risk* of f as the expected cost when using f to predict the onset-offset pair of the acoustic sample $\bar{\boldsymbol{x}}$, that is,

$$R(f) = \mathbb{E}_{(\bar{\boldsymbol{x}},\boldsymbol{t})\sim\rho}[\gamma(\boldsymbol{t}, f(\bar{\boldsymbol{x}}))], \tag{1}$$

where the expectation is taken with respect to the input acoustic sample $\bar{\boldsymbol{x}}$ and the annotated onset-offset pair $\boldsymbol{t}$ drawn from ρ. Our goal is to find f that minimizes the risk. Unfortunately, this cannot be done directly since ρ is unknown. Instead, we use a training set of examples that is served as a restricted window through which we can estimate the quality of the prediction function according to the distribution of unseen examples in the real world. The examples are assumed to be identically and independently distributed (i.i.d.) according to the distribution ρ.

Each example in the training set is composed of a segment of the acoustic signal and a manually annotated vowel onset and offset pair. The manual annotations are not exact, and naturally depend both on human errors as well as objective difficulties in placing the vowel boundaries, e.g., between the vowel and a sonorant. Hence in this work we focus on a cost

function that inherently takes into account the discrepancy in the annotations. That is,

$$\gamma(\boldsymbol{t}, \hat{\boldsymbol{t}}) = \left[|\hat{t}_b - t_b| - \tau_b\right]_+ + \left[|\hat{t}_e - t_e| - \tau_e\right]_+, \tag{2}$$

where $[\pi]_+ = \max\{0, \pi\}$, and τ_b, τ_e are pre-defined parameters. The above function measures the absolute differences between the predicted and the target vowel onsets and offsets. It allows a mistake of τ_b and τ_e frames at the onset and offset of the vowel respectively, and only penalizes predictions that are greater than τ_b or τ_e frames.

Our learning model belongs to the structured prediction framework. In this framework it is assumed that the output prediction is complex and has some internal structure. In our case, the vowel onset and vowel offset times are related and dependent, e.g., the vowel has a typical duration that depends on the vowel onset and offset.

In the structured prediction model, the function f is based on a fixed mapping $\boldsymbol{\phi} : \mathcal{X}^* \times \mathcal{T}^2 \to \mathbb{R}^n$ from the set of segments of acoustic signal and target onset-offset pairs to a real vector of length n; we call the elements of this mapping *feature functions* or *feature maps*. Intuitively, the feature functions represent our knowledge regarding good locations of the onset or the offset of the vowel within the acoustic signal. For example, consider Figure 1. It can be seen that the gradient of the spectrum is high at the areas of t_b and t_e. One feature function can be the distance between the spectrum a frame before and a frame after the presumed t_b. This function is going to be high if the presumed t_b is in the vicinity of the actual target vowel onset and is going to be low at a random place.

Our prediction function is a linear decoder with a vector of parameters $\boldsymbol{w} \in \mathbb{R}^n$ that is defined as follows:

$$f_{\boldsymbol{w}}(\bar{\boldsymbol{x}}) = \arg\max_{\hat{\boldsymbol{t}} \in \mathcal{T}^2} \ \boldsymbol{w}^\top \boldsymbol{\phi}(\bar{\boldsymbol{x}}, \hat{\boldsymbol{t}}). \tag{3}$$

The subscript $\boldsymbol{w}$ is added to the function f to stress that it depends on the weight vector $\boldsymbol{w}$.

Ideally, we would like our learning algorithm to find $\boldsymbol{w}$ such that the prediction minimizes the cost on unseen data. Recall, we assume there exists some unknown probability distribution ρ over pairs $(\bar{\boldsymbol{x}}, \boldsymbol{t})$. We would like to set $\boldsymbol{w}$ so as to minimize the expected cost, or the *risk*, for predicting $f_{\boldsymbol{w}}(\bar{\boldsymbol{x}})$,

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \ \mathbb{E}_{(\bar{\boldsymbol{x}}, \boldsymbol{t}) \sim \rho}[\gamma(\boldsymbol{t}, f_{\boldsymbol{w}}(\bar{\boldsymbol{x}}))]. \tag{4}$$

It is hard to directly minimize this objective function since ρ is unknown and the cost γ is often combinatorial non-convex function (Keshet, 2014). In the next section we describe the learning algorithm that aims at minimizing the risk, and then we describe the set of feature functions in Section 5.

