Automatic measurement of voice onset time using discriminative structured prediction[a),b)]

Morgan Sonderegger[c)]
Department of Computer Science and Department of Linguistics, University of Chicago, 1100 E 58th Street, Chicago, Illinois 60637

Joseph Keshet
Toyota Technological Institute at Chicago, 6045 S Kenwood Avenue, Chicago, Illinois 60637

(Received 12 March 2012; revised 19 June 2012; accepted 5 October 2012)

A discriminative large-margin algorithm for automatic measurement of voice onset time (VOT) is described, considered as a case of predicting structured output from speech. Manually labeled data are used to train a function that takes as input a speech segment of an arbitrary length containing a voiceless stop, and outputs its VOT. The function is explicitly trained to minimize the difference between predicted and manually measured VOT; it operates on a set of acoustic feature functions designed based on spectral and temporal cues used by human VOT annotators. The algorithm is applied to initial voiceless stops from four corpora, representing different types of speech. Using several evaluation methods, the algorithm's performance is near human intertranscriber reliability, and compares favorably with previous work. Furthermore, the algorithm's performance is minimally affected by training and testing on different corpora, and remains essentially constant as the amount of training data is reduced to 50–250 manually labeled examples, demonstrating the method's practical applicability to new datasets. *© 2012 Acoustical Society of America.* [http://dx.doi.org/10.1121/1.4763995]

PACS number(s): 43.72.Ar, 43.70.Jt, 43.70.Fq [SSN] Pages: 3965–3979

I. INTRODUCTION

Huge corpora of speech, both from laboratory and naturalistic settings, are becoming increasingly available and easy to construct, and promise to change the questions researchers can ask about human speech production. However, this promise depends on the development of accurate algorithms to quicken or replace manual measurement, which becomes infeasible for large corpora. With a few important exceptions (such as pitch and vowel formants), such algorithms do not currently exist for most quantities which are widely measured in phonetic research. This paper describes an automatic measurement algorithm for perhaps the most widely measured consonantal variable, voice onset time (VOT). VOT, the time difference between the onset of a stop consonant's burst and the onset of voicing in the following phone, is an important perceptual cue to stop voicing and place. It is measured in many clinical and research studies every year, requiring many transcriber hours; for example when studying how communication disorders affect speech (Auzou *et al.*, 2000) or how languages differ in the phonetic cues to stop contrasts (Lisker and Abramson, 1964; Cho and Ladefoged, 1999).

There have been a number of previous studies proposing algorithms for automatic VOT measurement. Previous work has used automatic measurements for speech recognition tasks (Niyogi and Ramesh, 1998, 2003; Ali, 1999; Stouten and van Hamme, 2009), phonetic measurement (Fowler *et al.*, 2008; Tauberer, 2010), and accented speech detection (Kazemzadeh *et al.*, 2006; Hansen *et al.*, 2010). Some studies, like ours, focus largely on the problem of VOT measurement itself, and evaluate the proposed algorithm by comparing automatic and manual measurements (Stouten and van Hamme, 2009; Yao, 2009; Hansen *et al.*, 2010; Lin and Wang, 2011). Our approach differs from all previous studies except one (Lin and Wang, 2011) in an important aspect. Instead of using a set of customized rules to estimate VOT, our system learns to estimate VOT from training data.

To replace manual measurement, we believe that an automatic VOT measurement algorithm should meet three criteria. Both because the burst and voicing onsets are often highly transient, and because the effects of interest (e.g., VOT difference between two conditions) in studies using VOT measurements are often very small, the algorithm should have high *accuracy* by the chosen measure of performance. The cues to the burst and voicing onset locations vary depending on many factors (speaking style, speaker's native language), and different labs have slightly different VOT measurement criteria. To account for such variation in the mapping between spectral/temporal cues and labeled VOT boundaries, the algorithm should be *trainable*: it should learn to measure VOT based on labeled data, and should perform well on diverse datasets. To meet the goal of replacing manual measurement, it should also be *adaptable* to a new dataset with little effort (i.e., training data).

[a)] We dedicate this article to the memory of Partha Niyogi.

[b)] An earlier version of some work presented in this paper was published as "Automatic discriminative measurement of voice onset time" in *Proceedings of Interspeech-2010* (Sonderegger and Keshet, 2010). The current paper reports a much larger set of experiments, which are performed on a larger number of datasets. In addition, all non-experimental sections are either new to this paper or have been substantially expanded and revised.

[c)] Author to whom correspondence should be addressed. Current address: Department of Linguistics, McGill University, 1085 Dr. Penfield Avenue, Montreal, Quebec H2S 2C2, Canada. Electronic mail: morgan.sonderegger@mcgill.ca

This paper proposes a supervised learning algorithm meeting all three criteria. The algorithm is trained on a set of manually labeled examples, each consisting of a speech segment of an arbitrary length containing a stop consonant, and a label indicating the burst onset and the voicing onset, which we denote an *onset pair*. At test time the algorithm receives as input a speech segment containing a stop consonant, and outputs an onset pair and its corresponding VOT. The goal of the algorithm is to predict VOT as accurately as possible on unseen data.

Our algorithm belongs to the family of discriminative large-margin learning algorithms. A well-known member of this family is the support vector machine (SVM). The classical SVM algorithm assumes a simple binary classification task, where each input is of fixed length. The task of predicting VOT is more complex: the input is a speech segment of arbitrary length, and the goal is to predict the time between two acoustic events in the signal. Our algorithm is based on recent advances in kernel machines and large margin classifiers for structured prediction (Taskar *et al.*, 2003; Tsochantaridis *et al.*, 2004; Shalev-Shwartz *et al.*, 2004). It maps the speech segment along with the target onset pair into a vector space endowed with an inner product. The vector space contains all possible onset pairs, and during training the algorithm tries to find a linear classifier which separates the target onset pair, as well as all "nearby" onset pairs (in terms of the cost function), from all other possible onset pairs in this vector space. At test time, the algorithm receives unseen speech segments. Each segment is mapped to the same vector space, and the most probable onset pair (and hence VOT) is predicted.

For this method to work and achieve high accuracy, the feature set must induce a vector space in which the target onset pair is both distinguishable and separable from other onset pairs. We achieve this by manually crafting a set of features which are informative about the precise locations of the burst and voicing onsets, and which tend to take on higher values for target onset pairs than for other onset pairs. The features leverage knowledge about how humans annotate VOT: using a variety of cues based on the spectrum, the waveform, and the output of speech processing algorithms (such as pitch trackers). We note that the feature sets typically used in speech recognition (e.g., MFCCs, PLPs) are not adequate for VOT measurement, since their time resolution is too coarse to accurately detect highly transient events such as burst onsets.

Another factor that controls the accuracy of the algorithm is the cost function used to evaluate how good a predicted VOT is, relative to its target value. Discriminative learning algorithms aim to minimize some measure of performance or cost function. The classic SVM, for example, is designed to minimize the zero-one loss function during training (i.e., the number of incorrect classifications). Our algorithm aims to minimize a special cost function, which is low if the predicted VOT is close to the manually measured VOT and high otherwise. The function also does not penalize small differences between predicted and labeled VOT values during training, taking into account the fact that some measurement inconsistency (within or across annotators) is expected.

We evaluate our algorithm's accuracy in experiments on four datasets, using several methods to evaluate the algorithm's predictions relative to manual measurements. The datasets range across very different types of speech, testing the algorithm's applicability in different settings. To test the algorithm's adaptability to novel datasets where little or no labeled data is available, we perform experiments testing the algorithm's robustness to reducing the amount of training data, or training and testing on different datasets.

The paper is structured as follows. We first formally describe the problem of VOT measurement (Sec. II), and describe our algorithm and the feature maps it takes as input (Sec. III). We then turn to our experiments: first the datasets and evaluation methods used (Sec. IV), then experiments testing our method's accuracy (Sec. V), and its robustness to decreasing the amount of training data and to mismatched train/test conditions (Sec. VI). We further evaluate our system by comparison with previous work (Sec. VII), and by comparing regression models of variation in VOT induced by automatic and manual measurements (Sec. VIII). In Sec. IX we sum up, and discuss directions for future work.

II. PROBLEM SETTING

In the problem of VOT measurement, we are given a segment of speech, containing a stop consonant (plosive) followed by a voiced phone. The goal is to predict the time difference between the onset of the stop burst and the onset of voicing in the following phone. The speech segment can be of arbitrary length, but should include at most one burst, and its beginning need not be precisely synchronized with the stop's burst or closure; it is only required that the segment begins before the burst onset.

Note that an important prerequisite to VOT measurement is *finding* such segments, i.e., determining where to begin looking for each burst. We view this as a separate problem, and focus on the problem of VOT measurement itself in this paper.

Throughout the paper we write scalars using lower case latin letters (x), and vectors using bold face letters ($\mathbf{x}$). A sequence of elements is denoted with a bar ($\bar{\mathbf{x}}$) and its length is written $|\bar{\mathbf{x}}|$.

We represent each speech segment by a sequence of acoustic feature vectors $\bar{\mathbf{x}} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$, where each $\mathbf{x}_t$ $(1 \leq t \leq T)$ is a D-dimensional vector. We denote the domain of the feature vectors by $\mathcal{X} \subset \mathbb{R}^D$. (The precise set of features used is described below.) Because different segments have different lengths, T is not fixed; we denote by $\mathcal{X}^*$ the set of all finite-length sequences over $\mathcal{X}$. Each segment is associated with an *onset pair*: $t_b \in \mathcal{T}$, the onset of the burst (in frames), and $t_v \in \mathcal{T}$, the onset of voicing of the following phone, where $\mathcal{T} = \{1, \ldots, T\}$. Given the speech segment $\bar{\mathbf{x}}$, our goal is to predict $t_v - t_b$: the length of time that passes between the beginning of the stop consonant's burst and the beginning of voicing in the following voiced phone. We assume here that $t_b < t_v$, and leave the case of "prevoiced" stops (where $t_b > t_v$), to future work. Our goal is to learn a function f from the domain of all speech segments $\mathcal{X}^*$ to the domain of all onset pairs $\mathcal{T}^2$.

