Point Process Models for Spotting Keywords in Continuous Speech

A. Jansen[*] and P. Niyogi[†]

September 19, 2008

Abstract

We investigate the hypothesis that the linguistic content underlying human speech may be coded in the pattern of timings of various acoustic "events " (landmarks) in the speech signal. This hypothesis is supported by several strands of research in the fields of linguistics, speech perception, and neuroscience. In this paper, we put these scientific motivations to the test by formulating a point process-based computational framework for the task of spotting keywords in continuous speech. We find that even with a noisy and extremely sparse, phone landmark-based point process representation, keywords can be spotted with accuracy levels comparable to recently studied hidden Markov model-based keyword spotting systems. We show that the performance of our keyword spotting system in the high precision regime is better predicted by the median duration of the keyword rather than simply the number of its constituent syllables or phonemes. When we are confronted with very few (in the extreme case, zero) examples of the keyword in question, we find that constructing a keyword detector from its component syllable detectors provides a viable approach.

1 Introduction

Investigating the speech recognition task of spotting predefined keywords in continuous speech has both practical and scientific motivations. Keyword spotting (KWS) is a technologically relevant problem, playing an important role in audio indexing and speech data mining applications. It is also a task that humans perform with astonishing ease, even in situations where little access to non-lexical linguistic constraints is provided (e.g. spotting native words in a unfamiliar language). Several computational approaches to this problem have been proposed (for a thorough review of the issues involved, see Rose, 1996):

1. One of the first keyword spotting strategies, proposed by Bridle (1973), involved sliding a frame-based keyword template along the speech signal and using a

[*] Department of Computer Science, University of Chicago

[†] Departments of Computer Science and Statistics, University of Chicago.

nonlinear dynamic time warping algorithm to efficiently search for a match. While the word models in later approaches changed significantly, this sliding model strategy was used in other approaches (see Wilpon et al., 1989; Silaghi and Bourlard, 2000).

2. A standard hidden Markov model (HMM) based method is the keyword-filler model. In this case, an HMM is constructed from three components: a keyword model, a background model, and a filler model. The keyword model is tied to the filler model, which is typically a phone or broad class loop, meant to represent the non-keyword portions of the speech signal. Finally, the background model is used to normalize keyword model scores. A Viterbi decode of a speech signal is performed using this keyword-filler HMM, producing predictions when the keyword occurs. Variations of this approach are provided by Wilpon et al. (1990); Hofstetter and Rose (1992); Szöke et al. (2005); and Ma and Lee (2007).

3. Another common approach is to perform a search through the phone or word lattice of a large vocabulary speech recognizer to spot keyword occurrences. The main research effort is focused on defining specialized confidence measures that maximize performance. Examples include James and Young (1994); Weintraub (1995); Junkawitsch et al. (1996); and Thambiratnam and Sridharan (2005). While these systems do not require a predefined vocabulary, they rely on language modelling and are thus highly tuned to the training environment.

In this paper, we operate within the sliding model paradigm, and thus do not need to explicitly account for the filler regions. Furthermore, our keyword models are not based on dynamic time warping or HMMs operating on a frame-based representation; instead, we consider keyword and background point process modelling of a sparse, event-based representation of the speech signal. Our motivation for considering such a representation may be traced to several scientific traditions.

First, acoustic phonetics and the study of speech production (for example, see Stevens, 1998) has provided the insight that speech is generated by the movement of independent articulators that produce acoustic signatures at specific points in time. These include the point of greatest sonority within a syllabic nucleus, the points of closure and release associated with various articulatory movements such as closure-burst transitions for stop consonants; obstruent-sonorant transitions; and onsets and offsets of nasal coupling, frication, or voicing. Linguistic information is coded both in terms of which events occur and the durations between these events (e.g. voice onset time). Stevens (2002) refers to these events as acoustic landmarks and assigns them a central status in lexical decoding.

Second, many neuroethological studies have demonstrated the existence of neurons that fire selectively when a constellation of acoustic properties are present in the stimulus (see Margoliash and Fortune, 1992; Esser et al., 1997; Fuzessery and Feng, 1983). In conjunction with these results is the synchronization hypothesis that auditory information is further encoded in the temporal pattern of such neural activity, i.e., temporal coding. See Suga (2006) for one articulation of these ideas.

If one takes seriously these findings, then it appears that the linguistic information may be coded in terms of the patterns of points in time — acoustic events in the speech signal, and neural firing patterns in the brain. In the context of keyword spotting, there are two guiding design principles for such a point process representation. First, this representation should efficiently encode the underlying linguistic content and produce as sparse a set of events as possible. Second, given a suitable point process statistical model, the temporal point patterns within instances of the keyword should be distinguishable from the background arrival rates. Since our representation is not frame based, we are led to a different statistical formalism to model the timing patterns of acoustic events. The framework of point processes is natural and we explore the applicability of such models in this paper (see Brown (2005) or Chi et al. (2007) for reviews and applications of point process models to neural spike pattern detection; as far as we know explicit applications to speech using such models did not exist before our own work).

Finally, we are interested not only in the case when training data is abundant, but also in the case where we have extremely limited access to examples of a particular keyword. Clearly, humans can easily spot a novel keyword in continuous speech after very limited exposure to others speaking it. This intuition implies that building keyword detectors from lower-level primitives may be a useful strategy (the lattice search methods implicitly take this point of view, as well). Indeed, the principle of compositionality (see Geman et al. (2002)) manifests itself in the observation that words are composed of syllables, and syllables themselves of phonemes. The underlying intuition is that although we may have very few examples of the word in question, we may have many more examples of the syllables that compose it. We test this experimentally.

Formally, our keyword spotting task amounts to learning a function of time $d_w(t)$ that takes high values when keyword w occurs and low values otherwise. This detector function, $d_w(t)$, may be defined in terms of the (log) likelihood ratio

$$d_w(t) = \log\left[\frac{P(O|\theta_w(t)=1)}{P(O|\theta_w(t)=0)}\right], \tag{1}$$

where O are the set of observations in the utterance and $\theta_w : \mathbb{R} \to \{0,1\}$ is an indicator function of time that takes the value 1 when the word is uttered[1] and 0 otherwise. The detector output is thresholded by a suitable value γ, and the local maxima that remain define a set of keyword detection times. To account for the variation in duration across instances of a particular keyword, we can introduce a nuisance duration parameter T, in which case the likelihood of the observations given $\theta_w(t) = 1$ takes the form

$$P(O|\theta_w(t)=1) = \int_0^t P(O|T, \theta_w(t)=1)P(T|\theta_w(t)=1)dT, \tag{2}$$

where we have imposed the constraint that $T < t$. In general, the models for $P(O|\theta_w(t)=0)$ and $P(O|T,\theta_w(t)=1)$ will depend largely on the nature of the observation space. However, the distribution over keyword durations, $P(T|\theta_w(t)=1)$, can be estimated directly from a set of keyword training instances.

[1] For the most part, we assume that $\theta_w(t)$ is 1 at the end of the word. However, one may take other appropriate definitions of θ_w and in particular, some of our experiments are done with θ_w taking the value 1 at the maximally sonorant point at the center of the vowel bearing primary stress in the keyword.

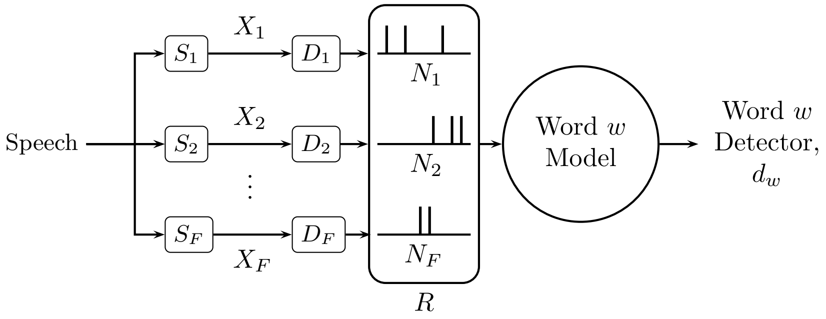


Figure 1: Architecture of our keyword spotting framework. In general, we may construct one signal processor S_i for each acoustic feature of interest ($F = |\mathcal{F}|$), which produces a specialized vector representation X_i. Each representation is input to a feature detector D_i that produces a temporal point pattern N_i. The combined set R of point patterns for all of the detectors analyzed compared with a statistical model for a given keyword w, determining the keyword detector output for the presented utterance.

As indicated above, we define our observations O to be a family of temporal point patterns, $R = \{N_\phi\}_{\phi\in\mathcal{F}}$, produced by detectors for a set $\mathcal{F}$ of linguistic properties (e.g. phones, distinctive features) or acoustic signatures (e.g. band energy inflection points, periodicity maxima).[2] For each $\phi \in \mathcal{F}$, the point pattern N_ϕ is specified by a collection of time points $\{t_1, \ldots, t_{n_\phi}\}$, where $t_i \in \mathbb{R}^+$. Arrivals of each process, which may be viewed as acoustic event landmarks, should ideally occur when and only when the corresponding feature ϕ is maximally expressed, articulated and/or most perceptually salient. Furthermore, asynchronous detectors imply that the quantization of arrivals of each feature's point process may vary. In practice, creating an ideal detector is of course unachievable, so we may generalize this notion to marked point process representations, (N_ϕ, M_ϕ), where the marks $M_\phi = \{f_1, \ldots, f_{n_\phi}\}$ are interpreted as the strengths (e.g. probabilities) of the corresponding landmarks. Figure 1 shows a schematic of our keyword spotting architecture.