4 DIRECT COST MINIMIZATION (DCM) ALGORITHM

Recall that our goal is to directly optimize the objective in Eq. (4). Unfortunately, if the output space is discrete we can not use direct gradient decent since the cost $\gamma(\boldsymbol{t}, f_{\boldsymbol{w}}(\bar{\boldsymbol{x}}))$ is not a differentiable function of $\boldsymbol{w}$ (Keshet, 2014). McAllester et al. (2010) showed that if

the input space $\mathcal{X}^*$ is continuous, we can compute the gradient of the expected cost, i.e., the risk, in Eq. (4) even when the output space is discrete in terms of the feature functions. Specifically the gradient can be expressed in a closed form solution as follows:

$$\nabla_{\boldsymbol{w}} \mathbb{E}\Big[\gamma(\boldsymbol{t}, f_{\boldsymbol{w}}(\bar{\boldsymbol{x}})\Big] = \lim_{\epsilon \to 0} \frac{\mathbb{E}\Big[\boldsymbol{\phi}(\bar{\boldsymbol{x}}, f^{\epsilon}_{\boldsymbol{w}}(\bar{\boldsymbol{x}})) - \boldsymbol{\phi}(\bar{\boldsymbol{x}}, f_{\boldsymbol{w}}(\bar{\boldsymbol{x}}))\Big]}{\epsilon}, \tag{5}$$

where the expectation on both sides is with respect to the tuple $(\bar{\boldsymbol{x}}, \boldsymbol{t})$ drawn from ρ, and $f^{\epsilon}_{\boldsymbol{w}}$ is defined as follows:

$$f^{\epsilon}_{\boldsymbol{w}}(\bar{\boldsymbol{x}}) = \arg\max_{\hat{\boldsymbol{t}} \in \mathcal{T}^2} \; \boldsymbol{w}^{\top} \boldsymbol{\phi}(\bar{\boldsymbol{x}}, \hat{\boldsymbol{t}}) + \epsilon\, \gamma(\boldsymbol{t}, \hat{\boldsymbol{t}}). \tag{6}$$

Using stochastic gradient decent we get the following update rule:

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \frac{\eta_t}{\epsilon} \left(\boldsymbol{\phi}(\bar{\boldsymbol{x}}, f_{\boldsymbol{w}}(\bar{\boldsymbol{x}})) - \boldsymbol{\phi}(\bar{\boldsymbol{x}}, f^{\epsilon}_{\boldsymbol{w}}(\bar{\boldsymbol{x}}))\right), \tag{7}$$

where η_t is the learning rate. At training time, we set $\eta_t = \eta_0/\sqrt{t}$, where η_0 is a parameter and t is the iteration number, and we set ϵ as a fixed small parameter, which is selected from a held-out development set. The actual values of the parameters are detailed in Section 8.

Since the objective in Eq. (5) is not a convex function in the model parameters $\boldsymbol{w}$, gradient descent is not guaranteed to find the optimal parameter settings; it may converge to a local minimum. We initialize the model parameters with a weight vector of parameters that was pre-trained using the structured prediction passive-aggressive (PA) algorithm (Crammer et al., 2006). The vector of parameters that was obtained from the PA training is denoted by $\boldsymbol{w}_{\text{PA}}$. A pseudo code of the training algorithm is given in Figure 2.

INPUT: training set $S = \{(\bar{\boldsymbol{x}}_i, \boldsymbol{t}_i)\}_{i=1}^{m}$; parameters η_0, ϵ

INITIALIZATION: $\boldsymbol{w}_1 = \boldsymbol{w}_{\text{PA}}$

FOR $t = 1, 2, \ldots,$ T

Pick example $(\bar{\boldsymbol{x}}_i, \boldsymbol{t}_i)$ from S uniformly at random

Predict $\quad \hat{\boldsymbol{t}} = \arg\max_{\hat{\boldsymbol{t}}} \ \boldsymbol{w}_t^\top \boldsymbol{\phi}(\bar{\boldsymbol{x}}_i, \hat{\boldsymbol{t}})$

Predict $\quad \hat{\boldsymbol{t}}^\epsilon = \arg\max_{\hat{\boldsymbol{t}}} \ \boldsymbol{w}_t^\top \boldsymbol{\phi}(\bar{\boldsymbol{x}}_i, \hat{\boldsymbol{t}}) + \epsilon\, \gamma(\boldsymbol{t}_i, \hat{\boldsymbol{t}})$

Set $\quad \boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \frac{\eta_0}{\epsilon\sqrt{t}}(\boldsymbol{\phi}(\bar{\boldsymbol{x}}_i, \hat{\boldsymbol{t}}) - \boldsymbol{\phi}(\bar{\boldsymbol{x}}_j, \hat{\boldsymbol{t}}^\epsilon)$

OUTPUT: the average weight vector $\quad \boldsymbol{w}^* = \frac{1}{T}\sum_t \boldsymbol{w}_t$

Figure 2: Direct cost minimization training procedure.

5 FEATURES AND FEATURE FUNCTIONS

In this section we describe the acoustic features $\boldsymbol{x} \in \mathcal{X}$ and the feature functions $\boldsymbol{\phi}(\bar{\boldsymbol{x}}, \boldsymbol{t})$, which were designed specifically for the problem of vowel duration measurement.