III. LEARNING APPARATUS

In this section we describe a discriminative supervised learning approach for learning a function f from a training set of examples. Each example consists of a speech segment $\bar{\mathbf{x}}$ and a label (t_b, t_v). Our goal is to find a function which performs well on the training set, as well as on unseen examples. The performance of f is measured by the percentage of predicted VOT values, $t_v - t_b$, which are within a time threshold of the manually labeled values.

Formally, given a speech segment $\bar{\mathbf{x}}$, let $(\hat{t}_b, \hat{t}_v) = f(\bar{\mathbf{x}})$ be the predicted onset pair. The *cost* associated with predicting $(\hat{t}_b, \hat{t}_v)$ when the manually labeled pair is (t_b, t_v) is measured by a cost function, γ: $\mathcal{T}^2 \times \mathcal{T}^2 \rightarrow \mathbb{R}$. The function used in our experiments is of the form

$$\gamma((t_b, t_v), (\hat{t}_b, \hat{t}_v)) = \max\{|(\hat{t}_v - \hat{t}_b) - (t_v - t_b)| - \epsilon, 0\}, \tag{1}$$

that is, only differences between the predicted VOT and the manually labeled VOT that are greater than a threshold ϵ, are penalized. This cost function takes into account that manual measurements are not exact, and ϵ can be adjusted according to the level of measurement uncertainty in a dataset. For brevity, we denote $\gamma = \gamma\left((t_b, t_v)\,,\ (\hat{t}_b, \hat{t}_v)\right)$.

We assume that the training examples are drawn from $\mathcal{Q}$, a fixed (but unknown) distribution over the domain of the examples, $\mathcal{X}^* \times \mathcal{T}^2$. The goal of training is to find the f that minimizes the expected cost between predicted and manually labeled VOT on examples from $\mathcal{Q}$, where the expectation is taken with respect to this distribution:

$$\mathbb{E}_{(\mathbf{x}, t_b, t_v) \sim \mathcal{Q}}[\gamma((t_b, t_v), f(\bar{\mathbf{x}}))].$$

Unfortunately, because we do not know $\mathcal{Q}$, we cannot simply compute this expectation. However, it still turns out to be possible to find f under lenient assumptions. We assume that our training examples are identically and independently distributed (i.i.d.) according to the distribution $\mathcal{Q}$, and that f is of a specific parameterized form. Below, we explain how to use the training set in order to find parameters of f which achieve a small cost on the training set, and a small cost on unseen examples with high probability as well.

We first describe the specific form used for the function f. Following the structured prediction scheme (Taskar *et al.*, 2003; Tsochantaridis *et al.*, 2004), f is constructed from a predefined set of N *feature maps*, $\{\phi_j\}_{j=1}^N$, each a function of the form $\phi_j : \mathcal{X}^* \times \mathcal{T}^2 \rightarrow \mathbb{R}$. That is, each feature map takes a speech segment $\bar{\mathbf{x}}$ and a proposed onset pair (t_b, t_v), and returns a scalar which, intuitively, should be higher if the onset pair makes sense given the speech segment, and should be lower if it does not. Each feature map can be thought of as an estimation of the probability of the onset pair given the speech segment (although the feature map need not actually be a proper probability distribution). For example, one feature map we use is the average energy of $\bar{\mathbf{x}}$ over frames in t_b to t_v, minus the average energy over frames in 1 to t_b. This feature map is expected to be high if t_b and t_v are located at the beginning and end of a stop burst following a closure, and low otherwise. Other feature maps might target the proper location of t_v or t_b (individually), or target VOT values $(t_v - t_b)$ within a particular range. Note that the *features*, which the sequence $\bar{\mathbf{x}}$ is composed of, are oblivious to the locations of t_b and t_v, whereas the *feature maps* are specifically tailored to handle them.

Our VOT prediction function f is a linear function of the feature maps, where each feature map ϕ_j is scaled by a weight w_j. Linearity is not a very strong restriction, since the feature maps are arbitrary (so a nonlinear dependency could be included as a further feature map). The overall score of an onset pair (t_b, t_v) is

$$\sum_{j=1}^{N} w_j \phi_j(\bar{\mathbf{x}}, t_b, t_v) = \mathbf{w} \cdot \phi(\bar{\mathbf{x}}, t_b, t_v),$$

where we use vector notation for the feature maps, $\phi = (\phi_1, \ldots, \phi_N)$, and for the weights $\mathbf{w} = (w_1, \ldots, w_N)$. Given $\bar{\mathbf{x}}$, f returns the onset pair which maximizes the overall score:

$$f(\bar{\mathbf{x}}) = \arg\max_{(t_b, t_v)} \ \mathbf{w} \cdot \phi(\bar{\mathbf{x}}, t_b, t_v). \tag{2}$$

In words, f gets as input a speech segment $\bar{\mathbf{x}}$ composed of a sequence of acoustic features, and returns a predicted onset pair by maximizing a weighted sum of the scores returned by each feature map ϕ_j.

We now describe the set of feature maps used (Sec. III A), then turn to how $\mathbf{w}$ is estimated from a training set of examples, so as to minimize the cost function defined in Eq. (1) (Sec. III B).

A. Features and feature maps

Consider the speech segment $\bar{\mathbf{x}} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$ consisting of T frames, where each acoustic feature vector $\mathbf{x}_t$ consists of D features. We extract 7 $(D = 7)$ acoustic features every 1 ms. The first 4 features refer to a short-time Fourier transform (STFT) taken with a 5 ms Hamming window: the log of the total spectral energy, E_{total}; the log of the energy between 50 and 1000 Hz, E_{low}; the log of the energy above 3000 Hz, E_{high}; and the *Wiener entropy*, H_{wiener}, a measure of spectral flatness:

$$H_{\text{wiener}}(t) = \log \int |P(f, t)|^2 df - \int \log |P(f, t)|^2 df,$$

where $P(f, t)$ is the STFT of the signal at frequency f and time t. The high frame rate and small window size are used for fine time resolution, because the burst and voicing onsets are highly transient events.

The fifth feature, $P_{\max}$, is extracted from the signal itself: the maximum of the power spectrum calculated in a region from 6 ms before to 18 ms after the frame center. The sixth feature is the 0/1 output of a voicing detector based on the RAPT pitch tracker (Talkin, 1995), smoothed with a 5 ms Hamming window. The seventh feature is the number of zero crossings in a 5 ms window around the frame center. Figure 1 shows the trajectories of the 7 features for one speech segment (the word "can't").

Before presenting the feature maps, we introduce notation for *local differences*. Let x^d be the dth acoustic feature (of an arbitrary speech segment). $\Delta_t^s(x^d)$, the local difference of resolution s applied to the acoustic feature x^d, is defined

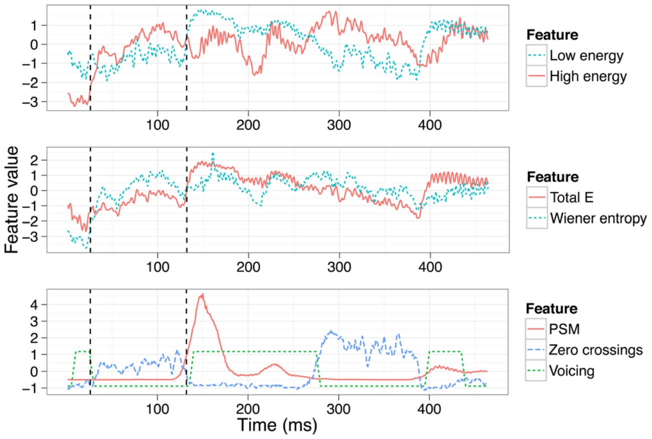

FIG. 1. (Color online) Values of the seven acoustic features for an example speech segment (the word "can't"). Vertical dashed lines show the burst and voicing onsets. PSM stands for power spectrum maximum.

as the difference between the mean of x^d over frames $\{t,\ldots,\min(t+s, T)\}$ and the mean of x^d over frames $\{\max(t-s, 0),\ldots, t\}$. This quantity provides a local approximation of the derivative of x^d at frame t, with resolution parametrized by s.

We now turn to the feature maps. For each example $(\bar{\mathbf{x}}, t_b, t_v)$, 63 feature maps ($N=63$) were calculated. As described above, each feature map describes a scalar quantity which should be high for an onset pair which makes sense given the speech segment, and low otherwise. To ensure comparability of the values of feature maps across examples, each feature map was z-scored within each example.

The feature maps are summarized in Table I, where they are split into 7 types. We describe the intuition behind each type in turn.

1. *Type 1*

We expect the correct t_b to occur at points of rapid increase in certain features, such as E_{high}, indicating the onset of turbulent airflow; at these points the corresponding local difference features (denoted by Δ in Table I) spike. In the example (Fig. 1), E_{high} and H_{wiener} rapidly increase at the correct t_b. The inclusion of the values of some features (denoted by F in Table I) at t_b helps rule out locations where a feature rapidly changes, but already has a high value.

2. *Type 2*

Similar to Type 1, but for features expected to change rapidly at voicing onset. In the example, all features change rapidly near the correct t_v.

3. *Type 3*

We expect $P_{\max}$ to not change during the burst (where there is no periodicity); hence the mean and maximum of its local difference over (t_b, t_v) should be low, as is the case in the example.

4. *Type 4*

Similar to Type 3, but taking into account that periodicity can begin towards the end of the burst; hence the mean and maximum are calculated over (t_b, t_v-10).

5. *Type 5*

Features indicating an aperiodic spectrum (E_{high}, H_{wiener}) should be much greater during the burst than before the burst. Hence, the difference between their mean in (t_b, t_v) and in $(1, t_b)$ should be large, as is the case in the example.

6. *Types 6, 7*

Features indicating a noisy spectrum (E_{high}, H_{wiener}) should be uniformly low before the burst begins, and hence should have small mean and max values over $(1, t_b-5)$. (An endpoint slightly before t_b is used because these features may already be rising by the burst onset.) We also expect there to be little voicing in this interval, and hence the voicing feature should have a low mean value.

The feature maps were chosen based on manual inspection of trajectories of the 7 acoustic features for labeled examples. They reflect knowledge about the effects of stop bursts and voicing on the spectrum, as well as knowledge about the process of VOT measurement itself. For example, feature types 3–4 take into account that the point labeled as the voicing onset can be somewhat later than the first signs of periodicity (Type 4), or synchronous with them (Type 3). (On the relationship between voicing onset's true location and common criteria for annotating it based on the speech signal, see Francis *et al.* (2003).)