The final ingredient of our keyword spotting strategy is the statistical model itself, which amounts to specifying the form of the terms $P(O|\theta_w(t) = 0)$ and $P(O|T, \theta_w(t) = 1)$ in Equations 1 and 2, where we take $O = R$. Outside the given keyword, we assume a constant arrival probability for each detector, and thus the background spike trains are modelled as homogeneous Poisson processes. However, for the keyword observations, for which we expect a characteristic spike pattern, we consider

[2] The design of a suitable family of detectors is itself the subject of an interesting program of research (see Stevens and Blumstein, 1981; Stevens, 2002; Niyogi and Sondhi, 2002; Pruthi and Espy-Wilson, 2004; Amit et al., 2005; Xie and Niyogi, 2006)). However, we will not explore this question in any detail here. Rather, we will assume that a detector based representation is made available to us and models for recognition will have to be constructed from such representations. In the experiments presented in this paper, we choose a simple phone-based detector set, which we define in Section 4.1.

inhomogeneous Poisson process modelling. This approach has previously been shown to be a useful means to describe the statistics of our sparse point process representation in the context of phone recognition (Jansen and Niyogi, 2008). In the present study, however, we instead consider Poisson process models of keywords and their constituent syllables rather than models of regions of constant sonority.

In Sections 2 and 3 we provide a formal treatment of the models employed, both in the case of many keyword training examples, as well as in the data-starved regime where we have few or no examples of the keyword itself (but have access to the constituent syllables). In Section 4 we present sets of toy and comprehensive keyword spotting experiments that demonstrate the viability of our approach. These results lead us to several interesting conclusions regarding both our proposed framework and the task of keyword spotting in general.

2 Learning with Several Keyword Examples

If we are provided a collection of training utterances, including several instances of a keyword w, we may proceed by estimating the distributions of Equations 1 and 2. While the distribution over keyword durations may be estimated directly, the form of the distributions over the observations will depend on the nature of implemented representation. In particular, if our set of observations O are taken to be the point process representation R introduced above, we consider homogeneous and inhomogeneous Poisson process models for the background and keyword w observations, respectively. Thus, we begin with a brief presentation of the theory of Poisson processes.

2.1 The Theory of Poisson Processes

A homogeneous Poisson process is founded on the assumption that in any differential time interval, dt, the probability of an arrival is λdt, where $\lambda \in \mathbb{R}^+$ is a constant rate parameter. This probability is independent of spiking history and hence the Poisson process is memoryless. For the inhomogeneous case, the constant rate parameter is generalized to a time-dependent function $\lambda(t)$, but the memoryless property still holds.

2.1.1 Homogeneous Poisson Processes

Consider a collection of independent point patterns $R = \{N_\phi\}_{\phi \in \mathcal{F}}$, where $N_\phi = \{t_1, \ldots, t_{n_\phi}\}$ and $t_i \in (0, T]$. If $\eta_\phi(t) \equiv |\{t_i \in N_\phi | t_i \leq t\}|$ is the number of time points in the interval $(0, t]$ and we assume N_ϕ is generated by a homogeneous Poisson process with rate parameter λ_ϕ, we may write

$$\mathbb{P}_{a,b}(k) \equiv \mathbb{P}[\eta_\phi(b) - \eta_\phi(a) = k] = \frac{(\lambda_\phi \tau)^k e^{-\lambda_\phi \tau}}{k!},$$

where $\tau = b - a$. It follows the probability that the first arrival occurs *after* time t is $\mathbb{P}[t_1 > t] = P_{0,t}(0) = e^{-\lambda_\phi t}$. Therefore, the probability that the first arrival lies in the interval $(t, t + dt]$ is $\mathbb{P}[t_1 \in (t, t + dt]] = \lambda_\phi e^{-\lambda_\phi t} dt$, which leads to a corresponding density function

$$f(t) = \lambda_\phi e^{-\lambda_\phi t}.$$

Since the process is memoryless, the likelihood[3] of the ϕ point pattern becomes

$$P(N_\phi) = \mathbb{P}_{t_{n_\phi},T}(0) \times f(t_1) \prod_{i=2}^{n_\phi} f(t_i - t_{i-1}) = (\lambda_\phi)^{n_\phi} e^{-\lambda_\phi T}. \tag{3}$$

It follows that the likelihood of the entire point processes observation $R = \{N_\phi\}_{\phi \in \mathcal{F}}$ takes the form

$$P(R) = \prod_{\phi \in \mathcal{F}} (\lambda_\phi)^{n_\phi} e^{-\lambda_\phi T}. \tag{4}$$

Training this homogeneous Poisson process model, then, amounts to estimating λ_ϕ for each $\phi \in \mathcal{F}$. In particular, if we are given N normalized-length training segments, and the total number K of landmarks of type ϕ observed in those segments, the maximum likelihood estimate of λ_ϕ is given by

$$\lambda_\phi^* = \arg\max_\lambda K \log \lambda - \lambda N T = K/NT. \tag{5}$$

2.1.2 Inhomogeneous Poisson Processes

In general, the inhomogeneous poisson process is characterized by the intensity function (rate parameter) $\lambda_\phi(t)$ which now varies as a function of time. One could consider many different forms for such a time varying intensity function. For simplicity, we consider a piecewise continuous rate parameter over D divisions of the interval $(0, T]$ given by $\lambda(t) = \lambda_d$ for $d = \text{ceiling}(t/\Delta T)$, where $\Delta T = T/D$. In this case, the Poisson process can be factored into D independent homogeneous processes operating in each division. That is, if

$$N_{\phi,d} \equiv N_\phi|_{I(d)},$$

where $I(d) = ((d-1)\Delta T, d\Delta T]$, and $n_{\phi,d} \equiv |N_{\phi,d}|$, then the likelihood of an individual point pattern is determined by

$$P(N_\phi) = \prod_{d=1}^{D} P(N_{\phi,d}),$$

where, according to Equation 3,

$$P(N_{\phi,d}) = (\lambda_{\phi,d})^{n_{\phi,d}} e^{-\lambda_{\phi,d} \Delta T}.$$

Here, $\lambda_{\phi,d}$ is defined as the rate parameter for the d^{th} homogeneous process for feature ϕ. It follows that the maximum likelihood estimate of the rate parameters are given by (*c.f.* Equation 5)

[3] Usually, we will use the notation P to denote likelihood of the data, i.e., the density evaluated at the data points. We use $\mathbb{P}(E)$ to denote the probability of the event E.

$$\lambda^*_{\phi,d} = K_{\phi,d} D/NT, \tag{6}$$

where we assume we have been provided with N training segments containing a total of $K_{\phi,d}$ landmarks for feature ϕ in the d^{th} segment piece. Finally, the likelihood of the whole point process representation can be computed as

$$P(R) = \prod_{\phi \in \mathcal{F}} \prod_{d=1}^{D} (\lambda_{\phi,d})^{n_{\phi,d}} e^{-\lambda_{\phi,d} \Delta T} \tag{7}$$

While we will not explore marked point process models in this paper, note that we can extend the Poisson approach described above by introducing mark-dependent rate parameters, $\lambda(t, f)$ (see Jansen and Niyogi, 2008).

2.2 Poisson Process Keyword Model

We assume there are two underlying stochastic processes generating the observed point process representation R. The first is a homogeneous Poisson process that generates the observations in regions outside keyword instances. The second is an inhomogeneous Poisson process that generates the various instances of the keyword.

Given a time t and a candidate keyword duration T, we can partition the point process representation R observed for an utterance of total duration L into three subsets: $R_1 = R|_{(0,t-T]}$, $R_2 = R|_{(t-T,t]}$, and $R_3 = R|_{(t,L]}$. We assume conditional independence between subsets and assume that R_1 and R_3 are generated by the homogeneous background process. Thus, we may write

$$P(R_1|T, \theta_w(t) = 1) P(R_3|T, \theta_w(t) = 1) = \frac{P(R|\theta_w(t) = 0)}{P(R_2|T, \theta_w(t) = 0)} \tag{8}$$

Since $P(R|\theta_w(t) = 0)$ does not depend on T, the detector function of Equation 1 ($O = R$) reduces to

$$d_w(t) = \log \left[\int_0^t \frac{P(R_2|T, \theta_w(t) = 1)}{P(R_2|T, \theta_w(t) = 0)} P(T|\theta_w(t) = 1) dT \right]. \tag{9}$$

Now, $P(R_2|T, \theta_w(t) = 0)$ is determined by the background homogeneous background model and $P(R_2|T, \theta_w(t) = 1)$ is determined by the inhomogeneous keyword model, as follows:

1. For the $P(R_2|T, \theta_w(t) = 1)$ distribution, we begin by normalizing $R_2 = R|_{(t-T,t]}$ to the interval $(0, 1]$; that is, we map R_2 to R_2' such that for each $t_i \in R_2$ there is a corresponding $t_i' \in R_2'$ where $t_i' = [t_i - (t - T)]/T$. Given this mapping, we make the simplifying assumption that

$$P(R_2|T, \theta_w(t) = 1) = \frac{1}{T^{|R_2|}} P(R_2'|\theta_w(t) = 1). \tag{10}$$

This equivalence assumes that the observations for each instance of the keyword are generated by a common, T-independent inhomogeneous Poisson process operating on the interval $(0, 1]$ that is subsequently scaled by T to a point pattern on the interval $(t - T, t]$. In this way, the number of firings of the different detectors in a keyword is invariant to the actual duration of the keyword. The duration of the keyword itself is modeled by a separate durational model. If we divide the normalized interval $(0, 1]$ into D regions, with $n_{\phi,d}$ landmarks of feature ϕ in the d^{th} division, from Equation 7 we may write

$$P(R_2' | \theta_w(t) = 1) = \prod_{\phi \in \mathcal{F}} \prod_{d=1}^{D} (\lambda_{\phi,d})^{n_{\phi,d}} e^{-\lambda_{\phi,d}/D}, \tag{11}$$

where $\lambda_{\phi,d}$ is the d^{th} division rate parameter for feature ϕ. Note that we have made use of the fact that the normalized interval has unit duration. Training the model for a given keyword amounts to computing the rate parameters of each feature detector over instances of the keyword (normalized to unit duration) according to Equation 6.