5.1 Acoustic Features

The main function of the acoustic features representation is to preserve the crucial information-bearing elements of the speech signal and to suppress irrelevant details. We extracted $d = 16$ acoustic features every 5 ms, in a similar way to the feature set used in Sonderegger and Keshet (2012), but with different time spans. The first 5 features are based on the short-time Fourier transform (STFT) taken with a 25 ms Hamming window. The features are short-term energy, $E_{\text{short-term}}$; the log of the total spectral energy, E_{total}; the log of the energy between 20 and 300 Hz, E_{low}; the log of the energy above 3000 Hz, E_{high}; and the Wiener entropy, H_{wiener}, a measure of spectral flatness (Sonderegger and Keshet, 2012). The sixth feature, S_{max}, is the maximum of the power spectrum calculated in a region from 6 ms before to 18 ms after the frame center. The seventh feature, $\hat{F}_0$, is the normalized fundamental frequency estimator, extracted using the algorithm of Sha et al. (2004) every 5 ms, and smoothed with a Hamming window. The eighth feature is the binary output of a voicing detector based on the RAPT pitch tracker (Talkin, 1995), smoothed with a Hamming window - denoted as

V_{RAPT}. The ninth feature is the number of zero crossings in a 5 ms window — denoted as N_{ZC}.

The next set of acoustic features are based on a phoneme classifier's predictions and scores (Dekel et al., 2004). The tenth acoustic feature is an estimated probability of whether a vowel is uttered at the input frame. The feature is a smoothed version of an indicator function that states if the phoneme predicted by the classifier is a vowel. The eleventh feature is defined to be the same as the tenth feature, but is specific to nasal phonemes. These features are dented as G_{vowel} and G_{nasal}, respectively. The twelfth feature is the likelihood of a vowel at the current time frame, L_{vowel}. The likelihood is computed as the Gibbs measure of the phoneme classifier's scores of all the vowel phonemes.

The last four features are based on the spectral changes between adjacent frames, using Mel-frequency cepstral coefficients (MFCCs) to represent the spectral properties of the frames. Define by $D_j = d(\boldsymbol{a}_{t-j}, \boldsymbol{a}_{t+j})$ the Euclidean distance between the MFCC feature vectors $\boldsymbol{a}_{t-j}$ and $\boldsymbol{a}_{t+j}$, where $\boldsymbol{a}_t \in \mathbb{R}^{39}$ for $1 \leq t \leq T$. The features are denoted by D_j, for $j \in \{1, 2, 3, 4\}$.

Figure 3 shows the trajectories of some of features for a typical vowel in a CVC context (the word "got").

Figure 3: Values of some of the acoustic features for an example acoustic sample (the word "got"). The vertical dashed lines indicate the annotated onset and offset of the vowel.

5.2 Feature Functions

We turn now to describe the feature functions. Recall that the feature functions are designed to be correlated with a good positioning of the onset-offset pair, $\boldsymbol{t}$, in the acoustic signal, $\bar{\boldsymbol{x}}$. While generally each feature function $\phi_i(\bar{\boldsymbol{x}}, \boldsymbol{t})$, for $1 \leq i \leq n$, gets as input a sequence of acoustic features, $\bar{\boldsymbol{x}}$, and a presumed onset-offset pair $\boldsymbol{t}$, they can practically use only a subset of the acoustic features (e.g., only a sequence over the sixth feature). Some of the feature functions are based on the average of an acoustic feature x from frame t_1 to frame t_2 defined as

$$\mu(\bar{x}, t_1, t_2) = \frac{1}{t_2 - t_1 + 1} \sum_{t=t_1}^{t_2} x_t. \tag{8}$$

The functions that we identified as being correlated with a good positioning of the onset-offset pair can be divided into four groups based on the structure of function; each group was implemented over a set of the above acoustic features:

Type 1: This type of feature functions gets as input a sequence of one of the features, $\bar{x}$, and a time t. The time t can represent the onset or the offset of the vowel. Formally, the features of this type are of the form

$$\phi(\bar{x}, t) = x_t \tag{9}$$

The set of feature functions from this type are computed for the acoustic features E_{total}, E_{low}, E_{high}, and S_{max} at the presumed vowel onset time t_b. It is also computed for the

acoustic feature D_j, $j = 1, 2, 3, 4$ for both vowel onset t_b and offset times t_e. It can be seen from Figure 3 that these acoustic features have a high value exactly at t_b or t_e or both, respectively.

Type 2: The second set of feature functions is a coarse estimation of the derivative around a time frame of interest. Given a sequence of acoustic feature values $\bar{x}$ and a specific time frame t, the feature functions of this type compute the difference in the mean of Δ frames before t and the mean of Δ frames after t. That is

$$\phi(\bar{x}, t, \Delta) = \mu(\bar{x}, t - \Delta, t - 1) \; - \; \mu(\bar{x}, t, t + \Delta - 1). \tag{10}$$

Table 1 describes all the feature functions of this type, for the values of the number of frames processed, Δ, and the acoustic features. For some acoustic features we wanted to take into account the possibility that might not be coincident with vowel boundary, but occur at an adjacent. We did that by considering the point of interest to be an offset version of the presumed onset or offset. For example, $t_b - 2$ means the feature computes averages before and after the time frame t_b with an offset of 2 frames. It can be seen from Figure 3 that indeed the acoustic features described in Table 1 have abrupt changes at the specified time frame t.

