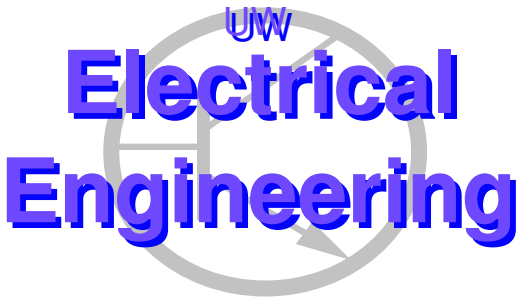
Graphical Models for Integrating Syllabic Information

Chris D. Bartels and Jeff A. Bilmes
{bartels,bilmes}@ee.washington.edu



UWEE Technical Report
Number UWEETR-2009-0007
July 2009

Department of Electrical Engineering
University of Washington
Box 352500
Seattle, Washington 98195-2500
PHN: (206) 543-2150
FAX: (206) 543-3842
URL: http://www.ee.washington.edu

Graphical Models for Integrating Syllabic Information

Chris D. Bartels and Jeff A. Bilmes
{bartels,bilmes}@ee.washington.edu

University of Washington, Dept. of EE, UWEETR-2009-0007

July 2009

Abstract

We present graphical models that enhance a speech recognizer with information about syllabic segmentations. The segmentations are specified by locations of syllable nuclei, and the graphical models are able to use these locations to specify a "soft" segmentation of the speech data. The graphs give improved discrimination between speech and noise when compared to a baseline model. When using locations derived from oracle information an overall improvement is given, and when the oracle syllable nuclei are augmented with information about lexical stress it gives additional improvements over locations alone.

1 Introduction

This article describes a set of graphical models that enhance a standard automatic speech recognition system with information about syllable locations. The location information comes in the form of estimated positions of syllable nuclei which are detected by locating peaks in neural network posteriors. The graphical models (GMs) count the number of detections in each hypothesized syllable, word, or utterance and the GM distributions include probabilities for detected counts given expected counts.[1]

There are a number of motivations for why one would want to use syllabic information in speech recognition. The first is as a basis for speech/noise discrimination. Burst noises occur frequently and in many cases automatic speech recognizers will incorrectly decode them as speech. This class of noise includes breath and microphone noise, cross-talk, car horns, and most any type of sound that may be unintentionally picked up by a microphone. These noises are neither speech nor silence, and many particular noises may never occur in the training data. Non-speech noise is typically not labeled in speech corpora and whenever a portion of the speech is labeled "silence" it may also include noise[2]. Speech consistently has syllable length modulations at 4 to 6 Hertz [Greenberg, 1999, Greenberg et al., 2006], and noise typically does not have this pattern. In this article it is shown that integrating information derived from syllable locations can improve a recognizer's ability to discriminate between speech and noise. Our model achieves 59% of the possible improvement between the baseline model and the same model with a silence oracle (Section 5.4).

The second motivation is that syllables provide acoustic information on the duration and segmentation of speech. Here it is hypothesized that segmentations based on syllable locations can add information that is missing from the standard duration model. Word durations in conventional automatic speech recognition (ASR) systems are determined by binary transition probabilities, a word insertion penalty (WIP), the number of phones in a given pronunciation, and the acoustic match of the feature vectors to the Gaussian mixture models (GMMs). Of these, only the GMMs make use of the acoustics of the utterance being decoded. In a typical system the transition probabilities and WIP do not change regardless of any change in the acoustics, variation in the rate of speech, or the length of the decoded words. The GMMs give frame-by-frame scores indicating how well they match the current acoustic vector, but typical systems do not analyze long time scale time windows specifically for segmentation or duration. The acoustic observation variables are independent of past and future observation variables given their corresponding state, so acoustic cues can only affect duration and segmentation via the scoring of individual sub-phone states. When longer time scale features

[1] Some of this paper appeared previously in the *2007 Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop* [Bartels and Bilmes, 2007], *2008 Proceedings of Interspeech* [Bartels and Bilmes, 2008], and Chapter 5 of [Bartels, 2008].

[2] Examples of noise during silence regions from Switchboard-I [Godfrey et al., 1992] using the segmentations given in [Deshmukh et al., 1998] are sw2474A-0050 and sw2614B-0029

are used (such as [Hermansky and Sharma, 1998]), they are often appended to the standard observation vector and, again, can only change the segmentation via the acoustic match to a sub-phone state. The transition probabilities have a small dynamic range (relative to multi-dimensional Gaussians) and no memory of how long the model has been in the current state. An acoustic match can cause the GMMs to overwhelm the scores given by the transition probabilities, even if this match has an unreasonably short duration. This is one explanation for why the WIP is needed. Without it most recognizers would incorrectly insert a large number of words, many of which would have unrealistically short durations. The models presented in this article use a syllable based segmentation that is derived from a long term analysis of the decoded utterance acoustics. These models integrate syllable nuclei detections into a GM based recognizer to specify a "soft" segmentation of the speech data. When oracle information is used to derive the number and approximate locations of the nuclei it gives a 21% relative improvement in word error rate (Section 5.4).