TABLE I. Summary of the 63 feature maps. The feature maps fall into several types described in the text, each of which is evaluated for some of the 7 acoustic features (one per column). F in row i and column corresponding to feature x_j indicates that there is a feature map of type i for feature x_j; Δ indicates there are three feature maps of type i for the local difference of feature x_j, evaluated at $s=5$, 10, 15. For example, the F, Δ in row 2 in the E_{low} column denotes four feature maps: $E_{\text{low}}(t_v)$, $\Delta^5_{t_v}(E_{\text{low}})$, $\Delta^{10}_{t_v}(E_{\text{low}})$, and $\Delta^{15}_{t_v}(E_{\text{low}})$.

Feature map type	E_{total}	E_{low}	E_{high}	H_{wiener}	$P_{\max}$	Voicing	Zero crossings
1. Value at t_b	F, Δ	Δ	F, Δ	F, Δ	Δ		
2. Value at t_v	F, Δ	F, Δ	F, Δ	F, Δ	F, Δ	Δ	Δ
3. Mean/max over (t_b, t_v)					Δ/Δ		
4. Mean/max over (t_b, t_v-10)					Δ/Δ		
5. Mean over (t_b, t_v) - mean over $(1, t_b)$			F	F			
6. Mean over $(1, t_b-5)$			F	F		F	
7. Max over $(1, t_b-5)$			F	F			

B. A discriminative algorithm

We now describe a simple iterative algorithm for learning the weight vector $\mathbf{w}$, based on the *Passive-Aggressive* family of algorithms for structured prediction described in Crammer *et al.* (2006), where the interested reader can find a more detailed description. Pseudocode for the algorithm is given in Fig. 2.

The algorithm receives as input a training set $S = \{(\bar{\mathbf{x}}^i, t_b^i, t_v^i)\}_{i=1}^m$ of m examples and a parameter C, and works in rounds. At each round, an example is presented to the algorithm, and the weight vector $\mathbf{w}$ is updated. We denote by $\mathbf{w}^\tau$ the value of the weight vector after the τth iteration. Initially we set $\mathbf{w}^0 = \mathbf{0}$. Let $(\hat{t}_b^\tau, \hat{t}_v^\tau)$ be the cost-adjusted predicted onset pair for the ith example according to $\mathbf{w}^{\tau-1}$,

$$(\hat{t}_b^\tau, \hat{t}_v^\tau) = \arg\max_{(t_b,t_v)} \mathbf{w}^{\tau-1}\cdot\phi(\bar{\mathbf{x}}^i, t_b, t_v) \; + \; \gamma[(t_b,t_v),(t_b^i,t_v^i)]\,. \qquad (3)$$

We set the weight vector $\mathbf{w}^\tau$ to be the minimizer of the following optimization problem,

$$\min_{(\mathbf{w},\xi\geq 0)} \frac{1}{2}||\mathbf{w}-\mathbf{w}^{\tau-1}||^2 + C\xi$$
$$\text{s.t. } \mathbf{w}\cdot\phi(\bar{\mathbf{x}}^i, t_b^i, t_v^i) - \mathbf{w}\cdot\phi(\bar{\mathbf{x}}^i, \hat{t}_b^\tau, \hat{t}_v^\tau) \geq \gamma^\tau - \xi, \qquad (4)$$

where $\gamma^\tau = \gamma\left[(t_b,t_v),(\hat{t}_b^\tau,\hat{t}_v^\tau)\right]$, C serves as a complexity-accuracy trade-off parameter (as in the SVM algorithm), and ξ is a nonnegative slack variable that indicates the loss of the ith example. Intuitively, we would like to minimize the loss of the current example (the slack variable ξ) while keeping the weight vector $\mathbf{w}$ as close as possible to our previous weight vector $\mathbf{w}^{\tau\text{-}1}$. The constraint ensures that the projection of the manually labeled onset pair (t_b^i, t_v^i) onto $\mathbf{w}$ is higher than the projection of the predicted pair $(\hat{t}_b^\tau, \hat{t}_v^\tau)$ onto $\mathbf{w}$ by at least the cost function between them (γ^τ). It can be shown (Crammer *et al.*, 2006) that the solution to the above optimization problem is

$$\mathbf{w}^\tau = \mathbf{w}^{\tau-1} + \alpha^\tau\Delta\phi^\tau\,, \qquad (5)$$

where $\Delta\phi^\tau = \phi(\bar{\mathbf{x}}^i, t_b^i, t_v^i) - \phi(\bar{\mathbf{x}}^i, \hat{t}_b^\tau, \hat{t}_v^\tau)$. The value of the scalar α^τ, shown in Fig. 2, is based on the cost function γ^τ, the different scores that the manually labeled onset pair and the predicted pair received according to $\mathbf{w}^{\tau-1}$, and a parameter C.

```
INPUT: training set $S = \{(\bar{\mathbf{x}}^1, t_b^1, t_v^1), \ldots, (\bar{\mathbf{x}}^m, t_b^m, t_v^m)\}$ ;
        parameters $C$ and $\epsilon$
INITIALIZATION: $\mathbf{w}^0 = \mathbf{0}$
FOR $\tau = 1, 2, \ldots$
  Pick example $(\bar{\mathbf{x}}^i, t_b^i, t_v^i)$ from $S$
  Predict
    $(\hat{t}_b^\tau, \hat{t}_v^\tau) = \arg\max_{(t_b,t_v)} \mathbf{w}^{\tau-1}\cdot\boldsymbol{\phi}(\bar{\mathbf{x}}^i, t_b, t_v) + \gamma((t_b,t_v),(t_b^i,t_v^i))$
  Set $\Delta\boldsymbol{\phi}^\tau = \boldsymbol{\phi}(\bar{\mathbf{x}}^i, t_b^i, t_v^i) - \boldsymbol{\phi}(\bar{\mathbf{x}}^i, \hat{t}_b^\tau, \hat{t}_v^\tau)$
  Set $\alpha^\tau = \min\left\{C, \frac{\gamma\left((t_b,t_v),(\hat{t}_b^\tau,\hat{t}_v^\tau)\right) - \mathbf{w}^{\tau-1}\cdot\Delta\boldsymbol{\phi}^\tau}{\|\Delta\boldsymbol{\phi}^\tau\|^2}\right\}$
  Set $\mathbf{w}^\tau = \mathbf{w}^{\tau-1} + \alpha^\tau\Delta\boldsymbol{\phi}^\tau$
OUTPUT: $\mathbf{w}^* = \sum_\tau \mathbf{w}^\tau$
```

FIG. 2. Passive-Aggressive algorithm for training the VOT prediction function.

Given a training set of m examples we iterate over its elements, possibly M times (epochs), and update the weight vector $M \cdot m$ times. To classify unseen utterances, we use the average of $\{\mathbf{w}^1,\ldots,\mathbf{w}^{Mm}\}$, denoted by $\mathbf{w}^*$.

A theoretical analysis (Dekel *et al.*, 2004; Keshet *et al.*, 2007) shows that with high probability, the function learned using our algorithm will have good generalization properties: the expected value of the cost function when the algorithm is applied to unseen data is upper-bounded by the loss of the algorithm during training, plus a complexity term which goes to zero linearly with the number of training examples. For readers familiar with structural SVMs (Taskar *et al.*, 2003; Tsochantaridis *et al.*, 2004), we note that the same analysis suggests that the average loss of the Passive-Aggressive solution is comparable to the average loss of the structural SVM solution, while the structured Passive-Aggressive algorithm is much easier to implement and faster to train.

IV. EXPERIMENTS: PRELIMINARIES

We first describe the datasets used in our experiments (Sec. IV A), and the different methods used to evaluate the algorithm's performance (Sec. IV B).

A. Datasets

Our experiments make use of four datasets, each consisting of audio of English words beginning with voiceless stops (/p/, /t/, /k/). For each word, the algorithm described above for training a VOT prediction function requires the VOT boundaries (t_b, t_v) and the word boundaries, i.e., where to begin and end searching for the VOT. We describe how VOT boundaries and word boundaries were annotated for each dataset, and briefly describe relevant aspects of each dataset.

The datasets vary along several dimensions, summarized in Table II. Their speaking styles range in naturalness, from isolated words to read sentences to conversational speech. Three broad types of English accents are represented (American, British, Portuguese-accented). Finally, the recording conditions vary greatly. We perform experiments on several different datasets in order to test the robustness of our approach. The promise of learning a function to measure VOT is that it should perform well on diverse datasets because it can be retrained on data from each one. By using several datasets, we show empirically that this is the case.

TABLE II. Comparison of datasets used in experiments. A = American, B = British, L1 = first- language, L2 = second-language.

Dataset	Style	Dialect	Environment	N
TIMIT	read sentences	A	laboratory	5535
BB	conversational	B	TV studio	704
SWITCHBOARD	conversational	A	telephone	893
PGWORDS	isolated words	A (L1, L2)	laboratory	6795

1. TIMIT

The TIMIT corpus (Garofolo *et al.*, 1993) consists of segmentally transcribed sentences read by 630 American English speakers from 8 dialect regions. It is widely used by speech recognition researchers and to a lesser extent by phoneticians (e.g., Keating *et al.*, 1994). The TIMIT transcriptions are phonetic rather than phonemic, and there are two phone labels corresponding to each stop phoneme (/p/, /t/, /k/, /b/, /d/, /g/), for the closure and burst (e.g., `pcl` and `p` for /p/). Thus, each underlying stop can be annotated as a closure alone, a burst alone, a closure and burst, a different phone altogether, or nothing (if it is deleted). We restrict ourselves to all words (excluding `SA1` and `SA2` utterances) transcribed as beginning with an unvoiced stop burst (either preceded by a closure or not), followed by a voiced segment; this results in 5535 tokens, from all 630 speakers.

The VOT boundaries (t_b, t_v) were taken to be the burst boundaries from the TIMIT transcription. Because the burst sometimes ends after the onset of voicing, this step is an approximation, one which allows us to take advantage of the size of TIMIT, and test our algorithm on a widely used dataset. The word boundaries were also taken from the TIMIT transcription, except for some pathological cases where a word boundary coincided with the beginning or end of the burst. For words annotated as beginning with only a burst (and no closure), the left word boundary was taken to be 50 ms before the burst onset. For words annotated as consisting solely of an unvoiced stop (e.g., "to" transcribed as `tcl t`), the right word boundary was taken to be 25 ms after the end of the burst. These corrections were made because our algorithm assumes that the burst and voicing onsets lie within the host word.