Type 3: The third type of feature functions is an extension of the second type. Rather than considering the average of an acoustic feature before and after a single time frame, we refer

Table 1: The feature functions of Type 2, $\phi(\bar{x}, t_1, \Delta)$. Values in the tables are the time frame t_1 used for each acoustic feature (columns) and for each window length (rows).

Δ	$E_{\text{short-term}}$	E_{low}	E_{high}	E_{total}	H_{wiener}	$S_{\max}$	$\hat{F}_0$	V_{RAPT}	N_{ZC}	G_{vowel}	G_{nasal}	L_{vowel}	D_1	D_2	D_3	D_4
1						t_b										
2						t_b										
3	t_b, t_e					t_b							t_b, t_e	t_b, t_e	t_b, t_e	t_b, t_e
4	t_b, t_e					t_b										
5	t_b					t_b										
6		t_b								t_b, t_e		t_b, t_e				
8		t_b, t_e	t_b, t_e	t_b, t_e	t_b, t_e		t_b, t_e	t_b, t_e	t_b, t_e	t_b, t_e	t_e	t_b, t_e				
8		$t_b - 2$	$t_b - 2$	$t_b - 2$												
10		t_b, t_e	t_b, t_e	t_b, t_e	t_b, t_e		t_b, t_e	t_b, t_e	t_b, t_e	t_b, t_e	t_e	t_b, t_e	t_b, t_e	t_b, t_e	t_b, t_e	t_b, t_e
10		$t_b - 4$	$t_b - 4$	$t_b - 4$												

to the average between the presumed onset and offset times. Feature functions of this type return two values: (i) the average between the presumed onset and offset times minus the average of Δ frames before t_b; and (ii) the average between the presumed onset and offset times minus the average of Δ frames after the offset time t_e. Those two values correspond to two elements in the vector

$$\boldsymbol{\phi}(\bar{x}, t_b, t_e, \Delta) = \begin{bmatrix} \mu(\boldsymbol{x}, t_b, t_e) - \mu(\boldsymbol{x}, t_b - \Delta, t_b - 1) \\ \mu(\boldsymbol{x}, t_b, t_e) - \mu(\boldsymbol{x}, t_e + 1, t_e + \Delta) \end{bmatrix}. \tag{11}$$

This type of feature function is computed for the acoustic features $E_{\text{short-term}}$, E_{low}, E_{high}, E_{total}, V_{RAPT}, N_{ZC}, and L_{vowel}. Again, it can be see from Figure 3 that these features have high (or low) mean values between the time frames of interest.

Type 4: The fourth type of feature functions is oblivious to the acoustic signal and returns a probability score for the presumed vowel duration, $t_e - t_b$. We have two feature functions of this type. The first one is based on the Normal distribution with parameters $\hat{\mu}$, $\hat{\sigma}^2$ that are estimated from the training data:

$$\phi(t_b, t_e) = \mathcal{N}(t_e - t_b; \hat{\mu}, \hat{\sigma}^2). \tag{12}$$

Similarly, the second feature function of this form is based on the Gamma distribution with parameters $\hat{k}$ and $\hat{\theta}$, that are estimated from the training set:

$$\phi(t_b, t_e) = \Gamma(t_e - t_b; \hat{k}, \hat{\theta}). \tag{13}$$

While the Gamma distribution is an appropriate distribution to describe the vowel duration alone, we found empirically that adding the Normal distribution improves performance.

6 DATASETS

In order to get reliable results we used three different datasets to evaluate the performance of our system. In this section we will give a short description of each dataset.

6.1 Heller and Goldrick, 2014 (HG)

This corpus (Heller and Goldrick, 2014) is drawn from a study investigating how grammatical class constraints influence the activation of phonological neighbors. The influence of neighbors on phonetic processing was indexed by vowel durations. It contains segments of acoustic signal from 64 native English speakers (55 female) aged 18-34 with no history of speech or language deficits. Participants were first familiarized to a set of pictures along with their intended labels. They were then asked to name aloud the noun depicted by a picture in two contexts: when the picture was presented alone, and when the picture occurred at the end of a non-predictive sentence. Participants were instructed to produce the name as quickly and accurately as possible. Trials with errors or disfluencies were excluded, along with two items with high error rates. In addition, data from three subjects reported in the original paper were excluded from the present analysis, as they did not consent to public use

of their data. The remaining 2395 recorded segments of acoustic signal contain one English CVC noun with vowels /i, ɛ, ae, ɑ, oʊ, u/.

6.2 Clopper and Tamati, 2014 (CT)

The second dataset (Clopper and Tamati, 2014) contains segments of acoustic signal from 20 female native English speakers aged 18-22 with no history of speech or language deficits. The participants were evenly split between two American English dialects (Northern and Midland). As part of a larger study, participants read aloud a list of 991 CVC words (total of 4049 tokens). This study focused on 39 target words (788 tokens) which did or did not have a lexical contrast between either /ɛ/ vs. /ae/ (e.g., dead-dad vs. deaf-*daff) or /ɑ/ vs. /ɔ/ (e.g., cot-caught vs. dock-*dawk). Words with a lexical contrast are referred to as *competitor* items and those without are referred to as *no competitor* items.