The next motivation is that syllable stress patterns contain useful and potentially non-redundant information. Lexical stress can be used to disambiguate between words that are phonetically identical. Examples are the noun "PUR-fect" versus the verb "per-FECT" or "com-BINE" meaning to bring together versus "COM-bine" referring to a farm implement [Ying et al., 1996]. More commonly, words are not phonetically identical but have enough similarity that they are difficult to differentiate due to noise or pronunciation variation. For example, the words "campus" and "com-pose" phonetically differ only in the vowels but they can be differentiated by syllable stress [Aull and Zue, 1985]. In this article it is shown that when the oracle syllable nuclei information is augmented with information about lexical stress it gives a 16% relative improvement over locations alone and a 31% improvement over the baseline (Section 5.7).

The final motivation is that perceptual experiments have given evidence that human speech recognition depends on syllable-length modulations. Specifically, human recognition rates degrade when syllable frequency modulations are suppressed from the speech signal [Drullman et al., 1994, Arai et al., 1996, Greenberg et al., 1998, Greenberg and Arai, 2004]. From this it can be argued that humans make use of information at time lengths that are too long to be captured using only a phonetic representation.

Syllables have been applied to ASR by numerous other authors. The next few paragraphs will describe the previously published work most closely related to our results, and an extensive discussion of related work is given in Section 6. The first piece of closesly research was presented in Wu et al. [1997], Wu [1998]. In that work, syllable onsets were detected by a neural network classifier, and this information was then used to prune away hypotheses in a lattice that were not consistent with the detections. They achieved a 38% improvement in word error rate using oracle onset information and a 10% improvement using estimated onset information. The first advantage of the models presented here over the previous work is the degree to which the syllable segmentation information is provided to the recognizer in a "soft" manner. The previous methods incorporated "soft" decisions in that they did not prune word hypotheses that were within a fixed time window of the syllable onsets, as opposed to requiring consistency at the resolution of a single frame. Here, the graphical model allows the syllabic information to be used to modify all active hypothesis scores for each syllable detection. Another advantage is that the experiments here make use of approximate nucleus locations, whereas the previous work required syllable boundaries which are more difficult to define. Also, the issue of speech/noise discrimination was not addressed in Wu et al. [1997], Wu [1998].

The work in Çetin [2004], Çetin and Ostendorf [2005] also has a significant relation to the work presented in this article. There, a graphical model was used to integrate phone and syllable models. The phonetic portion of the model was a standard hidden Markov model using within-word triphones and Gaussian mixtures. The syllable portion of the model used a hidden variable sequence, but instead of a sequence of triphones the sequence represented onset, coda, ambisyllabic consonant, unstressed vowel, stressed vowel, silence, and non-speech sounds. HATS features [Chen et al., 2004] and Gaussian mixture model output distributions were used to discriminate between these classes. The GM combined these models in a way that allowed the phonetic triphone mixtures contributed to the hypothesis score in every frame, and the syllable-state score contributed at a downsampled rate. This rate was fixed in some experiments and variable for others. In the experiments presented here, there is not a second set of GMMs that classify different portions of the syllable. Our features are discrete detections that segment the signal by looking for peaks in a neural network posterior. These features have the goal of segmenting the speech rather than performing a classification. The word hypothesis scores are altered by the probability that, if the hypothesis were correct, one would see the given estimated segmentations. Again, speech/noise discrimination was not discussed in Çetin [2004], Çetin and Ostendorf [2005].

The rest of this article is organized as follows. The syllable detection method is described in Section 2 and the novel GMs are described in Section 3. Section 4 gives the corpora used for experimentation, and this is followed in Section 5 by experiments and results. Section 6 contrasts the models presented here to previous work, Section 7 provides a discussion, Section 8 gives directions for future work, and a synopsis is provided in Section 9.