2. For the $P(R_2 | T, \theta_w(t) = 0)$ distribution, we need only consider a homogeneous Poisson process model that depends solely on the total number n_ϕ of landmarks observed for each feature ϕ and the total duration of the segment (in this case T). In particular, from Equation 4, we may write

$$P(R_2 | T, \theta_w(t) = 0) = \prod_{\phi \in \mathcal{F}} [\mu_\phi]^{n_\phi} e^{-\mu_\phi T}, \tag{12}$$

where μ_ϕ is the background rate parameter for feature ϕ.[4] Training this model for a given keyword amounts to computing the rate parameters as the average detector firing rates over a large collection of arbitrary speech (see Equation 5).

Given a novel utterance, we may evaluate the detector function by sliding a set of windows with durations $T \in \mathcal{T}$ (regularly-spaced with interval Δ) and approximating the integral expression of Equation 9 by

$$d_w(t) \approx \log \left[\sum_{T \in \mathcal{T}} \frac{P(R_2' | \theta_w(t) = 1) P(T | \theta_w(t) = 1) \Delta}{T^{|R_2|} P(R_2 | T, \theta_w(t) = 0)} \right], \tag{13}$$

where $P(R_2' | \theta_w(t) = 1)$ is given by Equation 11 and $P(R_2 | T, \theta_w(t) = 0)$ is given by Equation 12. Finally, the distribution $P(T | \theta_w(t) = 1)$ may be estimated from a set of measured keyword durations using kernel density estimation or any other non-parametric technique.

[4] It may be reasonable to expect the keyword duration T to scale inversely with the speaker's overall speaking rate. Therefore, the background rate parameters' T-independence makes the implicit assumption that the background detector firing rate is independent of speaking rate. Introducing a speaking rate nuisance parameter and estimating separate sets of background rate parameters for several speaker rate ranges may provide a more realistic background model. We leave this extension to future work.

2.3 Restricting the Search to Candidate Stressed Vowels

The above prescription for evaluating the detector function involves sliding a set of windows across the utterance and evaluating Equation 13. However, since we are ultimately interested in detecting a keyword that has a unique stressed vowel, we could conceivably limit our search to a set of candidate vowel locations, as determined by a vowel detector.

Let $V = \{t_1, \ldots, t_V\}$ be a set of times produced by such a vowel detector[5]. Then, we can define a modified detector function as

$$\hat{d}_w(t) = \begin{cases} d_w(t) & \text{if } t \in V \\ -\infty & \text{o/w} \end{cases}, \tag{14}$$

where $d_w(t)$ is defined by Equation 1. However, there is one caveat: the analysis window is now relative to the position of the stressed vowel, which is not invariant across instances of the keyword. To address this, we must introduce a second nuisance parameter $f \in [0, 1]$, the fraction of the word duration that occurs before the stressed vowel landmark. This requires two modifications to the above formulation:

1. The $\theta_w(t) = 1$ likelihood function of Equation 2 now takes the double integral form

   $$\begin{aligned} & P(O|\theta_w(t) = 1) \\ & \quad = \int_0^t \int_0^1 P(O|T, f, \theta_w(t) = 1)P(f|\theta_w(t) = 1)P(T|\theta_w(t) = 1)df dT, \end{aligned}$$

   where we have assumed conditional independence of T and f.

2. Since the evaluation of the detector function at time t is now relative to the stressed vowel position, the candidate keyword interval is given by $(t - fT, t + (1 - f)T]$. Thus, the point process representation restricted to the candidate keyword interval is now defined as $R_2 = R|_{(t-fT,t+(1-f)T]}$.

Given these adjustments, the derivation of Section 2.2 remains otherwise unchanged. Thus, given a set of candidate stressed vowel landmark positions $f \in \mathcal{W}$ (regularly-spaced with interval Δf), the sliding model detector function of Equation 13 now takes the form

$$d_w(t) \approx \log \left[\sum_{T \in \mathcal{T}} \sum_{f \in \mathcal{W}} \mathcal{I}(R_2, T, f) \Delta T \Delta f \right], \tag{15}$$

where

[5] A clarification is useful here. For the ith vowel, the time t_i is identified with a "vowel landmark" which corresponds to the point of maximal sonority (whose acoustic correlates can be measured through energy, periodicity, etc.) within that vowel. Such a vowel detector has been implemented in Xie and Niyogi (2006) for example.

$$\mathcal{I}(R_2, T, f) \equiv \frac{P(R_2'|\theta_w(t) = 1)P(f|\theta_w(t) = 1)P(T|\theta_w(t) = 1)}{T^{|R_2|}P(R_2|f, T, \theta_w(t) = 0)}. \tag{16}$$

Again, the forms of $P(R_2'|\theta_w(t) = 1)$ and $P(R_2|f, T, \theta_w(t) = 0)$ are given by Equations 11 and 12, respectively. The distribution $P(f|\theta_w(t) = 1)$ may be estimated from a set of keyword training instances using kernel density estimation.

Limiting our detector evaluation to a sparse set of vowel landmark times results in a speedup of $\frac{1}{\alpha}$, where α is the fraction of the sliding model time points that contain a vowel landmark. Thus, limiting the search to vowel landmarks can lead to significant reductions in processing times, especially in the setting of conversational speech where the speech content itself can be relatively sparse.

3 Learning with Minimal or No Keyword Examples

In the face of minimal or no training instances of the keyword of interest, we will not have adequate statistics to accurately estimate the distributions $P(O|T, \theta_w(t) = 1)$ and $P(T|\theta_w(t) = 1)$ of Equation 2. However, given a word is a sequence of syllables, and the most common syllables are shared among a plethora of words, we can reduce the keyword spotting task to syllable detection with a coincidence constraint. In this way, we may be able to build a detector for a word with no examples of the word but plenty of examples of the constituent syllables in question.

3.1 Poisson Process Syllable Model

Consider a keyword w composed of a sequence of syllables, $s_1 s_2 \ldots s_n$. If provided with adequate training examples of each syllable, we can construct a collection of syllable detector functions, $\{d_{s_i}\}_{i=1}^n$, in exactly the same manner used for keywords (see Section 2). Such syllable detectors will presumably function in a significantly more noisy manner than multisyllabic keyword detectors. However, we can combat this problem with the following strategy:

1. Determine a set of high-sensitivity syllable detector operating thresholds $\{\gamma_1, \gamma_2, \ldots, \gamma_n\}$, one for each syllable detector, in order to minimize false negatives.

2. Evaluate the syllable detectors at candidate vowel landmarks only (see Section 2.3).

3. Invoke the powerful constraint of detector coincidence (with delay) to integrate noisy syllable detectors and obtain relatively high keyword detection accuracy.

Formally, if each syllable detector d_{s_i} produces a set of candidate syllable detection times, $\mathcal{D}_i$, then, given a reasonable upper bound τ on the syllable duration, we may define the keyword w detector according to the Boolean function

$$d_w(t) = H_1(t, t) \wedge H_2(t_1, t_1 + \tau) \wedge \cdots \wedge H_n(t_{n-1}, t_{n-1} + \tau), \tag{17}$$

where

$$H_i(a,b) = \begin{cases} 1 & \text{if } \exists t \in \mathcal{D}_i \text{ s.t. } t \in [a,b] \\ 0 & \text{o/w} \end{cases}$$

and each $t_i \in \mathcal{D}_i$ is the detection for syllable s_i that satisfies H_i. Note that the Boolean function must be evaluated from left to right, as each H_j is a function of the $t_{j-1} \in \mathcal{D}_{j-1}$ that allows H_{j-1} to evaluate to one.

3.2 Sub-Syllable Models

The syllabary of English is made up of approximately 12,000 syllables, so collecting enough training examples of each to build a set of detectors for an arbitrary word can be practically infeasible. However, it is interesting to note that the most frequent 324 syllables in a typical speech corpus can cover two-thirds of it (Schweitzer and Möbius, 2004). To address rare syllables in our framework, one could fall back to a set of detectors for each constant sonority segment that comprise the rare syllable, in conjunction with an appropriately modified coincidence constraint. Nonetheless, we will limit the syllable model experiments in this paper to words comprised of common syllables only; exploring segmental detector-based strategies lies outside the scope of the current study.