6.3 Hillenbrand, Getty, Clark, and Wheeler, 1995 (HGCW)

The third dataset consists of data from a laboratory study conducted by Hillenbrand et al. (1995). It contains segments of acoustic signal from 45 men, 48 women, and 46 ten-to 12-year-old children (27 boys and 19 girls). 87% of the participants were raised in Michigan, primarily the southeastern and southwestern parts of the state. The audio recordings contain 12 different vowels (/i, ɪ, ɛ, ae, ɑ, ɔ, ʊ, u, ʌ, ɜ, e, o/) from the words: heed, hid, hayed, head,

had, hod, hawed, hoed, hood, who'd, hud, heard, hoyed, hide, hewed, and how'd.

7 METHODS

Code implementing this algorithm is publicly available at `https://github.com/adiyoss/AutoVowelDuration`. Segments of acoustic signal for the HG dataset along with algorithmic and manual annotations, are available in the Online Speech/Corpora Archive and Analysis Resource (`https://oscaar.ci.northwestern.edu/`; dataset 'Within Category Neighborhood Density'). For the HGCW dataset, segments of acoustic signal and manual annotations are available at: `http://homepages.wmich.edu/~hillenbr/voweldata.html`. Do we need to include Clopper's data too?

7.1 DCM

The direct cost minimization (DCM) algorithm proposed above was trained and tested on both HG and CT datasets. For the CT corpus, the training set was the unused 4049 tokens, and the test set was the same 777 tokens reported in Clopper and Tamati (2014). The model parameters were $\eta = 0.1$ and $\epsilon = -1.36$. We used $\tau_b = 1$ frame and $\tau_e = 2$ frames for the loss during training. The initial weight vector was set to be the averaged weight vector from Passive-Aggressive (PA) algorithm (Crammer et al., 2006) with $C = 0.5$ and 100 epochs. When trained and tested on the HG corpus, we used 10-fold cross validation (the reported

results are the average error over all 10 folds) with the same settings and parameters as described above.

In order to better comprehend the influence of the phoneme classifier, the only language dependent feature in our system, we train and test the direct cost algorithm without the acoustic features related to the phoneme classifier G_{vowel}, G_{nasal}, or L_{vowel}, and their corresponding feature functions. In the case when the phoneme classifier was used, we trained multiclass PA as described in (Dekel et al., 2004) on the TIMIT corpus of read speech.

7.2 HMM

To validate the effectiveness of the proposed approach, we compared it the most common approach currently used in automatic phonetic measurement of speech: forced alignment. This is an algorithm which, given a speech utterance and its phonetic content, finds the start time of each phoneme in the speech utterance. Often the orthographic content of the speech utterance is given, and it is converted to its phonetic content using a lexicon. This procedure may not be accurate as often the surface pronunciation uttered in spontaneous speech is not the same as the canonical pronunciation that appears in the lexicon.

Conventionally, the forced alignment is implemented by forcing the decoder of an HMM phoneme recognizer to pass through the states corresponding to the given phoneme sequence. So far, automatic vowel extraction has been done using such forced alignment procedures

(Reddy and Stanford, 2015). While there are several open source implementations of HMMs, all the publicly available forced aligner packages (Goldman, 2011; Gorman et al., 2011; Rosenfelder et al., 2014; Yuan and Liberman, 2008) are based on the HTK toolkit (Young and Young, 1994). Since this is the case, we chose to compare our results with the most recent one, namely the *FAVE aligner* (Rosenfelder et al., 2014), based on the *Penn Phonetics Lab Forced Aligner (P2FA)* (Yuan and Liberman, 2008). In our analyses below, we refer to this system as the HMM aligner.

In contrast to the proposed algorithm, the parameters of this system are not acquired via a training procedure but are rather set by the designers. Note as well that in contrast to the DCM the forced aligner requires a phonetic or orthographic transcription as input.

8 RESULTS: MEASUREMENT DEVIATION

The difference between the automatic and manual measurements of vowel duration are given in Table 2, where the evaluation metric is the cost in Eq. (2) with both τ_b and τ_e equals to 0, and in Table 3, where the error is given in terms of the percentage of predictions that do not fall within the boundaries of 20 ms from the manual onset and 50 ms from the manual offset.

	DCM		DCM (no classifier)		HMM	
	onset	offset	onset	offset	onset	offset
HG	5.21	22.80	5.84	28.98	16.67	24.72
CT	9.42	16.76	9.24	23.18	35.90	30.61

Table 2: *Results of DLM (with and without phoneme classifier) and HMM relative to manual annotation. Average deviation of onset and offset [in msec].*

	DCM		DCM (no classifier)		HMM	
	onset	offset	onset	offset	onset	offset
HG	6.15%	13.15%	8.05%	18.44%	31.90%	10.94%
CT	9.46%	8.31%	9.59%	9.34%	41.50%	13.14%

Table 3: *The percentage of predictions that do not fall within the boundaries of 20 ms at the onset and 50 ms at the offset from the manual annotation.*

8.1 Mismatched training and test datasets

We now test how the algorithm's performance varies when different training and testing corpora are used. To compare to the base experiments (where the training set and test set were drawn from the same corpus), we conduct additional experiments, corresponding to training on HG and test on CT and vice versa. We also present the result of our algorithm

trained on CT and tested on a third dataset, HGCW.