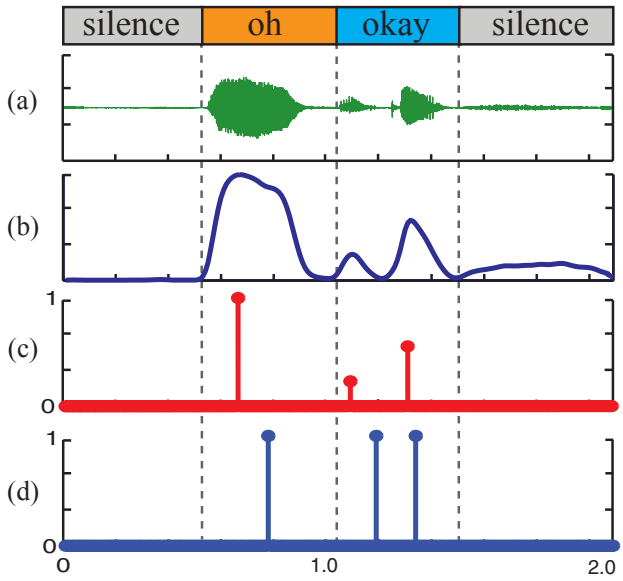


Figure 1: Illustration of syllable nuclei features. (a) acoustic waveform, (b) a measure of how "vowel-like" the sound is, (c) local maxima in the measure are the estimated syllable nuclei (d) word aligned (W.A.) oracle features; these are binary features evenly spaced within the word boundary. (This example was generated by the method described in [Wang and Narayanan, 2005] for use in [Bartels and Bilmes, 2007], and maxima in the silence and unvoiced regions are removed. The neural-network based method used in the results here gives similar examples.)

2 Detecting Syllables

This section describes the syllable detction method that is used in this article. This method produces a set of features that are utilized by the graphical models that we will present in Section 3.

English syllables are typically defined by an onset, nucleus, and coda. In the definition used here the nucleus is always a vowel, the onset is composed of one or more consonants preceding the vowel, and the coda is one more consonants following the vowel. All syllables have a nucleus, but many do not have an onset and/or a coda.

Syllables are often described as having a sonorancy hierarchy [Fujimura, 1975, Huang et al., 2001]. The nucleus is the most sonorant point in the syllable. It has more energy and more resonance in the vocal tract than the other portions the syllable. The most sonorant consonants in the onset and coda directly surround the nucleus. Consonants become decreasingly sonorant as one looks towards the beginning and end of the syllable. This implies that syllables start with low energy and a constricted vocal tract, then rise in energy and resonance, and finally decrease in energy and resonance as the vocal tract again becomes constricted. This rising and falling of energy and sonorance is perceived by humans as a temporal modulation and defines the "rhythm" of speech. The rhythm and locations of syllables can be automatically detected to some degree of accuracy without any explicit recognition of the word identities or phonetic content [Mermelstein, 1975, Shastri et al., 1999, Wang and Narayanan, 2007, Section 6.3].

Our syllable nuclei detection method uses the posterior of an artificial neural network (ANN) trained to detect vowels as a measure of sonorance, and it interprets the peaks in this posterior as syllable nuclei. The peak finding process is illustrated in Figure 1[3]. These detections are translated into binary, discrete features that equal 1 in the frames corresponding to a peak in the posterior and equal 0 in all frames that are not peaks. Some experiments also make use of syllable nuclei detections that are derived from oracle information. Additional details of the feature creation process are given in Section 5.1.

[3]Figure 1 was generated by the method described in [Wang and Narayanan, 2005] for use in [Bartels and Bilmes, 2007]. The neural-network based method used in the results here gives similar examples.

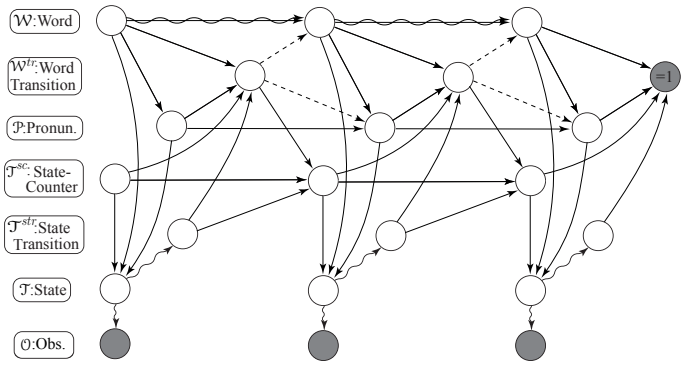


Figure 2: Baseline Model [Zweig et al., 2002, Livescu et al., 2007]. This is a standard speech HMM represented as a DBN. Hidden variables are white while observed variables are shaded. Straight arrows represent deterministic relationships, curvy arrows represent probabilistic relationships, and dashed arrows are switching relationships. Switching parents can be used to specify context specific independence which can speed inference in some cases [Geiger and Heckerman, 1996] and can also be a convenience for the graph designer. Figure appeared in [Bartels and Bilmes, 2007]