2. Big Brother (BB)

This corpus consists of spontaneous speech from the 2008 season of Big Brother UK, a British reality television show. The speech comes from four British speakers recorded in the "diary room," an acoustically clean environment, using clip-on microphones; sound quality is generally very good. The data used here, a subset of the corpus described in Bane *et al.* (2012), are VOTs for 704 word-initial voiceless stops, manually annotated by (one of) two transcribers. The end of each word has also been annotated. Because the beginnings of words have not been annotated, we took the left word boundary of each word to be 25 ms before the burst onset. Stops with no following voiced segment were kept if there was still abrupt spectral change at the end of the burst, and excluded otherwise.

3. SWITCHBOARD

The SWITCHBOARD corpus (Godfrey and Holliman, 1997) consists of spontaneous speech from telephone conversations between American English speakers. We chose subsets of 8 conversations, corresponding to 16 speakers. VOTs for all word-initial voiceless stops in these subsets were manually annotated by one transcriber if a burst was present (e.g., the stop was not realized as a flap), resulting in 893 examples. The boundaries of each word were manually determined. When a word boundary coincided with the burst or voicing onset (e.g., for a word realized as an isolated stop, with no following voicing), the word boundary was adjusted slightly left or right (for the left or right word boundaries, respectively), because our algorithm assumes that the burst and voicing onsets lie within the host word.

4. Paterson/Goldrick words (PGWORDS)

This corpus consists of data from a laboratory study by Nattalia Paterson and Matt Goldrick (Paterson, 2011), investigating VOT in the speech of American English monolinguals and Brazilian Portuguese (L1)-English bilinguals. In this study, each of 48 speakers (24 monolinguals, 24 bilinguals) produced 144 isolated words, each beginning with a stop (/p/, /t/, /k/, /b/, /d/, /g/), in a picture naming task. Productions other than the intended label as well as those with code-switching or disfluencies were excluded. The VOT of each remaining word was manually measured by a single transcriber. We consider a subset of 6795 VOTs from this data, only from words beginning with voiceless stops. Because this dataset consists of words spoken in isolation, the choice of word boundaries is somewhat arbitrary. We took the left boundary to be 50 ms before the burst onset and the right boundary to be when the next prompt was given to the subject (1–3 s later).

B. Evaluation methods

There is no single obvious method for evaluating the performance of an automatic VOT measurement algorithm. Several methods have been used in previous work on automatic measurement, all based on the degree of discrepancy between automatic and manual measurements. Below, we measure our algorithm's performance by three methods: pure automatic/manual measurement discrepancy, comparison of automatic/manual discrepancy to intertranscriber agreement, and comparison of statistical models fit to automatic and manual measurements. We now describe each method and its motivation.

1. Distribution of automatic/manual difference

The most common evaluation method used in previous work is examination of the distribution of differences between automatic and manual VOT measurements across a dataset. The algorithm's performance can then be reported either as the full (empirical) CDF of automatic/manual differences (as in Stouten and van Hamme, 2009), or as the percentage of examples with automatic/manual difference below some fixed set of values, the *tolerances* (as in Lin and Wang, 2011). Reporting statistics about the CDF of automatic/manual differences is a standard evaluation method in ASR tasks, such as forced alignment of phoneme sequences, where the goal is to predict the location of boundaries in a speech segment (e.g., Brugnara *et al.*, 1993; Keshet *et al.*, 2007). In our experiments, we always report performances at fixed tolerances, and report the full CDF when it gives helpful additional information.

Two other evaluation methods, where a single measure of error is calculated from the set of automatic/manual differences, have also been used in previous work. These are discussed in Sec. VII, where they are used to compare our algorithm with previous work.

2. *Comparison to interrater reliability*

A disadvantage of evaluation using the distribution of automatic/manual differences is that it is not clear what the gold standard is. VOT measurements for the same example are expected to vary somewhat between transcribers, or within a single transcriber (measuring at different times). Intuitively, progressively better automatic/manual agreement is good up to a point, but automatic/manual agreement which is *too* good means overfitting to the particular set of manual measurements. One solution is to compare the automatic/manual CDF to interrater reliability (IRR): a CDF of differences between two transcribers' VOT measurements. In this view, the gold standard is for automatic and manual measurements to agree as well as two sets of manual measurements of the same data. We compare the predicted/manual difference CDF to an IRR CDF for experiments on all datasets where IRR data is available (BB, SWITCHBOARD, PGWORDS).

3. *Model-based comparison*

Our last evaluation method is more directly related to the setting in which VOT is usually measured: phonetic studies addressing clinical or theoretical questions. In such studies, the question is how some predictor variables—such as the stop consonant's place of articulation, or whether the speaker has Parkinson's disease—affect VOT across a dataset. This is typically assessed by performing a statistical analysis (such as analysis of variance or multiple linear regression) of the effect of the predictors on VOT, and reporting the statistical significance and values of model parameters of interest. Thus, to test whether automatic VOT measurements from our algorithm can be used to replace manual measurements, a sensible test is to compare the *statistical models* induced from automatic and manual measurements for the same dataset, rather than directly comparing the automatic and measurements of individual tokens (as in the evaluation methods described above). The goal is for the values and statistical significances of the two models to be as similar as possible. We note that good performance on this model-based evaluation method does not trivially follow from good performance on an evaluation method based on individual tokens, or vice versa.

C. Experiments: Overview

The next four sections describe a series of experiments to evaluate the algorithm's performance, using each of the evaluation methods just described. In Sec. V we describe experiments using the full amount of data available, and where training and testing data are (disjoint subsets) from the same dataset; we call these *base experiments*. We then (Sec. VI) evaluate the robustness of the results obtained in the base experiments to decreasing the amount of training data, or training and testing on different datasets. In Sec. VII we compare our algorithm to previous work as closely as possible. Finally, we evaluate the algorithm by model-based comparison (Sec. VIII).

In all experiments, we only considered burst onsets t_b within 0–150 ms of the start of the word, and voicing onsets t_v 15–200 ms later than t_b; this step attempts to restrict the algorithm's focus to the first two segments of each word.

V. EXPERIMENTS I: BASE

The evaluation method used for the base experiments is simply the distribution of automatic/manual differences. Where IRR data is available (all datasets except TIMIT), this distribution is compared to the distribution of differences between transcribers.

The structure of each base experiment was the same: the dataset was split into training, development, and test sets corresponding to subsets of speakers. The parameters C, ϵ, and M (number of epochs) were tuned by training a weight vector on the training set for each parameter triplet in the ranges $C \in 0.01, 0.1, 1, 5, 10, 100$, $\epsilon \in \{2, 3, 4, 5\}$, and $M \in \{1,\ 2,\ 3,\ 4,\ 5\}$ (for TIMIT, PGWORDS) or $M \in \{1, 2, 3, 4, 5, 6, 7, 9, 11, 15\}$ (for BB, SWITCHBOARD).[1] The weight vector was selected which gave the lowest mean absolute difference between predicted and actual VOT over examples in the development set. This $\mathbf{w}^*$ was then applied to predict VOTs for examples in the test set.

For each experiment, Table III summarizes the distribution of automatic/manual differences over the test set, and the distribution of intertranscriber differences over the set of double-transcribed examples (except for TIMIT).

For the BB dataset, the training/development/test sets consisted of 405/142/160 examples (2/1/1 speakers), and the parameter values chosen by tuning on the development set were $C = 5$, $\epsilon = 3$, and $M = 15$. A subset of the data (108 stops; 15.3%) was double transcribed, by two independent transcribers. (Neither one trained the other, and there was no attempt made to synchronize transcription criteria.) In comparison to IRR, the algorithm performs very well: the automatic/manual and IRR performances at each tolerance are extremely similar (within 1.2%). That is, the automatic measurements match manual measurements as well as two human transcribers match each other.

For the SWITCHBOARD dataset, the training/development/test sets consisted of 563/102/288 examples (6/1/2 speakers), and the parameter values chosen by tuning on the development set were $C = 1$, $\epsilon = 5$, and $M = 5$. A subset of the data (171 stops; 19.1%) was double-transcribed, by two semi-independent transcribers. (One transcriber had trained the other about one year previously, on a different dataset,

TABLE III. Performance in base experiments (Sec. V), given as percentage of examples in the test set with automatic/manual difference (for all datasets) or intertranscriber difference (for all datasets except TIMIT) below a series of fixed tolerance values. (For example, 46.1% of examples in the TIMIT test set had automatic and manual measurements differing by ≤2 ms.)

Dataset	Experiment	≤2 ms	≤5 ms	≤10 ms	≤15 ms	≤25 ms	≤50 ms
BB	Auto/manual	53.5	79.3	88.1	93.1	96.2	98.7
	Intertranscriber	54.4	79.6	89.3	93.2	96.1	99.0
SWITCHBOARD	Auto/manual	53.1	73.3	83.3	89.0	93.4	96.5
	Intertranscriber	52.9	70.0	82.4	88.8	94.1	99.4
PGWORDS	Auto/manual	49.1	81.3	93.9	96.0	97.2	98.1
	Intertranscriber	61.9	90.0	96.9	98.6	99.5	100.0
TIMIT	Auto/manual	46.1	67.2	85.0	94.7	98.1	99.0

but no attempt at synchronizing transcription criteria was made for the SWITCHBOARD data.) The algorithm again performs very well, by comparison to IRR: automatic/manual differences are slightly lower than intertranscriber differences at tolerances up to about 20 ms, and slightly higher above 20 ms, becoming significantly higher above 40 ms.

For the PGWORDS dataset, the training/development/test sets consisted of 4151/403/1340 examples (28/3/6 speakers), and the parameter values chosen by tuning on the development set were $C = 10$, $\epsilon = 5$, and $M = 2$. A subset of the data (591 stops; 7.3%) was double-transcribed, by two transcribers. (One transcriber trained the other, and they worked together on synchronizing measurement criteria for this dataset.) The algorithm performs less well on this dataset than for BB or SWITCHBOARD, by comparison to IRR: automatic/manual VOT measurement differences are higher than intertranscriber differences, at all tolerances. A possible explanation for this difference in performance is that the intertranscriber data for the three datasets are not comparable. The transcribers for PGWORDS worked together to synchronize their measurement criteria on this dataset, while the transcribers for BB and SWITCHBOARD did not. Thus, the algorithm's performance on PGWORDS might be closer to IRR if intertranscriber data were used from two independent transcribers.