AS before, the difference between the automatic and manual measurements of vowel duration are given in Table 4, and the error in terms of the percentage of predictions that do not fall within the boundaries of 20 ms from the manual onset and 50 ms from the manual offset is given in Table 5.

	DCM		DCM (no classifier)		HMM	
	onset	offset	onset	offset	onset	offset
HG (*CT* model)	14.44	38.66	14.92	43.95	16.67	24.72
CT (*HG* model)	9.84	30.85	10.17	29.94	35.90	30.61
HGCW (*CT* model)	12.91	9.28	15.86	23.89	19.95	27.30

Table 4: *Results of DLM (with and without phoneme classifier) under mismatched training and test datasets. Average deviation of onset and offset [in msec].*

8.2 Measurement Correlation

Another measure of annotator agreement conventionally reported in phonetic studies is the correlation between measurements. This is typically done by assigning a random subset of the data to a second annotators. The results show that the DCM exceeds the HMM, showing correlations comparable to that of human annotators. For the HG dataset, the

	DCM		DCM (no classifier)		HMM	
	onset	offset	onset	offset	onset	offset
HG (*CT* model)	11.55%	23.99%	12.25%	28.80%	31.90%	10.94%
CT (*HG* model)	8.36%	14.92%	8.36%	10.94%	41.50%	13.14%
HGCW (*CT* model)	19.18%	2.16%	27.58%	6.54%	28.54%	11.69%

Table 5: *The percentage of predictions that do not fall within the boundaries of 20 ms at the onset and 50 ms at the offset from the manual annotation. Mismatched training and test datasets.*

second annotator's correlation with the original annotator was r(627) = 0.84. For the same subset of the data, the correlations between each algorithm and the original manual annotator are DCM=0.79, DCM (no classifier)=0.64, and HMM=0.73. For the CT dataset, the second annotator's correlation with the original annotator was r(398) = 0.95. For the same subset of the data, each algorithm's correlations are: DCM=0.93, DCM (no classifier)=0.93, and HMM=0.57.

8.3 Discussion

Analysis of measurement deviation suggest our model outperforms the HMM-forced alignment algorithm. Incorporating the phoneme classifier improved performance, but even with-

out the classifier the DCM typically outperforms the HMM.

9 RESULTS: REPRODUCING REGRESSION MODEL FINDINGS

Empirical studies of speech and language processing use acoustic properties such as vowel duration as behavioral measures of the effects of various types of variables that influence speech and language. Canonical examples of variables of study include the phonetic context of vowels (e.g. preceding voiced vs. voiceless stops), properties of speakers who those produce vowels (e.g. native vs. non-native speakers) and the linguistic (lexical, syntactic, etc.) context in which the vowels appear (e.g. predictable vs. unpredictable). The effects of these variables on acoustic properties are typically examined using inferential statistical models.

The second evaluation metric of our algorithms therefore examined the similarity of inferential statistical model fits based on measurements generated by algorithms vs. manual measurements. Out of three datasets, only HG constructed models based on duration data; our analysis therefore focused on this dataset. (In supplemental materials, we examine the CT dataset, which built inferential models based on spectral measures, derived via an additional algorithm.)

Heller and Goldrick (2014) examined whether processes involved in sentence planning influence the processing of sound. Speakers named pictures in a context that strongly emphasized sentence planning (following a sentence fragment) as well as a context that did not

require substantial planning (producing the picture name in isolation). The order of these contexts was counterbalanced across speakers. To index effects of sound structure processing, this study also manipulated the number of words phonologically similar to the target that share its grammatical category (category-specific lexical density).

To analyze the effect of this variables, HG used linear mixed effects regression models (Baayen et al., 2008), an approach that has become dominant in the analysis of speech and psycholinguistic data. These regression models predict dependent measures based on a linear combination of predictors, including both fixed and randomly distributed predictors capturing variation in effects by both participants and items. This allows researchers to examine the effects of interest while controlling for other properties of the words and speakers.

Analysis of the full data set showed that lexical density and vowel category had no effect on vowel durations (Heller and Goldrick, 2014, 2015); however, these effects were not stable when the subset of data used here was examined. Therefore, the models used here were simplified from those reported in the original paper, including two contrast-coded fixed effect factors: production context (isolation vs. sentence) and block (first vs. second). To control for contributions from the random sample of participants and items used in this experiment (compared to all English speakers and all words of English), we included two sets of random effects. Random intercepts for both speakers and words were included, along with uncorrelated slopes for context by both speaker and word.