3 Graphs Using Syllable Nuclei

The syllable detection features are incorporated into recognition using graphical models (GMs). GMs are a class of factorized multivariate distributions that have been applied to problems across many disciplines. They were proposed for use in ASR in Zweig and Russell [1998], Zweig [1998], Bilmes [1998]. The standard distribution for ASR is the hidden Markov model (HMM). The HMM is a factorized distribution and is an instance of a graphical model. In their most general form, HMMs have very few theoretical limits [Bilmes, 2006]. However, certain graphical model distributions cannot be expressed using an HMM, such as the buried Markov model from Bilmes [1998, 1999a,b]. Other factorized distributions can be expressed as an HMM, but when expressed as a graphical model these distributions require orders of magnitude fewer parameters and orders of magnitude less computation. Furthermore, the implementations of the specific HMMs seen in standard recognizers impose limits that do not exist in a graphical model formulation. Flexible GM software systems also allow a researcher to quickly specify and run experiments using their model without modifying the code of a complicated piece of software. Finally, even when ideas can be expressed as an HMM using a standard tool kit, GMs provide a visual language that assists in the design, communication, and understanding of the ideas [Bilmes, 2006].

The distributions used here are a particular type of graphical model called a dynamic Bayesian network (DBN) [Dean and Kanazawa, 1989]. The baseline recognizer is simply a DBN implementation of a conventional HMM and can be seen in Figure 2. This model is similar to the recognizer proposed in Bilmes et al. [2001, Figure 16]; Zweig et al. [2002] and was implemented and used as the baseline model for Livescu et al. [2007]. The details of the baseline model will not be described here, but more information can be found on it in Bilmes and Bartels [2005], Bartels [2008].

3.1 Utterance-Level

The first graph that makes use of syllable nuclei is called **Utterance-Level** and is seen in Figure 3. This DBN is only used with oracle features, and it will only decode hypotheses that have the same total number of syllables as the reference hypothesis. This graph can be viewed as having four parts:

- The portion of the graph below the *Word* variable remains the same as the baseline, and all the trained parameters from the baseline model are used unchanged.
- The second portion of the graph counts the number of syllables hypothesized by the baseline portion of the recognizer. The variable *Word Syllables*, $\mathcal{S}^w$, gives the number of canonical syllables in the given word/pronunciation combination. At each word transition the value of $\mathcal{S}^w$ is added to the variable *Hypothesized Count*, $\mathcal{S}^{hc}$. Hence, in the last frame $\mathcal{S}^{hc}$ contains the total number of canonical syllables in the hypothesis.

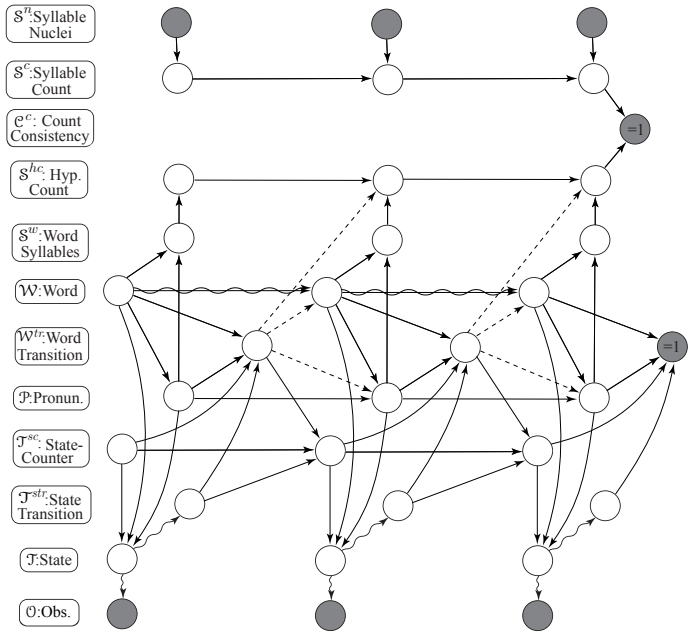


Figure 3: Utterance-Level decoder using oracle syllable count This DBN only allows hypotheses with the same total number of syllables as the reference transcription. Hidden variables are white while observed variables are shaded. Straight arrows represent deterministic relationships, curvy arrows represent probabilistic relationships, and dashed arrows are switching relationships. This graph has same the functionality as Bartels and Bilmes [2007, Figure 3] and was given in its current form in [Bartels, 2008].

- The next portion counts the number of observed syllable nuclei. The variable *Syllable Nuclei*, $\mathcal{S}^n$, is the observed binary nucleus indicator. *Syllable Count*, $\mathcal{S}^c$, counts the total number of observed oracle nuclei since the beginning of the utterance.
- Finally, the last frame in the DBN contains a variable called *Count Consistency*, $\mathcal{C}^c$. This variable is observed to be *true* and is defined by a deterministic function that is *true* whenever $\mathcal{S}^{hc}$ equals $\mathcal{S}^c$. This forces all hypotheses that have a different total number of syllables than the oracle syllable count to have zero probability.