4 Keyword Spotting Experiments

In this section, we consider the performance of the various models described above for the task of spotting instances of a given keyword in unconstrained speech. Each of our proposed algorithms employs a true detection paradigm, capable of spotting the keyword with minimal knowledge of the ambient linguistic environment. In particular, the background homogeneous Poisson process model is tantamount to a simple measurement of the mean firing rates of the detectors that form the point process representation.

Depending of the feature set $\mathcal{F}$, such mean firing rates can be largely independent of the environment, whether it be English, nonsense, or a foreign-language.[6] This is in contrast to other proposed KWS approaches that attempt to isolate keywords based on the probabilistic output of an HMM-based continuous speech recognizer. Such embedded keyword spotters rely on a more detailed model of the language, which may render them useless in the case of nonsense or foreign language background. Clearly, humans are adept at spotting native keywords in both nonsense speech and foreign languages, so we view this as a reasonable requirement for an automatic keyword spotter.

All experiments were conducted using the TIMIT (Garofolo et al., 1993) and Boston University radio news (Ostendorf et al., 1995) speech corpora. TIMIT was primarily used for training feature detectors that determine the point process representation, as well as for training and testing in the toy keyword spotting experiments

[6] For the phone-based detector set we implement, mean detector firing rates are roughly equivalent to a unigram phone language model. However, if lower-level detector sets are implemented (e.g. spectro-temporal features or band energy inflection points), our KWS strategy would have increased invariance to changing linguistic environments.

described below in Section 4.4. Boston University Radio News (BURadio) was used exclusively for large scale keyword spotting performance evaluations described below in 4.5. We begin with a precise description of the implemented point process representation, the vowel landmark detector, and keyword spotting evaluation procedures.

4.1 Construction of the Point Process Representation

We require a map from the speech signal $s(t)$ to a collection of point processes $R = \{N_\phi, M_\phi\}_{\phi\in\mathcal{F}}$, where $\mathcal{F}$ is some set acoustic or linguistic properties that is adequate to differentiate the phonological units in $\mathcal{P}$. This mapping is accomplished using the following three components:

1. Given W windows of the signal collected every Δ_ϕ seconds, construct for each $\phi \in \mathcal{F}$ an acoustic front end that produces a k_ϕ-dimensional vector representation $X_\phi = x_1, \ldots, x_W$, where $x_i \in \mathbb{R}^{k_\phi}$. Each representation X_ϕ should be capable of isolating frames in which feature ϕ is expressed and, to that end, the window and step sizes may be varied accordingly.

2. Construct a detector function $g_\phi : \mathbb{R}^{k_\phi} \rightarrow \mathbb{R}$ for each $\phi \in \mathcal{F}$ that takes high values when feature ϕ is expressed and low values otherwise. Each detector may be used to map X_ϕ to a detector time series $\{g_\phi(x_1), \ldots, g_\phi(x_W)\}$.

3. Given a threshold δ, we can compute the point process (N_ϕ, M_ϕ) for feature ϕ according to

$$\begin{aligned} N_\phi &= \{i\Delta_\phi | g_\phi(x_i) > \delta \text{ and } g_\phi(x_i) > g_\phi(x_{i\pm1})\} \\ M_\phi &= \{g_\phi(x_i) | i\Delta_\phi \in N_\phi\}. \end{aligned} \tag{18}$$

   Here, we assume $N_\phi = \{t_1, \ldots, t_{n_\phi}\}$ and $M_\phi = \{f_1, \ldots, f_{n_\phi}\}$ are ordered such that $t_{i+1} > t_i$ and $f_i = g_\phi(x_j)$, where $j = t_i/\Delta_\phi$.

In the experiments presented in this paper, we have taken our feature set $\mathcal{F}$ to be the set of phones $\mathcal{P}$ (i.e., there is a one-to-one correspondence between features $\phi \in \mathcal{F}$ and phones $p \in \mathcal{P}$). The precise structure of $\mathcal{P}$ is defined to be the standard 48 phone set of Lee and Hon (1989) and used in later work by Sha and Saul (2007). The definition of this set in terms of TIMIT labels is shown in Table 1. Since BURadio is not used in feature detector creation, the differences between TIMIT and BURadio phone labelling conventions are irrelevant.

While the point process representation can theoretically (and perhaps, ideally) be constructed from multiple acoustic representations tuned for each phonetic detector, we implemented a single shared front end for all of the phone detectors. In particular, we employed the rastamat package (Ellis, 2005) to compute a traditional 39-dimensional Mel-frequency cepstral coefficient (MFCC) feature set for 25 ms windows sampled every 10 ms. This included 13 cepstral coefficients computed over the full frequency

Table 1: The list of 48 phones used in our experiments and the corresponding TIMIT labels included for each (reproduced from Lee and Hon (1989)).

Phone	Example	Incl	Phone	Example	Incl
iy	b*ea*t		en	butt*on*	
ih	b*i*t		ng	si*ng*	eng
eh	b*e*t		ch	*ch*urch	
ae	b*a*t		jh	*j*udge	
ix	ros*e*s		dh	*th*ey	
ax	th*e*		b	*b*ob	
ah	b*u*tt		d	*d*ad	
uw	b*oo*t	ux	dx	bu*tt*er	
uh	b*oo*k		g	*g*ag	
ao	ab*ou*t		p	*p*op	
aa	c*o*t		t	*t*ot	
ey	b*ai*t		k	*k*ick	
ay	b*i*te		z	zoo	
oy	b*oy*		zh	mea*s*ure	
aw	b*ough*		v	*v*ery	
ow	b*oa*t		f	*f*ief	
l	*l*ed		th	*th*ief	
el	bott*le*		s	*s*is	
r	*r*ed		sh	*sh*oe	
y	*y*et		hh	*h*ay	hv
w	*w*et		cl (sil)	(unvoiced closure)	{p,t,k}cl
er	b*ir*d	axr	vcl (sil	(voiced closure)	{b,d,g}cl
m	*m*om	em	epi (sil)	(epenthetic closure)	epi
n	*n*on	nx	sil	(silence)	h#, pau

range (0-8 kHz), as well as 13 delta and 13 delta-delta (acceleration) coefficients. Cepstral mean subtraction was applied on the 13 original coefficients, and principal component diagonalization was subsequently performed for the resulting 39 dimensional vectors.

In general, the simplest approach to constructing the detector functions is to independently train a one-vs-all regressor for each phonological unit using any suitable machine learning method. That is, given L labelled MFCC training examples $\{(x_l, p_l\}_{l=1}^{L}$, where each $x_l \in \mathbb{R}^{39}$ is contained in a segment of phone $p_l \in \mathcal{P}$, we would like to compute a set of detector functions $g_p : \mathbb{R}^{39} \rightarrow [0, 1]$ such that $g_p(x) = P(p|x)$. In our implementation, we used the normalized MFCC vectors for each phone to estimate the $P(x|p)$ distributions assuming a C-component GMM for each $p \in \mathcal{P}$, given by

$$P(x|p) = \sum_{c=1}^{C} \omega_{pc} \mathcal{N}(\vec{\mu}_{pc}, \mathbf{\Sigma}_{pc})(x), \qquad (19)$$

where $\omega_{pc} > 0$ and $\sum_{c=1}^{C} \omega_{pc} = 1$ for each $p \in \mathcal{P}$; and $\mathcal{N}(\vec{\mu}, \mathbf{\Sigma})$ is a normal distribution with mean $\vec{\mu}$ and full covariance matrix $\mathbf{\Sigma}$. The maximum likelihood estimate of these GMM parameters are found using the expectation-maximization (EM) algorithm on the training data $\{(x_l, p_l)\}_{l=1}^{L}$. These distributions determine the family of detector functions, $\{g_p\}$, as

$$g_p(x) = P(p|x) = \frac{P(x|p)P(p)}{\sum_{p\in\mathcal{P}} P(x|p)P(p)}, \tag{20}$$

where $P(p)$ is the frame-level probability of phone p as computed from the training data. Note that for the toy experiments presented in Section 4.4, we measured performance for $C \in \{1, 2, 4, 8\}$ to study the performance for various detector reliabilities. For the large scale experiments presented in Section 4.5, we study just the $C = 8$ case.

Figure 2 shows for an example instance of the keyword “greasy” the evaluation of $\log P(x|p)$ and the corresponding point process representation after applying a threshold of $\delta = 0.5$.[7] The drastic reduction of information resulting from the conversion produces an exceedingly sparse point process representation. It is precisely this sparse representation that will be used in the the experiments presented below.