9.1 Significance of fixed effects in models of the HG dataset

We first examine the primary interest of many phonetic studies–the binary distinction between significant vs. insignificant effects of fixed-effect predictors (e.g., the effect of experimental condition). To control for skew, vowel durations were log-transformed prior to analysis. Given that outliers can have an outsized influence on parameter estimates, all regressions models were refit after excluding observations with standardized residuals exceeding 2.5 (Baayen, 2008). The significance of fixed-effects predictors was assessed by using the likelihood ratio test to compare models with and without the predictor of interest (Barr et al., 2013).Table 6 compares the estimates for the two fixed effects parameters for the manual model and each algorithmic model. Although there was some variation in the model estimates of the fixed effects for each algorithm compared to the manual annotations, each algorithm recovered the overall pattern of a significant effect of context but not block.

9.1.1 Comparison of predictions of models of the HG dataset

An alternative, more global, assessment of model similarity is to compare model predictions. This was assessed by leave-one-out validation. For each observation, we excluded it from the dataset and re-fit the regression to the remaining observations. After residual-based outlier trimming (and re-fitting), we examined the predictions of this re-fitted model for the excluded observation. Figure 4 shows distributions of the deviation of each algorithm's

	Context	t	Block	t
Manual	**-0.057 (0.017)**	**-3.29**	0.012 (0.016)	0.72
DCM	**-0.072 (0.019)**	**-3.82**	0.016 (0.018)	0.91
DCM (no classifier)	**-0.059 (0.018)**	**-3.34**	0.015 (0.017)	0.90
HMM	**-0.058 (0.018)**	**-3.13**	0.008 (0.014)	0.55

Table 6: Estimates and t-values for fixed effects in regression model, Heller & Goldrick dataset (standard error of estimate in parentheses). Significant effects ($p <= 0.05$, as assessed by likelihood ratio tests) are bolded.

predicted fit to the predicted model fit to the manual data.

The DCM algorithm out-performed the other algorithms. The mean squared error relative to the predictions of the model fit to the manually annotated data was lowest for the DCM algorithm (0.479 msec^2), higher for HMM (0.835 msec^2), and larger still for DCM (no classifier) (0.935 msec^2). Bootstrap confidence intervals of the differences across algorithms showed that DCM outperformed the other two algorithms (95% CI of differences from HMM: [0.00039, 0.00043] msec^2; DCM (no classifier): [0.00037, 0.00055] msec^2). The HMM algorithm did not significantly differ from DCM (no classifier) (95% CI [-0.000200, 0.000002] msec^2).

Parallel to the analysis of measurement deviation, these analyses suggest our model outperforms the HMM-forced aligner. Incorporating the phoneme classifier improved perfor-

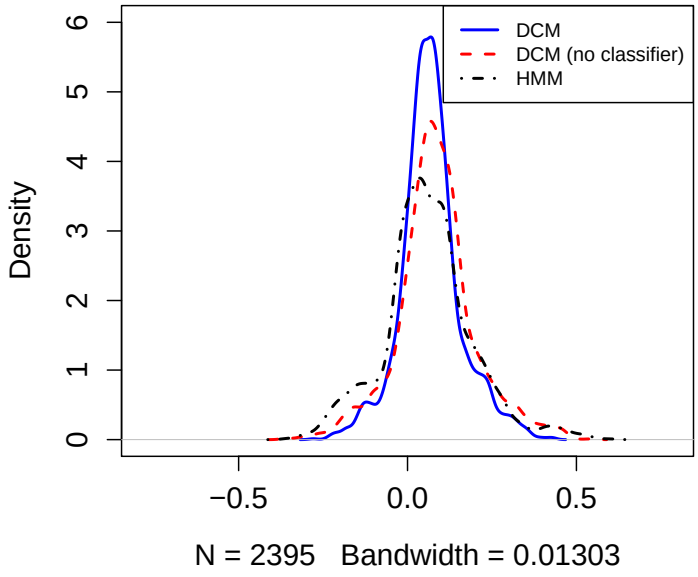

Figure 4: Comparisons of leave-one-out model predictions of vowel durations for the Heller & Goldrick (2014) dataset. Deviance of each algorithmic method as compared to the manual predictions (manual-algorithmic) are plotted as individual lines.

mance, but even without the classifier the performance of the DCM meets or exceeds that of the HMM.

10 GENERAL DISCUSSION

We presented an algorithm for automatically estimating the durations of vowels based on the structured prediction framework, relying on a set of acoustic features and feature functions finely tuned to reflect the properties of vowel acoustics relevant to vowel segmentation. The DCM algorithm, when including a phoneme classifier, clearly succeeds at matching manual measurements of vowel duration. With respect to both measurement deviation and reproduction of regression model results, the DCM algorithm out-performs or matches a commonly-used forced alignment system while not requiring a phonetic transcription. This approach can allow laboratory experiments to address much larger samples of data in a way that is replicable and reliable.

Having achieved some success with monosyllabic, laboratory stimuli, future development of this algorithm should extend this approach to more naturalistic production. In current work, we are extending this approach to vowel durations in multisyllabic words. We believe that moving outside the laboratory will be facilitated by the structure of our approach; in contrast with existing systems, our algorithm has the additional benefit of not requiring a transcript of the desired speech prior to analysis. Extending this capability will

support the analysis of more naturalistic, connected speech, including speech styles or dialects that may be difficult to robustly sample in a laboratory context (Labov, 1972; Rischel, 1992).