3.2 Word-Level, Syllable-Level, and Nucleus-Level

The graph used for most of the experiments in this article is given in Figure 4. This same structure is used for three different implementations called **Word-Level**, **Syllable-Level**, and **Nucleus-Level**. **Word-Level** will be discussed first. It works in a similar manner as Utterance-Level but with two major differences. First, it checks the observed nuclei count at the end of each hypothesized word rather than waiting until the end of the utterance. Second, it has the ability to handle hypothesized (imperfect) nucleus observations. The variables in the graph work as follows:

- The portion of the graph below the *Word* variable remains the same as in the baseline and Utterance-Level.
- As in Utterance-Level, *Syllable Nuclei*, $\mathcal{S}^n$, is the observed binary nucleus indicator. In this graph it can be either estimated or observed.
- *Syllable Count*, $\mathcal{S}^c$, counts the total number of observed oracle nuclei since the beginning of the word. Its value is set to zero whenever there is a word transition in the previous frame.
- The variable *Transition Type*, notated $\mathcal{S}^{str}$, indicates if and what type of word transition is occurring in the current frame. Its possible values are the following: no word transition, last frame of a silence region, last frame of a 1 syllable word, last frame of 2 syllable word, and so fourth.

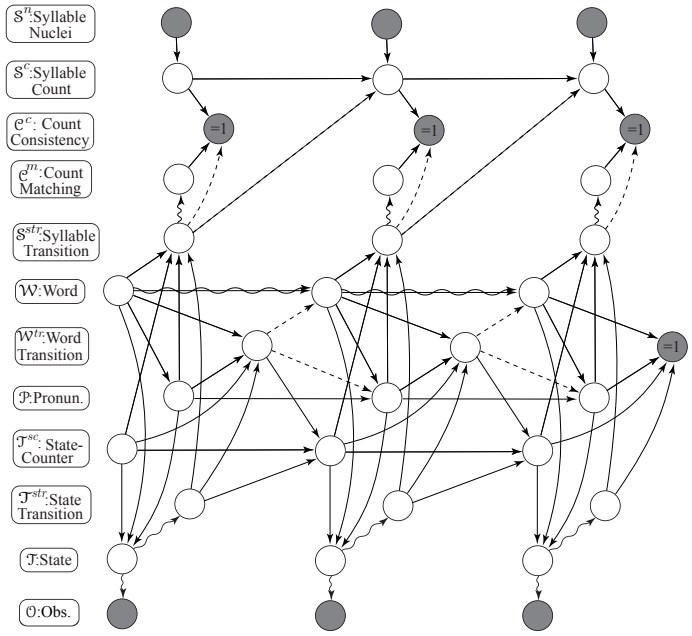


Figure 4: Word-Level, Syllable-Level, and Nucleus-Level graph (see Figure 2 for key). The bottom portion of this graph is identical to the baseline model given in Figure 2. The upper portion of the model counts the number of estimated syllable detections that occur during the duration of each word, syllable or vowel hypothesized by the lower portion of the model. [Bartels and Bilmes, 2008]

- *Count Matching*, $\mathcal{C}^m$, is a random variable that gives a distribution over detected syllable counts given the *Transition Type*. For example, when $\mathcal{S}^{str}$ indicates the end of a 1 syllable word this distribution will have a high probability for 1 detected syllable nuclei, lower probabilities for 0 and 2 detected nuclei, and even lower probabilities for 3 or more detected nuclei. This distribution, $p(\mathcal{C}^m = c|\mathcal{S}^{str} = s)$, is learned from the training data and is called the Count Matching CPT. $\mathcal{S}^{str}$ has no effect when there is no transition.
- *Count Consistency* is a constraint that is enforced whenever *Transition Type* does not equal *no word transition*. When the constraint is turned on it forces *Syllable Count* to be equal to *Count Matching*. This will cause $p(\mathcal{C}^m = c|\mathcal{S}^{str} = s)$ to be multiplied into the hypothesis score. Whenever *Transition Type* equals *no word transition* and the constraint is turned off, $p(\mathcal{C}^m = c|\mathcal{S}^{str} = s)$ has no effect on the model. When oracle features are used $p(\mathcal{C}^m = c|\mathcal{S}^{str} = s)$ is always equal to 0 or 1, the constraint only allows hypotheses that are consistent with the oracle features (in a similar manner as Utterance-Level).