4.2 The Vowel Landmark Detector

For evaluating our keyword spotting strategies that operate on a vowel-by-vowel basis (see Sections 2.3 and 3), we require a vowel landmark detector. We can construct such a detector in much the same manner used for the individual phone detectors described in Section 4.1. In particular, given the GMM estimate of $P(x|p)$ for each $p \in \mathcal{P}$ (see Equation 19), we can define a detector for the set of vowels, $\mathcal{V} \subset \mathcal{P}$, as

$$g_{\mathcal{V}}(x) = P(\mathcal{V}|x) = \frac{\sum_{p\in\mathcal{V}} P(x|p)P(p)}{\sum_{p\in\mathcal{P}} P(x|p)P(p)}.$$

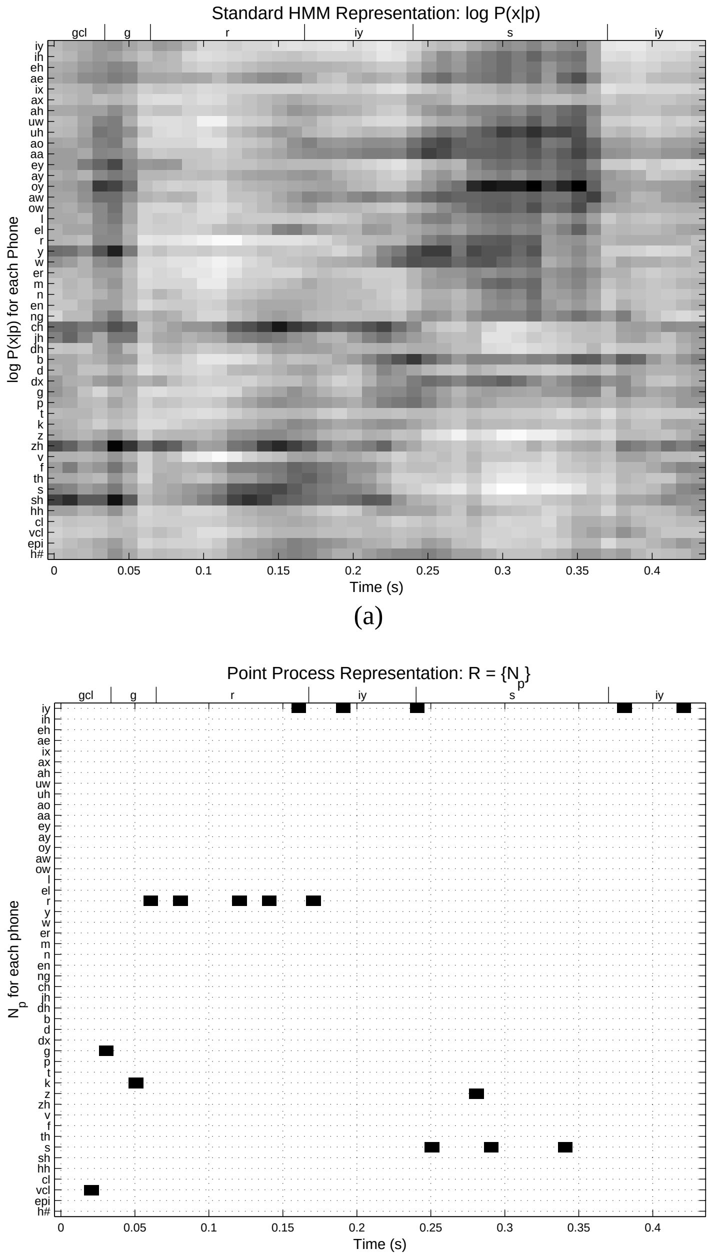
Finally, given a suitable threshold, we can use $g_{\mathcal{V}}$ to determine a set of candidate vowel locations according to Equation 18.

4.3 Model Evaluation Procedure

Both the TIMIT and BURadio corpora provide a time aligned word transcription. This transcription may be used to determine a set of intervals $I_w = \{[a_i, b_i]\}_{i=1}^{N_w}$ that contain the keyword w. While keyword spotting literature has relied on multiple performance metrics in the past, we employ a community standard figure of merit (FOM score) in our evaluation (for other examples of it use, see Junkawitsch et al., 1996; Ma and Lee, 2007, and the references therein). Given a set $\mathcal{D}_w$ of detections of keyword w, this figure of merit is defined as the mean detection rate when we allow $1, 2, \ldots, 10$ false positives per keyword hour. This metric is a means to summarize the high precision performance[8] of the detectors; this performance may be graphically characterized by

[7] This is an intuitive choice, as it corresponds to an optimal Bayes binary classification for each landmark (i.e., is the phone more likely present than not).

[8] It is worth noting that the ROC curve provides a complete characterization of the performance. In some applications, one may be interested in the high precision regime that that is captured by the FOM score described here. In other applications, however, the high recall regime may be of greater interest. For reasons that are not entirely clear to us, many recent papers on keyword spotting have provided the average FOM score and for ease of comparison, we have used this FOM score to evaluate our performance.



(a)

(b)

Figure 2: (a) The lattice of $\log P(x|p)$ values for an utterance of the word "greasy," where higher probability is lighter. (b) The corresponding (unmarked) point process representation, $R = \{N_p\}_{p \in \mathcal{P}}$ for $\delta = 0.5$.

the initial region of operating curves measuring the relationship between detection rates vs. false alarms per keyword, per hour, as threshold is varied.

In computing this figure of merit, we consider a keyword detection $t \in \mathcal{D}_w$ to be "correct" if there exists an interval $[a, b] \in I_w$ such that $t \in [a - \Delta, b + \Delta]$, where Δ is a short (e.g. 10-20 ms) tolerance that is set according to the precision of the word transcription. Any degenerate detections (i.e., multiple "correct" detections in a single occurrence of the keyword) are discarded.[9] Finally, any detection that is neither correct nor degenerate is considered to be a false alarm.

Our initial exploration of the keyword model behavior exposed a rather weak dependence of performance on the number of divisions D in the inhomogeneous Poisson process; thus, we chose not to perform an exhaustive validation procedure for each model. While not necessarily optimal, a setting of $D = 10$ led to reasonable performance early on, so we used that value for all experiments discussed in this paper (both whole word and syllable models). Ultimately, we believe any significant improvements in this department will result from implementing parameterized estimation of the inhomogeneous rate parameters, not from tweaking values of D.

Finally, for each whole word or syllable detector implemented, the training instances produced a sample of durations and fractional vowel positions. We used the sample means ($\langle T \rangle$ and $\langle f \rangle$) and standard deviations (σ_T and σ_f) to determine a set of values with which we compute Equations 13 and 15: $\mathcal{T} = \{\langle T \rangle + n\sigma_T\}$ for $n \in \{-1, 0, 1, 2\}$ and $\mathcal{W} = \{\langle f \rangle + m\sigma_f\}$ for $m \in \{-2, -1, 0, 1, 2\}$. Increasing the number of evaluation points produces slight performance improvements at the expense of longer run times.

4.4 "greasy" Experiments

Since we have constructed our point process representation using TIMIT phone data, it is useful to evaluate our keyword spotting strategy on TIMIT test data. Since the TIMIT corpus was designed to be phonetically and lexically diverse, the only content words with high frequency are those contained in the `sa` sentences that are spoken by all 630 speakers. Given this circumstance, we chose the word "greasy", which is contained in the `sa1` sentences, to evaluate performance of each proposed method as a function of both the number keyword training examples and the reliability of the detector set. It should be emphasized, however, that these are truly toy experiments, since every instance of "greasy" occurs in the same context. However, this also provides a control that allows us to isolate the speaker invariance of each approach as we provide fewer and fewer speakers from which we may learn the keyword model. Note that for each of the "greasy" performance values listed in this section, we tested on all `sa1`, `sx` (phonetically compact), and `si` (phonetically diverse) TIMIT test sentences, amounting to 1512 sentences totalling 1.31 hours of continuous speech.

[9] In practice, it is very easy to suppress such degenerates by simply discarding all but the highest probability detection in a small (of the order of the keyword duration) window around each candidate. In our experiments, this strategy removes *all* degenerate detections without reducing the overall detection rates.

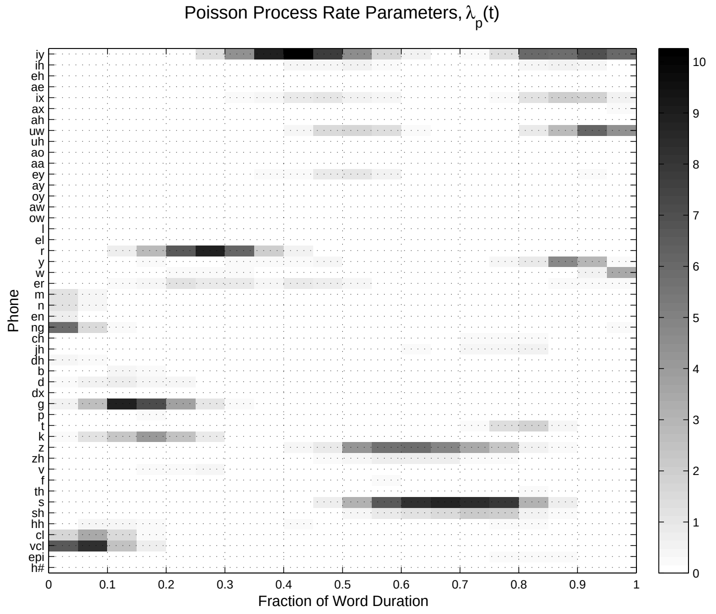


Figure 3: The inhomogeneous Poisson process rate parameters of each phone detector for the word "greasy", where we have set $D = 20$ (see Equation 11) and $C = 8$ (see Equation 19).

4.4.1 Keyword Model Performance

Figure 3 shows the Poisson process rate parameters for the keyword "greasy," trained on all 462 training instances of the keyword in the TIMIT `sa1` sentences. As expected we observe large rate parameters for the [vcl g r iy s/z iy] phone detectors as we pass through the word. However, also present are moderate rate parameters for the phones that are highly confusable with those actually present. For example, when the [g] detector exhibits a high firing rate, we observe a moderate [k] detector firing rate. Note that since the keyword occurs in the same context in each `sa1` sentence ("... in greasy wash ..."), significant rate parameters for the [ng] and [w] detectors at the beginning and end, respectively, become part of the word model. This is an artificial cue that may actually benefit performance in this toy setting.