In terms of algorithmic point of view, future work will be focused on combining new advances in sequence deep learning Elman (1990); Graves et al. (2013); Graves and Jaitly (2014) to our current structured prediction scheme.

Acknowledgements

Research supported by NIH grant 1R21HD077140 and NSF grant BCS1056409.

REFERENCES

Y. Adi, J. Keshet, and M. Goldrick. Vowel duration measurement using deep neural networks. In *Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing*, 2015.

R. H. Baayen. *Analyzing linguistic data: A practical introduction to statistics using R.* Cambridge University Press, 2008.

R. H. Baayen, D. J. Davidson, and D. M. Bates. Mixed-effects modeling with crossed random effects for subjects and items. *Journal of Memory and Language*, 59(4):390–412, 2008.

D. J. Barr, R. Levy, C. Scheepers, and H. J. Tily. Random effects structure for

confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, 68 (3):255–278, 2013.

C. G. Clopper and T. N. Tamati. Effects of local lexical competition and regional dialect on vowel production. *Journal of the Acoustical Society of America*, 136(1):1–4, 2014.

K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online passive aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585, 2006.

O. Dekel, J. Keshet, and Y. Singer. An online algorithm for hierarchical phoneme classification. In *Workshop on Multimodal Interaction and Related Machine Learning Algorithms; Lecture Notes in Computer Science*, volume 3361/2005, pages 146–159. Springer-Verlag, 2004.

J. L. Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

K. Evanini. *The permeability of dialect boundaries: A case study of the region surrounding Erie*. PhD thesis, University of Pennsylvania, 2009.

J.-P. Goldman. Easyalign: an automatic phonetic alignment tool under Praat. In *Proceeding of Interspeech*, 2011.

K. Gorman, J. Howell, and M. Wagner. Prosodylab-aligner: A tool for forced alignment of laboratory speech. *Canadian Acoustics*, 39(3):192–193, 2011.

A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1764–1772, 2014.

A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 6645–6649. IEEE, 2013.

J. R. Heller and M. Goldrick. Grammatical constraints on phonological encoding in speech production. *Psychonomic Bulletin & Review*, 21(6):1576–1582, 2014.

J. R. Heller and M. Goldrick. Erratum to: 'grammatical constraints on phonological encoding in speech production'. *Psychonomic Bulletin & Review*, 22(5):1475, 2015.

J. Hillenbrand, L. A. Getty, M. J. Clark, and K. Wheeler. Acoustic characteristics of american english vowels. *The Journal of the Acoustical society of America*, 97(5): 3099–3111, 1995.

J. Keshet. Optimizing the measure of performance in structured prediction. In S. Nowozin, P. V. Gehler, J. Jancsary, and C. H. Lampert, editors, *Advanced Structured Prediction*. The MIT Press, 2014.

J. Keshet, S. Shalev-Shwartz, Y. Singer, and D. Chazan. A large margin algorithm for

speech and audio segmentation. *IEEE Trans. on Audio, Speech and Language Processing*, Nov 2007.

W. Labov. *Sociolinguistic patterns*. Number 4. University of Pennsylvania Press, 1972.

D. McAllester, T. Hazan, and J. Keshet. Direct loss minimization for structured prediction. In *Advances in Neural Information Processing Systems*, pages 1594–1602, 2010.

G. E. Peterson and I. Lehiste. Duration of syllable nuclei in english. *The Journal of the Acoustical Society of America*, 32(6):693–703, 1960.

S. Reddy and J. N. Stanford. Toward completely automated vowel extraction: Introducing DARLA. *Linguistics Vanguard*, 2015.

J. Rischel. Formal linguistics and real speech. *Speech Communication*, 11(4):379–392, 1992.

I. Rosenfelder, J. Fruehwald, K. Evanini, S. Seyfarth, K. Gorman, H. Prichard, and J. Yuan. Fave (forced alignment and vowel extraction). Program suite v1.2.2 10.5281/zenodo.22281, 2014.

F. Sha, J. A. Burgoyne, and L. K. Saul. Multiband statistical learning for f0 estimation in speech. In *Proceeding of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 5, pages 661–664, 2004.

M. Sonderegger and J. Keshet. Automatic discriminative measurement of voice onset time. *Journal of the Acoustical Society of America*, pages 3965–3979, 2012.

D. Talkin. A robust algorithm for pitch tracking. *Speech coding and synthesis*, 495:518, 1995.

S. Young and S. Young. The htk hidden markov model toolkit: Design and philosophy. *Entropic Cambridge Research Laboratory, Ltd*, 2:2–44, 1994.

J. Yuan and M. Liberman. Speaker identification on the SCOTUS corpus. *Journal of the Acoustical Society of America*, 123(5):3878, 2008.

J. Yuan, N. Ryant, M. Liberman, A. Stolcke, V. Mitra, and W. Wang. Automatic phonetic segmentation using boundary models. In *INTERSPEECH*, pages 2306–2310. Citeseer, 2013.