One might question why the existing implementation uses *Count Matching* and why not consider removing it, moving its functionality to the *Count Consistency* variable. First consider making *Count Consistency* a hidden random variable with parents *Syllable Count* and *Syllable Transition*. If this were the case, *Count Consistency* could be described as "barren" as it would be hidden and have no children. Barren variables only occur in a single factor and summing or maximizing over them removes them from the distribution without affecting any of the other probabilities. In such an implementation the values of *Count Consistency*'s CPT would not change any decoded probabilities (see Pearl [1988]). Alternatively, one could keep *Count Consistency* as an observed variable but make it random with parents *Syllable Count* and *Syllable Transition*. If this were the case, the CPT $p(\mathcal{C}^c = 1|\mathcal{S}^c, \mathcal{S}^{str})$ could encode the same information that the Count Matching CPT does in the existing implementation; however, this new CPT could not be trained using expectation maximization. This is because $\mathcal{C}^c$ is always set to 1 and it will always be the case that $p(\mathcal{C}^c = 1|\mathcal{S}^c, \mathcal{S}^{str}) = 1$. Similar GM training issues are discussed in Reynolds and Bilmes [2005], Lin et al. [2009].

The next graph is called **Syllable-Level**. This uses the same structure as Word-Level, but the *Transition Type* variable behaves differently. In Syllable-Level *Transition Type* can take on four values: no transition, end of a silence region, end of a short pause, and end of a syllable. *Syllable Count* has essentially the same behavior as in Word-Level, but now the count indicates the number of detected syllables since the beginning of a single syllable or short

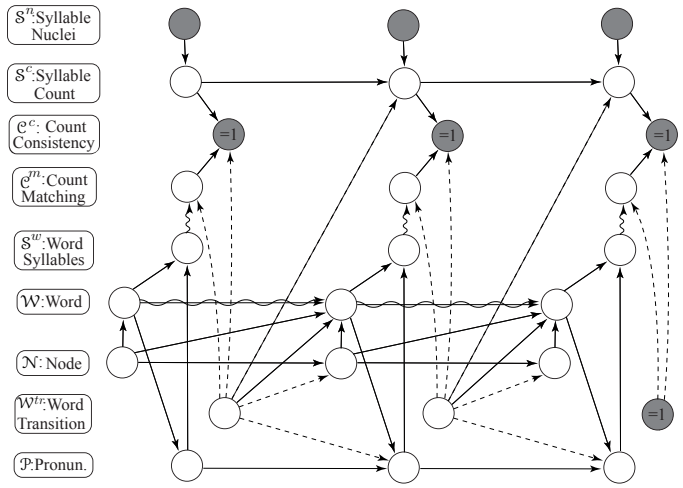


Figure 5: Word-Level Lattice (see Figure 2 for key). The bottom portion of the graph implements all possible paths through the lattice. The upper portion of the model counts of the number of estimated syllable detections that occur during the duration of each word hypotheses coming from the lattice. [Bartels, 2008]

pause rather than an entire word. Similarly, *Count Consistency* is now enabled at the end of every individual syllable and short pause rather than waiting until the end of the word. Syllable-Level behaves the same as Word-Level for silence regions. The syllable boundaries within multiple syllable words were determined using the same heuristic as was described in Section 2 for the creation of oracle nuclei features. This information is encoded in the deterministic function governing the *Transition Type* variable.

Along the same lines is the graph called **Nucleus-Level**. Again, this uses the same structure as Word-Level and Syllable-Level and once again the *Transition Type* variable behaves differently. *Transition Type* can now take on six values: no transition, end of a silence region, end of a short pause, and end of a syllable onset, end of a syllable nucleus, and end of a syllable coda. *Syllable Count* counts the number of detected syllables since the beginning of each of these regions, and *Count Consistency* is enabled at the end of each of them.

3.3 Word-Level Lattice

The next graph is **Word-Level Lattice** as seen in Figure 5. This baseline portion of this graph rescores lattices rather than performing first pass decoding. The lattice rescoring portion of the graph is based on [Ji et al., 2006]. The acoustic scores from the GMMs are stored in the lattice so there is no need for the PLP observation vectors. The functionality of the syllable counting stream is identical to the original Word-Level graph, but the implementation is slightly different. This graph uses a *Word Syllables* variable to specify the number of syllables in the current word. Together with the *Word Transition*, *Word Syllables* replaces the functionality of *Transition Type*. The reason for the difference is that using a *Transition Type* variable allows the structure to be used for Word-Level, Syllable-Level, and Nucleus-Level. This option is not available in the lattice-based graph because it does not have the *State Counter* variable that is needed for the general case.