Table 2 list the "greasy" keyword model figure-of-merit performance, using both sliding model and vowel landmark-based ("grea" is stressed) processing, as we vary the number of Gaussian mixture components used to construct the phone detectors (C in Equation 19). Each of the word models produce exceedingly reliable keyword spotters when compared with the average figure-of-merit values report for other systems. However, it is again important to note that the present experimental protocol is extremely artificial, so these high figure-of-merit values should be taken with a grain of salt (see the BURadio experiments in Section 4.5 for a more standard benchmark). Still, the controlled nature of our toy experiments allows us to make several strong conclusions about the nature of the model presented above:

1. The keyword spotting performance is remarkably stable as we decrease the number of mixture components used for the phone models. Reducing mixture components produces less reliable detectors and, accordingly, more detector confusions. The Poisson process model's robustness to this degradation is due in part to the fact that it models the behavior of each detector regardless of whether that detector is behaving poorly. In this way, a reliable false alarm is as useful as a true positive. In fact, this property makes it possible to use phone detectors trained using TIMIT for detecting keywords in BURadio data with mismatched acoustic conditions. Furthermore, if we instead implemented a non-phonetic detector set, the notion of feature detector-level false alarms would be inapplicable, but the keyword models could still produce reliable keyword spotters.

2. For all values of C, vowel landmark-based processing produces essentially equivalent performance compared with the exhaustive sliding model method. This indicates that the set of vowel landmarks contains all the relevant positions to look for the occurrences of a given keyword. The small improvements over the sliding model method for $C \in \{4, 8\}$ can be attributed to vowel-based methods reducing the opportunities for false alarms.

3. When testing the vowel landmark-based processing, we employ a high threshold of 0.95 for vowel detector described in Section 4.2 (i.e., we keep vowel landmarks that have $P(\mathcal{V}|x) > 0.95$). Since our goal is to evaluate the word models around stressed vowels only, all the landmarks of interest are sufficiently high probability. (This is not necessarily the case for syllable models; we will return to this point in Section 4.4.2.)

4. It is important to note that the vowel detector is defined by the same C-component phone GMMs used to construct the phone detector set. Thus, the drop in performance as we decrease C is in part due to missed vowel landmarks from the degraded vowel detector. This explains the slightly lower performance of the vowel landmark-based method relative to the sliding model method for $C \in \{1, 2\}$.

Figure 4 displays the dependence of the word model, using vowel landmark-based processing, on the number of training examples of the keyword.[10] More interestingly, since each speaker provides exactly one example, this plot may be interpreted as the figure-of-merit performance as we vary the number of *speakers* we learn the keyword from. This curve provides two important insights into the nature of the word model.

The first is that we observe remarkably stable performance, with the figure-of-merit dropping only ten points when we reduce the number of speakers from 462 to 25. It

[10] It is important to note that when we provide very few training examples, the performance of the word model depends on which particular training examples are used. Thus the figure-of-merit values displayed in Figure 4 are averages over several random selections of training examples for each number of training speakers value.

Table 2: Word-based model figure-of-merit performance for the keyword "greasy," using both the sliding model and vowel landmark methods to perform the search, with various settings of C, the number of GMM components used in each phone model.

	Sliding	Landmark
C	Method FOM (%)	Method FOM (%)
8	96.3	97.4
4	93.2	93.6
2	91.4	90.7
1	90.4	88.5

is important to re-emphasize here that the figure-of-merit is a measurement in the very high precision regime, making even the lower values admirable. For example, when we achieve a figure-of-merit of 80% when providing only 10 examples, it means that from 10 training speakers we can recognize 135 of the 168 distinct test speakers with next to no false alarms. Given the large variation in the TIMIT corpus across age, dialect region, and gender, this is no small feat. Even when providing only 5 training speakers, we still can detect the keywords of 123 test speakers.

With that being said, the second property to notice is that there is in fact a steep fall off in figure-of-merit performance below 50 training speakers. While this should be expected of any machine learning method, it begs the question of how many speakers would humans need to learn a word from before they could generalize to a much larger set. Indeed, it would seem reasonable that five or less would be all a human might require. However, it is clear from Figure 4 that such a small number is not adequate for the present whole word models.

4.4.2 Syllable-Based Model Performance

When we have access to zero or a very limited number of training instances of a given keyword, it becomes untenable to achieve a good estimate of the parameters of a Poisson process model of the entire keyword. As we saw in Figure 4, there is a significant drop in performance when we provide as few as five training examples, a trend which would surely continue as the number of training speakers continues to fall to zero. In this situation, however, we can fall back to the syllable-based keyword detectors presented in Section 3.

To construct a syllable-based model for the keyword "greasy", we constructed syllable detectors for "grea" and "sy". There is a fair amount of pronunciation variability, especially across dialect regions. Thus, the "grea" model included syllable pronunciations [vcl g r iy] and [vcl g r ix], while the "sy" model included [s iy], [z iy], [s ix], and [z ix]. Next, we set the syllable detector coincidence delay to 400 ms. Finally, since the syllable "sy" is unstressed, and the syllable detectors are processed on a vowel landmark basis, our vowel detector must be capable of detecting unstressed vowels in this case. We found this could be accomplished by simply reducing the vowel threshold from 0.95 to 0.5.

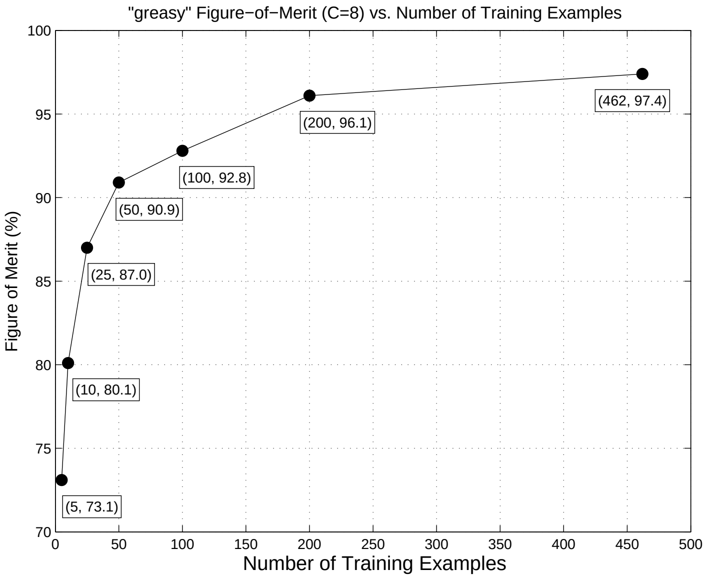


Figure 4: Figure-of-merit performance of the whole word model plotted against the number of training speakers (1 keyword example per speaker), used to construct the model.

Table 3 lists the syllable-based "greasy" detector performance for various phone detector reliabilities. Since the TIMIT corpus does not include a syllabic transcription, automatically distinguishing true instances of a particular syllable from the equivalent non-syllabic phonetic sequences is not possible without a syllabic dictionary. Hence, we performed two experiments; the first involved training the syllable models solely from the `sa1` sentences (the "`sa1` Only" column in Table 3), which produce exactly one *actual* syllable example per speaker. The second experiment involved training on all occurrences in the entire TIMIT training set of the phonetic sequences listed above for each syllable ("All TIMIT" column in Table 3). Again, there are many observations that can be made upon inspection of these results:

1. When we train the syllable models using examples culled from instances of the keyword only, we nearly match or slightly outperform the performance of the whole word model over various values of C. In this case, we are using exactly the same information from the speech signal, but modelling the constituent syllables separately. Since the two syllable detectors are significantly noisier than their whole word counterpart, this high keyword spotting performance demonstrates the power of a detector coincidence constraint for preventing false alarms.

2. When we instead train using all phone sequences in the TIMIT database that match the syllable of interest, we observe a significant drop in performance. However, it is important to remember that, in this case, we are (i) provided

Table 3: Syllable-based model keyword spotting figure-of-merit performance for the keyword "greasy" with various settings of C, the number of GMM components used in each phone model. The "`sa1` Only" results are from training only on actual syllables, which are all contained in the word greasy. The "All TIMIT" results are from models trained on the entire TIMIT train set, but include instances that are not true syllables.

C	`sa1` Only FOM (%)	All TIMIT FOM (%)
8	94.8	82.6
4	94.9	82.0
2	91.9	86.1
1	88.5	81.1

no training instances of the keyword and (ii) the syllable examples may occur in arbitrary contexts. Still, we outperform the whole word model performance with ten keyword training examples (see Figure 4). This means that it is possible to decompose the keyword spotting task into that of constituent syllable spotting, and still achieve good performance with no exposure to the keywords themselves. Thus, if it is possible to store and process an adequately large number of syllable detectors, this keyword spotting strategy provides a entirely point process-based path to lexical access.

3. Finally, it is important to note that (i) the syllable training instances in the "All TIMIT" case are not all actual syllables, and (ii) in the "`sa1` Only" case, all training instances are the appropriate syllables, but always occur in the context of the word. Thus, we believe the true performance of the proposed method, which would ideally be training on true syllable examples in arbitrary contexts, lies somewhere in between the two sets of experimental results shown in Table 3.