For the TIMIT dataset, we used Halberstadt's (1998) split of speakers into training, development, and test sets (specifically, "full" test) consisting of 4132/397/1006 examples (462/50/118 speakers). The parameter values chosen by tuning on the development set were $C = 5$, $\epsilon = 4$, and $M = 2$. Performance for this dataset is worse than other datasets (in the sense of greater automatic/manual differences) for tolerances up to about 10 ms, and slightly better at tolerances above 10 ms. However, it is not clear how comparable the results for different datasets are, given that the TIMIT annotations actually denote burst boundaries rather than VOT. Below (Sec. VII) we will more directly evaluate our TIMIT results, by comparing them with previous work on automatic VOT estimation which also uses test data from TIMIT.

VI. EXPERIMENTS II: ROBUSTNESS

The base experiments show that our algorithm generally performs very well on several datasets, evaluated against IRR; we show below that it also performs well relative to previous work (Sec. VII). However, in both cases we assume ideal training conditions: a relatively large training set is available to train **w***, and the training and test sets consist of examples from the same corpus. In contrast, the typical use case for a VOT measurement algorithm is a corpus where little or no annotated data is available. For our algorithm to be practically useful, we must test how performance varies as these conditions are relaxed. If relatively few examples are needed to train **w***, other researchers can annotate a small subset of data to train our algorithm; if performance varies little when the training and test corpora are not the same, researchers working on a new dataset can use one of our weight vectors pretrained on a large corpus. This section presents experiments testing the algorithm's robustness to decreasing the amount of training data (Sec. VI A), and to mismatched training and testing datasets, where a weight vector trained on one corpus is used to measure VOT for data from a different test corpus (Sec. VI B).

A. Varying the amount of training data

For each of the base experiments, we tested the robustness of our algorithm to decreasing the amount of training data, holding the test set constant, as follows. Let N be the size of the training set for a given experiment. We chose a series of percentages p of the test set, such that pN spanned the range $(0, N]$, including N ($p = 1$). (We did not use the same values of p for each dataset because N varies greatly across our datasets.) For each p we chose a random subset of the training set of size pN and re-ran the experiment, using the same test set as the original ($p = 1$) experiment, and using the same parameter values (for C, ϵ, and M) as in the original experiment. Since the results depend on the particular subset of pN chosen, this procedure was repeated 25 times for each p.

The results of the experiments are summarized in Fig. 3. To focus on how performance changes as the amount of training data is decreased, we show results only for a subset of tolerances (and do not show the full CDFs of automatic/manual differences). Each point and its associated errorbars represent the mean and ±2 standard deviations of the 25 runs at a fixed amount of training data.

For all datasets, the algorithm's performance is extremely robust to decreasing the amount of data. Performance stays essentially constant (error bars overlap with those for the full training set)—until the amount of data is decreased below 250 training examples for PGWORDS and TIMIT, and below 25–50 examples for SWITCHBOARD and BB. Performance decreases more at lower tolerances

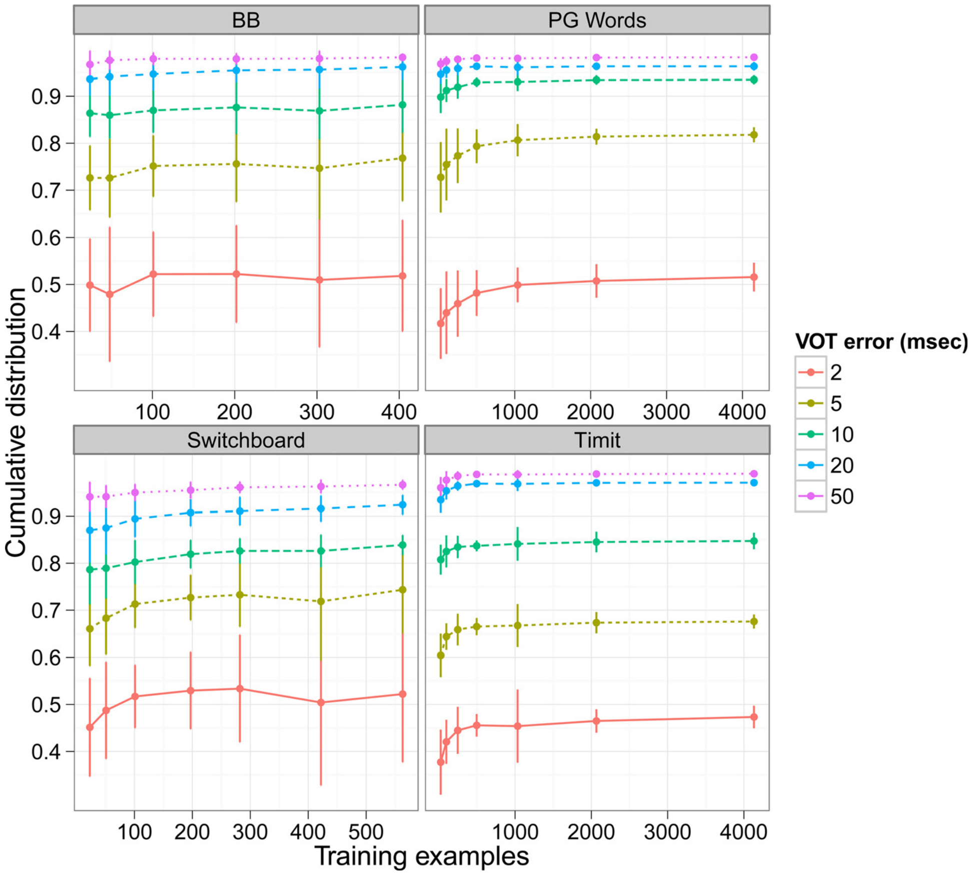

FIG. 3. (Color online) Results for experiments varying the amount of training data (Sec. VI A): Percentage of tokens with automatic/manual difference below tolerance values (2, 5, 10, 20, 50 ms) as the amount of training data is varied. Points and errorbars indicate means ±2 standard deviations across 25 runs.

(2–5 ms) than at higher tolerances (10–50 ms), especially for TIMIT and PGWORDS. To speak more quantitatively, we can focus on performance at 10 ms tolerance, shown in Table IV. Across all datasets, training on just 25 examples decreases performance at this tolerance by 1.8%–5.2%. These results suggest our algorithm can be quickly adapted to a new dataset with little training data.

B. Mismatched training and test corpora

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 12 additional experiments, corresponding to all possible choices of two distinct datasets for training and testing **w***. We denote the weight vectors trained on each corpus in the base experiments as $\mathbf{w}^*_{\text{TIMIT}}$, etc. The weight vector for each corpus is applied to give automatic measurements for examples in the test sets of the other three corpora.

TABLE IV. Mean performance at 10 ms tolerance (across 25 runs) in experiments where the amount of training data is decreased (Sec. VI A), with number of training examples in parentheses, for the lowest number of training examples (n_1), the highest number of training examples (N), and the lowest number of training examples (n_2) for which performance is within 2σ of mean performance with the highest number of training examples.

Dataset	n_1	n_2	N
BB	86.4 (24)	86.4 (24)	88.2 (404)
SWITCHBOARD	78.6 (23)	78.6 (23)	83.8 (563)
TIMIT	80.8 (25)	82.5 (99)	84.7 (4132)
PGWORDS	89.8 (25)	91.2 (100)	93.5 (4151)

The distributions of automatic/manual differences for each pair of training and testing corpora are shown in Fig. 4. To discuss performance differences quantitatively, it will again be helpful to refer to performances at 10 ms for each curve, given in Table V.

We note some patterns in these results. First, examining the full CDFs, it is always the case that the best performance for a test set from a given corpus is achieved using training data from that corpus. (This is visually clear except for the TIMIT test set, where the CDF corresponding to $\mathbf{w}^*_{\text{TIMIT}}$ is in fact higher than the CDF corresponding to $\mathbf{w}^*_{\text{PGWORDS}}$ at all tolerances.) Better performance when training and test data are drawn from the same distribution is not surprising, but it is useful to investigate how much performance drop to expect, for potential applications of the algorithm where retraining on data drawn from the same distribution as the test corpus would not be possible. (For example, real-time VOT detection in a novel recording environment.)

How performance changes when different training and test corpora are used depends largely on the test corpus. For the BB, TIMIT, and SWITCHBOARD test sets, there is significant variance in how much using a different test corpus affects performance (1%–11% at 10 ms), with some mismatched train/test pairs achieving performance near the corresponding matched train/test conditions in certain tolerance ranges. Performance on the PGWORDS test set is more dramatically affected by training on a different corpus, with a