3.4 Two Stream Syllable-Level

Two Stream Syllable-Level is designed to use *Oracle Stressed* features and is given in Figure 6. This graph has two counting streams:

- *Stressed Syllable Count*, $\mathcal{S}^{sc}$, counts the number of observed nuclei that are stressed. *Stressed Consistency* enforces the constraint that *Stressed Syllable Count* must be consistent with the syllable or silence region hypothesized by the lower portion or the recognizer. As usual, this constraint only occurs at the end of each syllable or silence. The graph uses oracle features, so the count must be exactly 1 for stressed syllables and exactly 0 for unstressed syllables, silences, and short pauses.

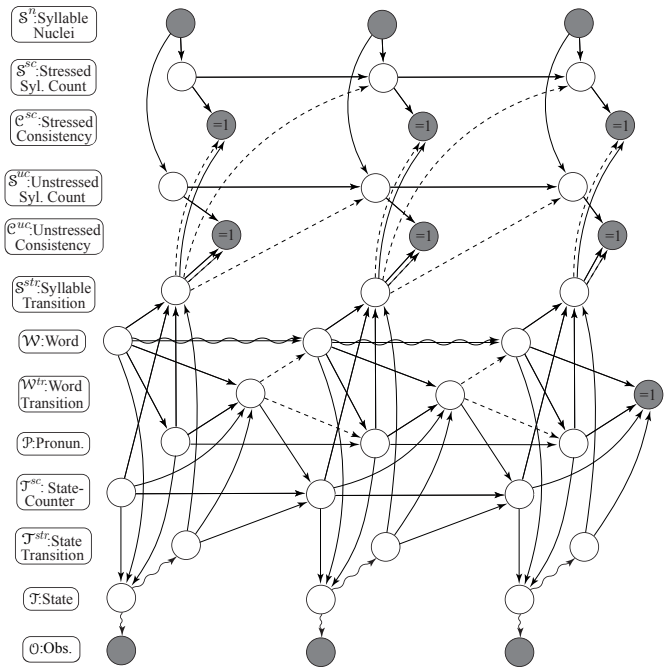


Figure 6: Two Stream Syllable-Level DBN (see Figure 2 for key). The bottom portion of this graph is identical to the baseline model given in Figure 2. The upper portion of the model has two syllable counting streams. The first counts the number of stressed syllables and the second counts the number of unstressed syllables. This model uses oracle features and the counts must match with probability 1. [Bartels, 2008]

- *Unstressed Syllable Count*, $\mathcal{S}^{uc}$, counts the number of observed nuclei that are unstressed. Analogous to the previous stream, *Stressed Consistency* enforces the constraint that the *Unstressed Syllable Count* must be consistent with the syllable or silence region hypothesized by the lower portion or the recognizer. Again, this only occurs at the end of each syllable or silence. The count must be exactly 1 for unstressed syllables and exactly 0 for stressed syllables, silences, and short pauses.

Note that this model could be extended to handle estimated nuclei by simply adding a *Count Consistency* variable to each stream.

3.5 Further Details

Note that the syllable counts in all of these models add additional state. This state allows the model to give differing scores depending on how the beginnings and ends of the hypothesized words align with the stream of detected syllables. One view of this system is that the phone-based recognizer hypothesizes a particular number of syllables, the syllable detector hypothesizes another number, and the graphical model provides probabilistic "glue" that encourages consistency between the two.

The baseline Gaussian parameters and transition probabilities were trained by maximizing likelihood using expectation maximization (EM). These parameters were imported directly into all of the other models. The only distribution in Word-Level and Syllable-Level that needs to be trained is $p(\mathcal{C}^m|\mathcal{S}^{str})$. The baseline parameters are held fixed while training $p(\mathcal{C}^m|\mathcal{S}^{str})$, and the training converges with four additional EM iterations.

All of these models including the baseline use a standard language model scale and word insertion penalty. When using estimated nuclei Word-Level and Syllable-Level have an additional scaling factor on $p(\mathcal{C}^m|\mathcal{S}^{str})$. This scale along with the language model scale and word insertion penalty are optimized on the development set separately for each model and feature set. The values for the scale and WIP are evaluated by the resulting word error rate on the development set. These three values were optimized jointly using a combination of grid searches and Nelder-Mead searches [Nelder and Mead, 1965].

Finally, an important part of all of the experiments is the mapping from a phonetic word pronunciation to syllables. In this system, there is one syllable for every vowel in the pronunciation. The dictionaries used here have explicit phones for syllabic consonants, and these phones are grouped with the vowels. Although this definition matches human intuition for most words, the precise definition of a syllable is not universally agreed upon. For some words the number of syllables is not clear, especially when several vowels appear consecutively and when vowels are followed by glides ('w' and 'y'). For example, one could reasonably argue for either two or three syllables in the word "really" when pronounced "r iy ax l iy". Fortunately we do not need to know the "true" definition of syllable, we only need a mapping that is consistent within our features and models. The syllable detection works by finding peaks in the posterior of the vowel detector, which is consistent with the model's definition of a syllable. Furthermore, any discrepancies between the syllable detector, the model, and the actual speech are reflected in the entropy of the *Count Matching* CPT, $p(\mathcal{C}^m|\mathcal{S}^{str})$. This issue is discussed again in Section 7.