4.5 Boston University Radio News Experiments

While the toy experiments described above provide a controlled benchmark for internal comparison, we also require a large scale keyword spotting testbed that can be used to establish our performance relative to established systems. Unfortunately, there is currently no standard corpus or task defined by the speech recognition community for this purpose.[11] In the spirit of Ma and Lee (2007), who use the WSJ0 corpus, we test our system on the Boston University radio news corpus; both corpora consist of read newscaster style speech.

Like the TIMIT database, BURadio is clean, 16 kHz/16 bit read speech, making it a tolerable acoustic match for our phone detector set, which was trained in TIMIT as described above. Furthermore, the radio newscaster style is a natural but controlled style that minimizes the complication of major pronunciation variations, which the

[11] The one caveat to this statement is evaluation protocol used in the recent and ongoing NIST spoken term detection (STD) evaluation. While this protocol is similar to those used in word spotting experiments, the differences between KWS and STD tasks (STD is explicitly vocabulary independent) make the comparison with STD results meaningless for the time being.

Table 4: Keywords used in the BURadio experiments, along with the number of training/testing instances and the median duration (computed with training instances) for each keyword. For multi-syllabic keywords, the stressed syllable used is underlined.

Keyword	# Train	# Test	Median T
about	116	87	250 ms
Boston	272	122	470 ms
by	337	250	180 ms
city	41	54	320 ms
committee	41	37	380 ms
congress	13	16	550 ms
government	43	55	440 ms
hundred	121	98	310 ms
Massachusetts	334	102	710 ms
official	7	89	410 ms
percent	80	47	450 ms
president	52	33	490 ms
program	44	99	510 ms
public	68	122	340 ms
seven	39	60	370 ms
state	273	312	300 ms
thousand	56	54	490 ms
time	82	88	320 ms
year	144	163	230 ms
yesterday	90	53	550 ms

present models are not explicitly designed to accommodate.[12] BURadio consists of 7 speakers (4 males and 3 females), each reading on the order of one hour of speech for a total of 7.3 hours. We partitioned the speakers into a training group, consisting of the two males and two females (f1a,f3a,m1b,m2b), and a testing group of the remaining speakers (f2b,m3b,m4b).

Unlike the TIMIT database, the broadcast news content provides several multisyllabic words of relatively high frequency in arbitrary contexts. Table 4 lists the 20 keywords (18 content, 2 function) used in our experiments, along with the stressed vowels considered and number of occurrences in each division of the data. These words were chosen to cover a wide range of word complexities in both duration and numbers of phones and syllables.

4.5.1 Keyword Model Performance

Each BURadio keyword model is trained on all instances of the target word in the training group. Each keyword detector is evaluated on at least one hour of test group speech containing all of the instances of both the keyword and words that contain that

[12] One possible solution to this problem would be to employ a Poisson process mixture model, with the hope that each component would handle a given pronunciation of the keyword. We leave exploration of such a model to future work.

Table 5: Figure-of-merit performance for each keyword using the whole word Poisson process models.

Keyword	FOM (%)	Keyword	FOM (%)
Massachusetts	98.5	yesterday	59.6
program	97.9	government	52.4
Boston	89.3	city	45.6
president	83.0	hundred	34.1
thousand	78.9	year	33.1
congress	74.4	seven	31.3
percent	71.3	about	27.9
official	71.3	state	26.6
committee	66.1	time	25.7
public	60.1	by	9.0

Average FOM: 56.8%

keyword.[13] Table 5 lists the figure of merit performance using whole word models, for each keyword. Several insights emerge from this evaluation:

1. Our average figure-of-merit of 56.8% for whole word models is well within the range of other quoted values in the literature. While each prior study uses varying corpora, keyword complexities/durations, and acoustic model capacities, there are still several results that are relatively fair comparisons to our work. To list a few examples, each using a context-independent (monophone) acoustic model, Ma and Lee (2007) report an average figure-of-merit ranging from 42.6–61.5% and 18.4–33.1% for 30 content and 20 function keywords, respectively (WSJ0 corpus); Junkawitsch et al. (1996) report an average figure-of-merit of 58.5% for a set of one monosyllabic and 24 multisyllabic content keywords (Multicom 94.4 corpus); and, Szöke et al. (2005) report an average figure-of-merit of 47.7–64.5% for a set of 17 (mostly multisyllabic) content keywords in conversational speech (ICSI Meeting corpus).

   While our choice of keywords and corpus are not exactly matched to the studies listed above, we believe our approach of providing the individual performance of each keyword, along with numbers of training/testing examples and durational information, is the only reasonable manner to report performance. The large variation across keyword in Table 5, which is presumably present for all keyword spotting systems, attests to this need. However, very few studies of other proposed keyword spotting systems have taken their analysis beyond reporting an average figure-of-merit value across all keywords (Fernández et al., 2007, is one notable exception).

2. Upon inspection of the individual keyword figure-of-merit values, it is not immediately clear what word property—number of syllables, number of phones, or

[13] Note that care is taken to manage the imperfect correspondence between embedded keyword strings and embedded keyword utterances. For example, “timely” and “bipartisan” are treated as containing a positive examples of the keywords “time” and “by”, respectively; “sentiment” and “abysmal” are not

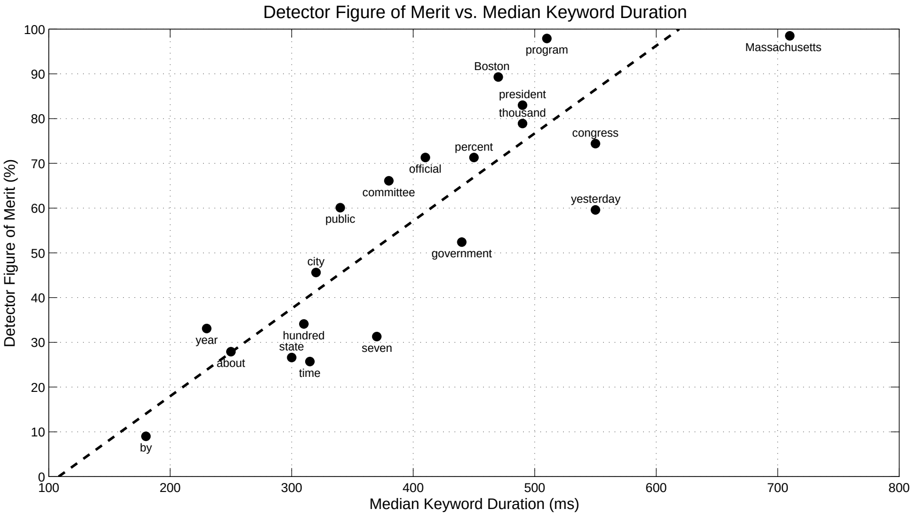


Figure 5: Figure-of-merit performance of the whole word models plotted against median duration for each of the 20 keywords. The dashed line shows the best linear fit to the data.

median duration—is the best predictor of performance. It turns out that the number of syllables is the least predictive, with a correlation coefficient of $r = 0.65$. Next is the number of phones (counting closure silences) with $r = 0.72$. While these correlations are significant, they are egregiously incompatible with a handful of keywords' performance. In particular, consider the keywords "hundred", which has one of the poorest performance levels, and "program", which has one of the best. These are both two-syllable words with the same number of constituent phones; however, the median duration of "program" is 65% longer than that of "hundred". Indeed, the correlation coefficient between figure-of-merit and median duration is $r = 0.86$, making duration the best single predictor of keyword spotting performance.

Figure 5 plots figure-of-merit vs. median keyword duration for each keyword, along with the best linear fit for the relationship. We attribute the remaining variance about this fit line to three second-order factors: (i) variations in individual phone detector performance exposed by variation of phonetic composition across keywords; (ii) variation in the number of training instances used to construct each keyword; and (iii) differing levels of pronunciation variability for each keyword. Interestingly, the relative hypoarticulation of function words did not cause a significant deviation from the linear relationship.

3. In listening to the various samples extracted by the word detector, the important role of pronunciation variability becomes clear, even in broadcast news data intended to minimize it. We believe it plays a major role in the linear relation-

ship between performance and median duration. This somewhat counterintuitive claim is motivated by the fact that if two pronunciations of the short word "year" differ by the particular vowel produced, that change accounts for roughly a third of the perceptual cues; it may therefore be more difficult to describe both pronunciations with a single keyword model. If the comparatively longer word "Massachusetts" had two pronunciations that differed by a single vowel, they would sound much more similar on the whole and thus would be easier to associate with one another using a single keyword model.

In fact, the significant variation in pronunciation of short keywords, especially function words, across contexts highlights the vital role higher level linguistic constraints must play in human word recognition. Our context independent sliding model method is using no such information, so it is not surprising many mistakes are made.[14] However, when a word is sufficiently long and/or phonetically rich, our system performs exceedingly well.

4.5.2 Syllable-Based Model Performance

Finally, we consider the syllable model based keyword spotting performance on the BURadio corpus. For the monosyllabic keywords, the whole word and syllable-based detectors are nearly equivalent, so there is no change in performance.[15] To get a flavor for the effect on multisyllabic keywords, we built syllable-based detectors for three of the better performing keywords: "program," "Boston," and "committee" When we trained on syllables embedded in the keywords only (c.f. the "`sa1` Only" results in Section 4.4.2), we measured figure-of-merit values of 94.0%, 87.9%, and 61.4% for "program," "Boston," and "committee," respectively. The relatively small drops from the whole word model performance listed in Table 5 are consistent with the behavior observed in the TIMIT "greasy" experiments described above.