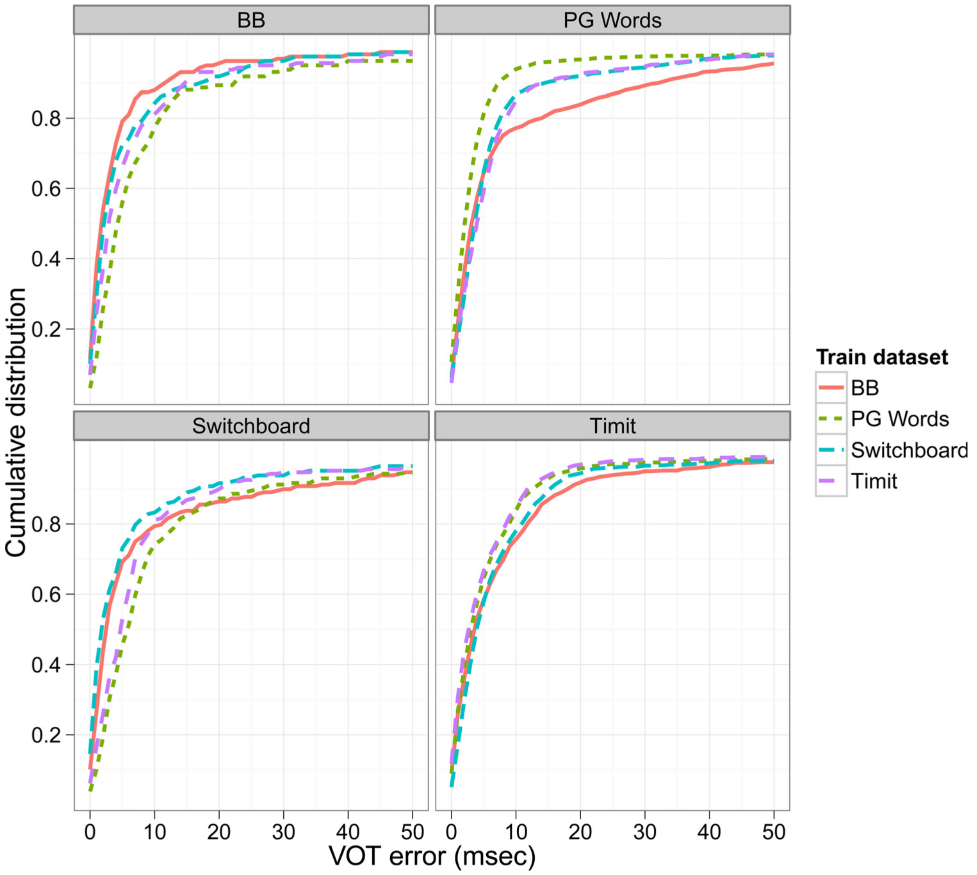

FIG. 4. (Color online) Distribution of automatic/manual differences as a function of the dataset the training set is drawn from, holding the test set constant. Lines for mismatched train/test datasets correspond to experiments described in Sec. VI B. Lines for same train/test datasets correspond to the base experiments (Sec. V).

7%–17% performance drop at 10 ms depending on the training set. It is not clear why changing the weight vector used matters more for PGWORDS, compared to the other datasets.

C. Discussion

In this section, we have described experiments testing the robustness of our algorithm's performance to decreasing the amount of training data, and on mismatched training and test conditions. We found that performance is very robust to decreasing the amount of training data, and that the effect of mismatched training and test datasets depends on the particular datasets used. The motivation for these experiments was to determine whether our algorithm can be practically useful in applications to *new* datasets, without manually labeling a large amount of data. Our results suggest a positive answer: only a small number of manually labeled VOTs (<250) are needed for training, in addition to a small number for validation (perhaps 50–200), to achieve near-maximum performance.

TABLE V. Mean performance at 10 ms tolerance in experiments with mismatched training and test corpora (Sec. VI B).

	Test corpus			
Training corpus	BB	SWITCHBOARD	TIMIT	PGWORDS
BB	88.0	79.4	75.6	77.2
SWITCHBOARD	84.3	83.3	78.0	86.9
TIMIT	81.1	81.1	84.8	84.9
PGWORDS	77.4	74.1	83.9	93.9

VII. EVALUATION I: COMPARISON WITH PREVIOUS WORK

In this section, we compare our algorithm's performance with all previous studies on automatic VOT measurement (to our knowledge) which have examined agreement between automatic and manual measurements. We are able to compare directly (testing on the same test set) to two previous approaches (Stouten and van Hamme, 2009; Lin and Wang, 2011), and indirectly to two other approaches (Yao, 2009; Hansen *et al.*, 2010).

A. Stouten and van Hamme (2009)

Stouten and van Hamme consider voiced and voiceless stops with bursts in TIMIT, in all positions. They perform manual VOT measurements for a subset of 582 stops (the "manual" dataset), and compare these to their automatic measurements. For each stop, an HMM-based forced aligner is first applied to the TIMIT phone transcription to find the approximate location of the stop's burst. A knowledge-based algorithm operating on time frequency reassigned spectrograms is used to determine the burst and voicing onsets. If either is not found, the force-aligned burst boundaries are used as a fallback to determine VOT.

We applied our algorithm to the 293 voiceless stops from SvH's "manual" dataset, using **w*** from the TIMIT base experiment. Because we are now not dealing only with stops in initial position, the left boundary where the algorithm begins searching for t_b was determined differently from our earlier TIMIT experiments. Each example was

TABLE VI. Comparison of results for test data from Stouten and van Hamme (2009), using their algorithm and using our approach.

Algorithm	≤2 ms	≤5 ms	≤10 ms	≤15 ms	≤25 ms	≤50 ms
Stouten and van Hamme (2009)	28.6	42.8	77.0	86.2	**94.7**	**99.5**
Our approach	**44.6**	**67.6**	**85.1**	**91.1**	94.1	96.1

taken to start at the beginning of the segment preceding the burst (i.e., the closure, if one was present), and end at the right word boundary, where the segment and word boundaries were taken from TIMIT.

Table VI summarizes the distribution of automatic/manual differences, relative to SvH's manual measurements, for the two automatic measurement methods: our method (with **w*** from the TIMIT base experiment) and SvH's method. The error distribution for SvH was determined from Fig. 5 of Stouten and van Hamme (2009), using voiceless stops only. (We averaged the CDFs for /p/, /t/, and /k/, weighted by the number of tokens for each in the "manual" dataset.) Our method performs better for tolerances below about 22 ms, corresponding to 94% of examples.

A few differences between our setup and SvH's are relevant for comparing our results. To determine where to begin searching for the burst and voicing onsets, SvH use the force-aligned burst boundaries, while we assume the left word boundary is known. It is possible that our results would worsen using force-aligned word boundaries.

In addition, SvH's method includes a fallback step (in case a burst or voicing onset is not detected) where VOT is set to the force-aligned burst's duration, while ours does not. The fallback step occurs for two types of examples. For some, the forced-aligned burst boundaries are off; for others, the alignment is correct, but there is no prominent burst or voicing onset. On the first type of example our algorithm should do better, since the word boundary is known; on the second type, SvH's algorithm will likely do better. Most gross errors by our algorithm are in fact cases lacking either a clear burst onset or a clear voicing onset. Our method's prediction in these cases is often wildly off, while SvH's falls back to the force-aligned burst duration. Thus, it is not clear what net effect the inclusion of a fallback step has on the SvH results relative to ours.

Finally, our algorithm has one clear disadvantage in the comparison with SvH: it was only trained on initial stops, but tested in stops in all positions, and the training and testing data were labeled by different annotators.

B. Lin and Wang (2011)

Lin and Wang automatically measure VOT for word-initial stops from TIMIT, using a multi-step process. Their approach makes use of two tools: (1) an HMM-based forced aligner, at either the state or phone level (they try both), using MFCC features; (2) two random forest detectors, trained to detect the onset of a burst phone and the onset of a voiced phone. For a given utterance, the forced aligner is first applied to the TIMIT phone transcription to find the approximate location of the burst for each stop. For a given stop, the random forest detectors are then deployed to find the burst and voicing onsets, possibly selecting from several candidates. If no candidates are found for an onset, the force-aligned burst boundary is used instead. The burst boundaries from the TIMIT annotation are used as a proxy for VOT (as we also did in our TIMIT experiments above).

Lin and Wang test their algorithm on 2344 word-initial stops from the TIMIT `test` set, of which 1174 are voiceless. We applied our algorithm to the voiceless stops, using the **w*** from our TIMIT experiments above. Because this set of voiceless stops forms a subset of the complete TIMIT test set used in our experiments above, the word boundaries for each example have already been determined.

Table VII shows performance on the test set for Lin and Wang's method (their Table IV, "voiceless" row) and our method. (Results are shown in the format used in their paper.) Our algorithm gives better performance at all tolerances, on average by 1.8%.

There are two important differences between our setup and that of Lin and Wang. First, as Lin and Wang note with respect to results in a previous paper (Sonderegger and Keshet, 2010), our training set contains data from many more TIMIT speakers than theirs (462 speakers in our setting versus 4 speakers in theirs). While the exact numbers of speakers used in the two approaches are not comparable because of differences in the training procedures,[2] we have shown above (Sec. VI A) that our method is very robust to decreasing the amount of training data.

Second, like SvH, Lin and Wang use the force-aligned TIMIT phone transcription to determine where to search for the burst and voicing onsets, while we assume the left word boundary is known. It is again possible that our results would worsen using force-aligned word boundaries.

C. Yao (2009) and Hansen *et al.* (2010)

Our results can be compared less directly with two other studies where automatic and manual measurements are compared. Yao (2009) determines VOT for initial voiceless stops in the Buckeye Corpus, using MFCC "spectral templates" in a knowledge-based algorithm. The evaluation metric used is the RMS error for automatic measurement of t_b only. Hansen *et al.* (2010) determine VOT for initial voiceless stops from CU-Accent, a corpus of laboratory speech consisting of single-word productions by native and non-native English speakers. They measure VOT with a knowledge-based algorithm, acting on the Teager Energy Operator representation of the speech signal. The evaluation metric used is the percentage of stops where the automatic measurement differs by <10% from the manual measurement.

TABLE VII. Comparison of results for test data from Lin and Wang (2011), using their algorithm and using our approach.

	<5 ms	<10 ms	<15 ms	<20 ms
Lin and Wang (2011)	58.9	80.7	90.5	94.2
Our approach	**60.5**	**81.4**	**93.0**	**96.8**

TABLE VIII. Base experiments performance evaluated by metrics used by Yao (2009) and Hansen *et al.* (2010).

	RMS t_b error (ms)	$<$10% VOT error
TIMIT	6.5	66.5
BB	5.8	73.4
SWITCHBOARD	10.5	68.7
PGWORDS	6.8	81.2
Yao (2009)	10.8	–
Hansen *et al.* (2010)	–	74.9

Table VIII gives these metrics for the experiments on our four datasets reported above. With the caveat that comparison is difficult because of the different datasets used, our experiments' performance on these metrics compares favorably to previous work. All experiments have RMS t_b error less than Yao (2009). Our best-performing experiment, PGWORDS, does better than Hansen *et al.* (2010) on the $<$10% VOT error metric. Importantly, the PGWORDS dataset is arguably the most comparable to the CU-Accent dataset of Hansen *et al.* both consist of single-word productions in laboratory conditions, produced by both native and non-native speakers.

VIII. EVALUATION II: REGRESSION MODEL COMPARISON

We now evaluate the algorithm by comparing the regression models induced by automatically and manually measured data, as described above (Sec. IV B 3), for the PGWORDS dataset. We model two well-documented patterns of variation in VOT in this dataset. VOT is affected by the stop consonant's place of articulation (POA), with the expected pattern /p/ $<$ /t/ $<$ /k/; (e.g., Cho and Ladefoged, 1999). VOT is also affected by the speakers linguistic knowledge. Bilinguals who speak languages with contrasting VOT systems produce distinct VOT patterns from monolingual speakers, reflecting the interaction of their two linguistic systems. When speaking English, bilinguals whose native language contrasts prevoiced vs short-lag VOTs produce shorter VOTs than those of English monolinguals (e.g., Fowler *et al.*, 2008). Thus, we expect the bilingual Portuguese-English speakers to have lower VOT than the monolingual English speakers in the PGWORDS data.

For each set of measurements (automatic, manual), we build a mixed-effects linear regression model of how VOT depends on a speakers's language background and the place of articulation of the initial consonant of the host word. [We do not describe mixed models in depth; see e.g., Hox (2010), or Baayen *et al.* (2008) for the particular type of model with "crossed random-effects" used here.] We first describe the model's structure, then compare the results of fitting it to automatic and manual measurements on the PGWORDS dataset.

A. Model description

VOT is modeled as the sum of several types of terms:

(1) An overall mean value;
(2) A speaker-specific adjustment to the mean; adjustments are normally distributed across speakers;
(3) The same, for words;
(4) Terms indexing the speaker's L1, what phone the word begins with, and the product of the two, which allows how VOT depends on POA to differ depending on the speaker's L1.

Terms in (2) and (3) are called *random-effects*; terms in (4) are called *fixed-effects*.

Formally, each set of measurements consists of n data points, with VOT values $y_1, \ldots, y_n$. Let $s[i]$ and $w[i]$ be the speaker and the word corresponding to data point i. Three input variables index L1 background and phone for each data point:

(1) x_i^1 : 1 if speaker $s[i]$ has L1 = English, 0 otherwise;
(2) x_i^2 : 1 if word $w[i]$ begins with /k/, 0 otherwise;
(3) x_i^3 : 1 if word $w[i]$ begins with /t/, 0 otherwise.

One variable indexes L1 (which has two possible values) and two variables index place of articulation (which has three possible values) so that the model remains identifiable. Specifying different values (0 or 1) for these three variables allows us to express any combination of L1 and place of articulation.

We denote the predictors and fixed-effect coefficients as $\mathbf{x}_i = (1, x_i^1, x_i^2, x_i^3, x_i^1 x_i^2, x_i^1 x_i^3)$ and $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)$. VOT is then modeled as

$$
\begin{aligned}
& y_i = \boldsymbol{\beta} \cdot \mathbf{x}_i + \gamma_{s[i]} + \delta_{w[i]} + \epsilon_i \\
& \gamma_{s[i]} \sim N(0, \sigma_s^2), \;\; \delta_{w[i]} \sim N(0, \sigma_w^2), \;\; \epsilon_{\mathrm{i}} \sim N(0, \sigma^2). \qquad (6)
\end{aligned}
$$

B. Model summary

Below, we describe the models by summarizing their fixed-effect and random-effect terms. For the fixed-effects, we give coefficient estimates $(\hat{\beta})$ and their standard errors $[\mathrm{SE}(\hat{\beta})$, as well as corresponding p-values quantifying the significance of the coefficient estimate (for a Wald test applied to $\hat{\beta}/\mathrm{SE}(\hat{\beta}))]$.[3] While the random-effects for each speaker and each word ($\delta_{w[i]}$, $\gamma_{s[i]}$) are not actually fitted parameters, it is possible to extract estimates of them known as best linear unbiased predictors (BLUPs) (Pinheiro and Bates, 2000). We will show BLUPs for the deviation of each speaker and word from the mean, along with their standard errors.

C. Comparison of models

We constructed two models, one fit using automatic measurements (the "automatic model") and one using manual measurements (the "manual model") as follows. The PGWORDS dataset was split into a `split` and `heldout` set, such that `split` contained a random 75% of data points for each speaker, and `heldout` contained the remaining 25%. An automatic measurement was assigned to each data point in `split` by applying our algorithm (with the values of C, M, and ϵ used for the PGWORDS base experiment), using four-fold cross validation. (Speakers were split randomly into four groups; for each group, automatic VOT measurements were computed using $\mathbf{w}^*$ trained on data from the other 3 groups.) Each data point in `split` now had one automatic and one manual measurement, resulting in two datasets, differing only in whether VOT (y_i, in the notation above) was measured automatically or manually.

Each of the two datasets was trimmed for outliers, by discarding measurements further than 3 standard deviations from the mean within a speaker. For each dataset, a model of the form in Eq. (6) was fit using the `lmer` function in the R package `lme4` (Bates *et al.*, 2011). We compare the automatic and manual models in two ways: by fitted model parameters, and by predictions on held-out data.

1. Comparison of model parameters

We first consider the two models' fixed-effects, then turn to the BLUPs of their random-effects. Table IX shows the fixed-effect coefficient estimates, along with their standard errors and associated significances. The estimates and standard errors are extremely similar in the two models, with no fitted coefficient having a value in one model more than 1.25 standard errors away from its value in the other model. The significances of the fixed-effect coefficients in the two models are also very similar: all coefficients are highly significant except β_4, which is marginal in the automatic model and not significant in the manual model.

The group means of the empirical data are /p/ = 67.9 ms, /t/ = 80.6 ms, /k/ = 79.7 ms for monolinguals; and /p/ = 41.3 ms, /t/ = 52.8 ms, /k/ = 65.2 ms for bilinguals. The fixed-effects for both models suggest that the trends observed in the empirical data are significant. English speakers have longer VOTs than bilingual Portuguese-English speakers and VOT depends on place of articulation as /p/ < /t/ < /k/, both expected results. There is also a significant interaction between L1 and phone. For bilingual speakers the entire /p/ < /t/ < /k/ pattern is significant. For monolinguals, the VOT difference between /p/ and /t/ is greater, and the VOT difference between /p/ and /k/ is smaller. The pattern of VOT dependence on place of articulation is thus closer to /p/ < /t/ = /k/ for monolinguals, consistent with some previous studies of VOT in English word-initial voiceless stops (e.g., Cooper, 1991; Docherty, 1992).

Turning to the random-effects, the automatic and manual models each predict a random intercept for each speaker (the $\gamma_{s[i]}$), and a 95% confidence interval (1.96 × standard errors). The models predict similar deviations from the mean for each speaker, with the confidence intervals for $\gamma_{s[i]}$ overlapping for all but 1 of the 34 speakers. The models also predict similar deviations from the mean for each word (the $\delta_{w[i]}$), with the 95% automatic and manual confidence intervals overlapping for all 206 words.

TABLE IX. Summary of fixed-effects in automatic and manual models: fixed-effects coefficient estimates ($\hat{\beta}$), their standard errors, associated t statistic ($\hat{\beta}/\mathrm{SE}(\beta)$), and p-values.

	Automatic model				Manual model			
	$\hat{\beta}$	SE($\hat{\beta}$)	t	p	$\hat{\beta}$	SE($\hat{\beta}$)	t	p
β_0	52.8	2.8	18.6	***[a]	51.5	2.9	17.7	***
β_1	11.0	2.4	4.6	***	10.6	2.5	4.3	***
β_2	8.9	2.1	4.3	***	9.6	2.1	4.7	***
β_3	17.1	1.7	10.3	***	17.7	1.7	10.6	***
β_4	1.4	0.8	1.8	0.072	0.79	0.72	1.1	0.27
β_5	−4.7	0.7	−7.0	***	−5.5	0.6	−8.8	***

[a] The three asterisks denote $p < 0.0001$.

Thus, both the fixed-effects and random-effects are very similar for the two models. The similarity of the fixed-effect coefficients means that the models make quantitatively similar predictions for the effects of place of articulation and first language. The similarity of the random-effect BLUPs means that the two models predict similar deviations from the overall mean for each individual speaker and word.

2. Comparison of model predictions

We can also compare the models' predictions on the 25% of held-out data, to get a sense of how similar their predictions are on unseen data. The automatic and manual models make extremely similar predictions, differing by ≤5 ms for 90.2% of data points in the held-out set, and with correlation $r = 0.992$ and mean absolute difference 2.31 ms across all data points in this set. By comparison, measurements by two human transcribers on a subset of the PGWORDS dataset (see Sec. V) differ by ≤5 ms for 90.0% of data points, and have correlation $r = 0.987$ and mean absolute difference 2.49 ms. Thus, the automatic and manual models make predictions which agree as well as two human transcribers.

IX. DISCUSSION

A. Summary

We have described a machine-learning approach to the problem of automatic VOT measurement which treats this task as a case of structured prediction. A function to measure positive VOT is learned from manual measurements, using a discriminative large-margin training procedure which aims to minimize error in the difference between predicted and actual VOT. The function takes as input feature maps which are specialized for the task of VOT measurement. Because our system is trainable, it can adapt to particular datasets and measurement criteria.

In a first set of experiments, we showed that the algorithm achieves excellent performance for each of four datasets, when all data available for training is used, and in particular near-IRR performance on the three datasets were IRR data was available. In a second set of experiments, we showed that the algorithm is robust to decreasing the amount of training data, with performance remaining essentially constant down to about 50–250 training examples (depending on the dataset). Thus, the algorithm is adaptable to new datasets with relatively little effort. We also found that performance generally suffers for mismatched versus matched training and test corpora. Thus, the algorithm is learning something about the particular type of speech and measuring criteria used for each dataset.

The algorithm generally outperforms previous work where automatic and manual VOT measurements are compared, with the caveat that precise comparisons are difficult because of differences in the datasets and experimental setups used. We also evaluated the algorithm by comparing two mixed-effects regression models for the effect of several covariates on VOT in a dataset of laboratory speech: one model fitted using automatic measurements, and the other fitted using manual measurements. The two models were extremely similar, both in terms of fitted model parameters and predictions on held-out data. This shows that a study of how the covariates affect VOT in this dataset would have reached the same conclusions whether VOT measurements were done manually, or automatically using our method.