The second experiment involved training the models on syllable instances *not contained* in the keywords of interest, when possible. Even though BURadio does not provide syllabic transcriptions for all speakers, we were able to isolate a significant number of true occurrences for some syllables by searching for an alternative set of words that also contain the syllables of interest. For example, to get examples of "pro" not contained in "program," we collected training instances from "protest, prohibit, probation, and protein." (The syllable "gram," however, was only contained in "program.") In this more realistic setting, we measured a figure-of-merit values of 90.6% for "program."

For the keyword "Boston," we were unable to collect sufficient number of syllable examples outside of the keyword to train on them alone; instead, we augmented the original training set with true examples contained in other words (for example, "boss" and "Washington"). This lead to a figure-of-merit performance of 86.4%. Lastly, since

[14] Consider spotting native keywords in a foreign language. In this setting, the listener's access to high level linguistic constraints is severely limited. Everyone's informal experience is that native keywords can be spotted fairly easily. However, it is clear that short words, especially one-syllable words, can be often be misheard throughout the foreign speech, if one is looking for them, when they have not actually occurred.

[15] Technically, the syllable model for a monosyllabic word could also be trained on instances embedded in other words. However, this expansion produce a negligible effect on the already low performance.

the keyword “committee” contains three very common syllables, we were able to construct syllable models solely from true instances contained in other words. This led to a figure-of-merit performance of 55.8%, representing a less than six point drop from that when we trained on syllable examples taken from the actual keyword.

5 Conclusions

We have shown that Poisson process modelling of a highly sparse phone-based point process representation is sufficient to spot keyword occurrences in continuous speech at performance levels comparable to other HMM-based methods. We have demonstrated that our system has the ability to generalize from a relatively small number of training speakers and is robust to the degradation of the phone detector set reliability. We also found that processing the speech signal on a vowel-by-vowel basis using landmarks is equivalent to an exhaustive sliding window search.

In extremely data starved regimes, where keyword instances are not available, we found that using constituent syllable detectors in conjunction with a delayed coincidence constraint is adequate to nearly reproduce the performance of whole word models constructed with several keyword examples. We believe this approach would likely work using other sliding window-based keyword spotting approaches as well. Moreover, this syllable-based strategy provides a computationally plausible path to event-based lexical access and dealing with out of vocabulary words quickly.

Finally, we have found that the figure-of-merit performance of our system is most highly correlated with median keyword duration. If this property is true of all keyword spotting systems, the parameters of this linear relationship (e.g. slope or y-intercept) may provide a keyword-independent performance metric, which would help normalize system benchmarking.

References

Yali Amit, Alexey Koloydenko, and Partha Niyogi. Robust acoustic object detection. *J. Acoust. Soc. Am*, 118(4), 2005.

J. S. Bridle. An efficient elastic-template method for detecting given words in running speech. In *Proc. of the Brit. Acoust. Soc. Meeting*, 1973.

Emery N. Brown. Theory of point processes for neural systems. In *Methods and Models in Neurophysics (Chow CC, Gutkin B, Hansel D, Meunier C, Dalibard J)*, chapter 14, pages 691–726. Elsevier, Paris, 2005.

Zhiyi Chi, Wei Wu, and Zach Haga. Template-based spike pattern identification with linear convolution and dynamic time warping. *J. Neurophysiology*, 97(2):1221–1235, 2007.

Daniel P. W. Ellis. PLP and RASTA (and MFCC, and inversion) in Matlab, 2005. URL `http://www.ee.columbia.edu/ dpwe/resources/matlab/rastamat/`. (online web resource).

Karl-Heinz Esser, Curtis J. Condon, Nobuo Suga, and Jagmeet S. Kanwal. Syntax processing by auditory cortical neurons in the FM-FM area of the mustached bat pteronotus parnellii. *Proc. Natl. Acad. Sci. USA*, 94:14019–14024, 1997.

Santiago Fernández, Alex Graves, and Jürgen Schmidhuber. An application of recurrent neural networks to discriminative keyword spotting. In *Lecture Notes in Computer Science: Artificial Neural Networks - ICANN 2007 (J. Marques de Sá et al.)*, pages 220–229. Springer, Berlin, 2007.

Zoltan M. Fuzessery and Albert S. Feng. Mating call selectivity in the thalamus and midbrain of the leopard frog (Rana p. pipiens): single and multiunit responses. *Journal of Comparitive Psychology*, 150:333–334, 1983.

John S. Garofolo, Lori F. Lamel, William M. Fisher, Jonathan G. Fiscus, David S. Pallett, Nancy L. Dahlgren, and Victor Zue. *TIMIT Acoustic-Phonetic Continuous Speech Corpus*. Linguistic Data Consortium, Philadelphia, PA, 1993.

S. Geman, D. F. Potter, and Z. Chi. Composition systems. *Quarterly of Applied Mathematics*, LX:707–736, 2002.

E. M. Hofstetter and R. C. Rose. Techniques for task independent word spotting in continuous speech messages. In *Proc. of ICASSP*, 1992.

D. A. James and S. J. Young. A fast lattice-based approach to vocabulary independent wordspotting. In *Proc. of ICASSP*, 1994.

Aren Jansen and Partha Niyogi. Point process models for event-based speech recognition. Technical Report TR-2008-04, U. of Chicago, Computer Science Dept., 2008.

J. Junkawitsch, L. Neubauer L, H. Hoege, and G. Ruske. A new keyword spotting algorithm with pre-calculated optimal thresholds. In *Proc. of ICSLP*, 1996.

Kai-Fu Lee and Hsiao-Wuen Hon. Speaker-independent phone recognition using hidden Markov models. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(11):1641–1648, 1989.

Chengyuan Ma and Chin-Hui Lee. A study on word detector design and knowledge-based pruning and rescoring. In *Proc. of Interspeech*, 2007.

Daniel Margoliash and Eric S. Fortune. Temporal and harmonic combination-sensitive neurons in the zebra finch's HVc. *Journal of Neuroscience*, 12:4309–4326, 1992.

Partha Niyogi and M. M. Sondhi. Detecting stop consonants in continuous speech. *J. Acoust. Soc. Am*, 111(2):1063–1076, 2002.

M. Ostendorf, P. J. Price, and S. Shattuck-Hufnagel. The Boston University Radio News Corpus. Technical Report ECS-95-001, Electrical, Computer and Systems Engineering Department, Boston University, 1995.

Tarun Pruthi and Carol Espy-Wilson. Acoustic parameters for automatic detection of nasal manner. *Speech Communication*, 43:225–239, 2004.

Richard C. Rose. Word spotting from continuous speech utterances. In *Automatic Speech and Speaker Recognition: Advanced Topics (Lee, Chin-Hui Lee, Frank K. Soong, and Kuldip K. Paliwal, Eds.)*, pages 303–329. Springer, Berlin, 1996.

Antje Schweitzer and Bernd Möbius. Exemplar-based production of prosody: Evidence from segment and syllable duration. In *Proc. of Speech Prosody*, 2004.

Fei Sha and Lawrence K. Saul. Comparison of large margin training to other discriminative methods for phonetic recognition by hidden Markov models. In *Proc. of ICASSP*, 2007.

M.-C. Silaghi and H. Bourlard. Iterative posterior-based keyword spotting without filler models. In *Proc. of ICASSP*, 2000.

K. N. Stevens. *Acoustic Phoenetics*. MIT Press, Cambridge, MA, 1998.

Kenneth N. Stevens. Toward a model for lexical access based on acoustic landmarks and distinctive features. *J. Acoust. Soc. Am*, 111(4):1872–1891, 2002.

Kenneth N. Stevens and Sheila E. Blumstein. The search for invariant acoustic correlates of phonetic features. In *Perspectives on the Study of Speech (P. Eimas and J. L. Miller)*, chapter 1, pages 1–38. Erlbaum, Hillsdale, NJ, 1981.

Nobuo Suga. Basic acoustic patterns and neural mechanisms shared by humans and animals for auditory perception. In *Listening to Speech: An Auditory Perspective (Steven Greenberg and William A. Ainsworth)*, pages 159–182. Lawrence Erlbaum Associcates, Mahwah, NJ, 2006.

Igor Szöke, Petr Schwarz, Pavel Matějka, Lukáš Burget, Martin Karafiát, and Jan Černocký. Phoneme based acoustics keyword spotting in informal continuous speech. In *Lecture Notes in Computer Science - TSD 2005 (V. Matousek et al.)*, pages 302–309. Springer-Verlag, Berlin, 2005.

K. Thambiratnam and S. Sridharan. Dynamic match phone-lattice searches for very fast and unrestricted vocabulary KWS. In *Proc. of ICASSP*, 2005.

M. Weintraub. LVSCR log-likelihood scoring for keyword spotting. In *Proc. of ICASSP*, 1995.

J. G. Wilpon, L. R. Rabiner, C.-H. Lee, and E. R. Goldman. Application of hidden Markov models for recognition of a limited set of words in unconstrained speech. In *Proc. of ICASSP*, 1989.

J. G. Wilpon, L. R. Rabiner, C.-H. Lee, and E. R. Goldman. Automatic recognition of keywords in unconstrained speech using hidden Markov models. *IEEE Transactions of Acoustic, Speech, and Signal Processing*, 38(11):1870–1878, 1990.

Zhimin Xie and Partha Niyogi. Robust acoustic-based syllable detection. In *Proc. of ICSLP*, 2006.