B. Future directions

In this paper, we have considered word-initial English voiceless stops, because we know that they are nearly always realized with a burst, and hence have positive VOT. (In TIMIT, for example, 99.6% of word-initial voiceless stops for words other than "to," which is sometimes flapped, are realized with a burst.) However, for stops which are not word-initial or not voiceless, this is often not the case. English stops in non-initial position are often not realized with a burst; for example, Randolph (1989) found that in 3 corpora of read speech (including TIMIT), 31% of stops occurring as syllable codas were realized with a burst, compared with 97% for syllable-initial stops. Voiced stops in English are sometimes realized with negative VOT, though estimates of how often such "prevoicing" occurs vary greatly (e.g., 23% in Lisker and Abramson (1964) vs 62% in Smith (1978) for word-initial /b/ in isolated words; see Docherty (1992) for discussion) The negative VOT case is even more important for languages such as European French or Thai, where phonologically voiced stops are almost universally realized with prevoicing (Lisker and Abramson, 1964; Caramazza and Yeni-Komshian, 1974; Kessinger and Blumstein, 1997). Future work will deal with both the task of measuring negative VOT, and the task of deciding whether or not a burst occurred for a given stop. Both are necessary for our ultimate goal: an automatic measurement system that can take an arbitrary segment of speech and its orthographic transcription, and output a VOT measurement (positive, negative, or no burst) for each stop which is expected to occur.

The approach taken here combines knowledge about the cues human annotators use to measure VOT with machine-learning techniques for predicting structured output, to tailor an algorithm to measure VOT nearly as accurately as humans, and which meets the three criteria laid out in the introduction: accuracy, trainability, and robustness. Given suitable features and training data, it would be straightforward to extend the approach taken here to other widely measured phonetic variables where the output is a sequence of time points, such as vowel duration, segmenting a stop into different parts (closure, burst, frication), or the duration of vowel nasalization. More generally, for most phonetic quantities of interest (e.g., VOT, vowel formants, spectral measures for fricatives) measurement is a skilled task, and expert annotators usually use several types of cues (spectral, auditory, what the quantity "should" look like) in reaching a decision. The approach taken here is to supply these cues as features to an appropriate machine-learning procedure to learn how to annotate a particular quantity. Knowledge about the annotation task is also tied to the algorithm's structure by using a specialized cost function related to the quantity being annotated, which is directly optimized using discriminative training. The results of this paper suggest that combining knowledge about the annotation task to be performed with appropriate machine-learning techniques is a promising direction for designing algorithms to automate phonetic measurement in general.

ACKNOWLEDGMENTS

We thank Matt Goldrick and Nattalia Paterson for providing the PGWORDS data, Hugo van Hamme for providing the manual annotations used in Stouten and van Hamme (2009), and Chi-Yueh Lin for providing the list of stops used in Lin and Wang (2011). We also thank Karen Livescu and Matt Goldrick for helpful feedback, and Natalie Rothfels and Max Bane for VOT annotation. The first author was supported in part by a Department of Education GAANN grant.

[1] For the two larger datasets (TIMIT and PGWORDS), experiments for $M > 5$ took prohibitively long, and it was clear that performance worsened for $M > 2$.

[2] Our algorithm only considers portions of the speech signal in the training utterances from words beginning with initial voiceless stops; Lin and Wang's random forest detectors are trained using *all* of the training utterances. Thus, we believe the number of speakers whose utterances are used in training is not a good method for comparing the amount of training data used in the two approaches.

[3] For a dataset as large as the one considered here, $\hat{\beta} / \mathrm{SE}(\hat{\beta})$ should follow a standard normal distribution, making a Wald test appropriate (Hox, 2010).

Ali, A. (**1999**). "Auditory-based acoustic-phonetic signal processing for robust continuous speech recognition," Ph.D. thesis, Univ. of Pennsylvania, Philadelphia, PA.

Auzou, P., Ozsancak, C., Morris, R., Jan, M., Eustache, F., and Hannequin, D. (**2000**). "Voice onset time in aphasia, apraxia of speech and dysarthria: A review," Clin. Linguist. Phonet. **14**, 131–150.

Baayen, R., Davidson, D., and Bates, D. (**2008**). "Mixed-effects modeling with crossed random-effects for subjects and items," J. Mem. Lang. **59**, 390–412.

Bane, M., Graff, P., and Sonderegger, M. (**2012**). "Longitudinal phonetic variation in a closed system," in *Proc. of the 46th Chicago Ling. Soc.* (in press).

Bates, D., Maechler, M., and Bolker, B. (**2011**). *lme4: Linear mixed-effects models using S4 classes*, R package version 0.999375-40.

Brugnara, F., Falavigna, D., and Omologo, M. (**1993**). "Automatic segmentation and labeling of speech based on hidden markov models," Speech Commun. **12**, 357–370.

Caramazza, A., and Yeni-Komshian, G. (**1974**). "Voice onset time in two French dialects," J. Phon. **2**, 239–245.

Cho, T., and Ladefoged, P. (**1999**). "Variation and universals in VOT: Evidence from 18 languages," J. Phon. **27**, 207–229.

Cooper, A. (**1991**). "An articulatory account of aspiration in English," Ph.D. thesis, Yale University, New Haven, CT.

Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., and Singer, Y. (**2006**). "Online passive-aggressive algorithms," J. Mach. Learn. Res. **7**, 551–585.

Dekel, O., Keshet, J., and Singer, Y. (**2004**). "Large margin hierarchical classification," in *Proc. of the 21st ICML*, pp. 209–216.

Docherty, G. (**1992**). *The Timing of Voicing in British English Obstruents* (Foris, Berlin), pp. 29–32.

Fowler, C., Sramko, V., Ostry, D., Rowland, S., and Hallé, P. (**2008**). "Cross language phonetic influences on the speech of French-English bilinguals," J. Phon. **36**, 649–663.

Francis, A., Ciocca, V., and Yu, J. (**2003**). "Accuracy and variability of acoustic measures of voicing onset," J. Acoust. Soc. Am. **113**, 1025–1032.

Garofolo, J., Lamel, L., Fisher, W., Fiscus, J., Pallett, D., and Dahlgren, N. (**1993**). *TIMIT Acoustic-Phonetic Continuous Speech Corpus* (Linguistic Data Consortium, Philadelphia, PA).

Godfrey, J., and Holliman, E. (**1997**). *Switchboard-1 Release 2* (Linguistic Data Consortium, Philadelphia, PA).

Halberstadt, A. (**1998**). "Heterogeneous measurements and multiple classifiers for speech recognition," Ph.D. thesis, Mass. Inst. Technol., Cambridge, MA.

Hansen, J., Gray, S., and Kim, W. (**2010**). "Automatic voice onset time detection for unvoiced stops (/p/,/t/,/k/) with application to accent classification," Speech Commun. **52**, 777–789.

Hox, J. (**2010**). *Multilevel Analysis: Techniques and Applications* (Routledge, New York), Chap. 12.

Kazemzadeh, A., Tepperman, J., Silva, J., You, H., Lee, S., Alwan, A., and Narayanan, S. (**2006**). "Automatic detection of voice onset time contrasts for use in pronunciation assessment," in *Proc. of INTERSPEECH-2006*, pp. 721–724.

Keating, P., Byrd, D., Flemming, E., and Todaka, Y. (**1994**). "Phonetic analyses of word and segment variation using the TIMIT corpus of American English," Speech Commun. **14**, 131–142.

Keshet, J., Shalev-Shwartz, S., Singer, Y., and Chazan, D. (**2007**). "A large margin algorithm for speech-to-phoneme and music-to-score alignment," IEEE T. Audio Speech **15**, 2373–2382.

Kessinger, R., and Blumstein, S. (**1997**). "Effects of speaking rate on voice-onset time in Thai, French, and English," J. Phon. **25**, 143–168.

Lin, C., and Wang, H. (**2011**). "Automatic estimation of voice onset time for word-initial stops by applying random forest to onset detection," J. Acoust. Soc. Am. **130**, 514–525.

Lisker, L., and Abramson, A. (**1964**). "A cross-language study of voicing in initial stops: acoustical measurements," Word **20**, 384–422.

Niyogi, P., and Ramesh, P. (**1998**). "Incorporating voice onset time to improve letter recognition accuracies," in *Proc. of ICASSP-98*, pp. 13–16.

Niyogi, P., and Ramesh, P. (**2003**). "The voicing feature for stop consonants: Recognition experiments with continuously spoken alphabets," Speech Commun. **41**, 349–367.

Paterson, N. (**2011**). "Interactions in bilingual speech processing," Ph.D. thesis, Northwestern Univ., Evanston, IL.

Pinheiro, J., and Bates, D. (**2000**). *Mixed-Effects Models in S and S-PLUS* (Springer, New York), Chap. 2.

Randolph, M. (**1989**). "Syllable-based constraints on properties of English sounds," Ph.D. thesis, Mass. Inst. Technol., Cambridge, MA.

Shalev-Shwartz, S., Keshet, J., and Singer, Y. (**2004**). "Learning to align polyphonic music," in *Proc. of ISMIR-2004*, pap. 411.

Smith, B. (**1978**). "Effects of place of articulation and vowel environment on voiced stop consonant production," Glossa **12**, 163–175.

Sonderegger, M., and Keshet, J. (**2010**). "Automatic discriminative measurement of voice onset time," in *Proc. of INTERSPEECH-2010*, pp. 2242–2245.

Stouten, V., and van Hamme, H. (**2009**). "Automatic voice onset time estimation from reassignment spectra," Speech Commun. **51**, 1194–1205.

Talkin, D. (**1995**). "A robust algorithm for pitch tracking (RAPT)", in *Speech Coding and Synthesis*, edited by W. Kleijn and K. Paliwal (Elsevier, New York), pp. 495–518.

Taskar, B., Guestrin, C., and Koller, D. (**2003**). "Max-margin Markov networks," in *Proc. of NIPS 16*, pap. AA04.

Tauberer, J. (**2010**). "Learning [voice]," Ph.D. thesis, Univ. of Pennsylvania, Philadelphia, PA.

Tsochantaridis, I., Hofmann, T., Joachims, T., and Altun, Y. (**2004**). "Support vector machine-learning for interdependent and structured output spaces," in *Proc. of the 21st ICML*, pp. 104–112.

Yao, Y. (**2009**). "An exemplar-based approach to automatic burst detection in spontaneous speech," in *UC Berkeley Phonology Lab 2009 Annual Report* (UC Berkeley Phonology Laboratory, Berkeley, CA), pp. 13–28.