4 Data

This section describes the corpora that are used throughout the article. Large vocabulary training was performed using the data from Switchboard-I Release 2 [Godfrey et al., 1992]. Switchboard is a set of approximately 2400 telephone conversations from 543 speakers, giving approximately 250 hours of training data. The speakers are from varying parts of the United States. Most importantly, this data is spontaneous conversational speech. Compared to read speech, spontaneous speech has more variability in pronunciation and speaking rate and a higher degree of coarticulation.

A majority of the experiments were performed on the 500 word task of the *SVitchboard* corpus from [King, Bartels, and Bilmes, 2005]. SVitchboard is a small, closed vocabulary subset of Switchboard I. This allows experimentation on spontaneous continuous speech but with less computational complexity, which therefore allows us to experiment with prototype ASR systems quickly. The name SVitchboard will be abbreviated with the name SVB. The A, B, and C folds were used for training. This is 8.4 hours of data, 3.7 hours of which is speech. The D fold was used as the development set giving 3.0 total hours of data which has 1.3 hours of speech. The E fold was used as the evaluation set with 3.2 hours of data, 1.4 hours of which are speech.

Most of the large vocabulary testing was performed on the development and evaluation sets of the 2001 NIST test sets for conversational telephone speech. This is part of a set of evaluations referred to as Hub5. Both the development and test sets have 20 conversations that were collected along with Switchboard-I, 20 telephone conversations from a set called Switchboard-II, and 20 cellular phone conversations.

An experiment was also performed on the large vocabulary development set for the 2003 Rich Transcription Spring English speech-to-text evaluation (abbreviated RT03). This consists of 40 land line telephone conversations and 20 cellular phone conversations. Each conversation is around 5 minutes in length.

5 Experiments and Results

This section describes the speech recognition experiments and gives the results. It will begin by describing the details of creating the syllable detection features. This is followed by a description of a series experiments accompanied by discussions of why each subsequent experiment was performed.

5.1 Feature Creation Details

Two sets of ANNs are used in the results. For both sets, the inputs to the network are PLP cepstra plus energy along with their deltas and double deltas. The features are calculated every 10ms with 25ms windows and are mean and variance normalized on a per speaker basis.

The first set of neural networks was trained using data only from the SVB training set. Training labels were created using a forced alignment by the baseline recognizer. This alignment marked each frame in the training and development sets as one of vowel, consonant, or silence. The alignment process used the word time boundaries from the transcriptions given by [Deshmukh et al., 1998], but it was allowed to move the word boundaries 2 frames to either side of frame the transcriptions list as the boundary (using the "slack" mechanism from Ji et al. [2006]). The choice of word pronunciation is given by the baseline recognizer as part of generating the alignments. The ANN learning rate parameters, number of hidden units, and number of input frames were chosen by the frame level accuracy on the

Table 1: Summary of syllable nuclei features

Estimated	Automatically detected, using SVB trained ANN
Estimated (Fisher)	Automatically detected, using Fisher trained ANN
Word Aligned (W.A.) Oracle	From forced alignments, spaced evenly within each word
Syllable Aligned (S.A.) Oracle	From forced alignments, in time center of each syllable
Vowel Aligned (V.A.) Oracle	From forced alignments, in time center of each vowel
Syllable Aligned (S.A.) Oracle Stress	From forced alignments, three valued feature reflecting lexical stress, in time center of each syllable

development set. This gave 200 hidden nodes and a 17 frame input window. The neural network training and decoding was performed using RegNet from Xiao Li [Li, 2007].

For the second ANN, an existing set of publicly available networks was used. A detailed description of the networks can be found in Frankel et al. [2007]. They were trained on 2000 hours of Fisher data using QuickNet from the International Computer Science Institute[4]. These neural networks produce posteriors for 46 phones, and the posteriors for individual vowels are summed to give a single overall vowel posterior.

Binary nuclei detection features were created from the posteriors of both ANNs in the following manner. For each utterance, the vowel and silence posteriors are smoothed in time using a Hamming window. The length of the window was chosen by the recognition performance of the resulting features on the development set. Next, the maxima in the smoothed vowel posterior are found. A maximum is taken to be any frame with a posterior larger than its adjacent frames. Maxima that occur less than 5 frames after a previous maximum are thrown away (a value of 5 was chosen to match a similar peak finding method given in Wang and Narayanan [2005]). These remaining maxima are interpreted as estimated locations of the syllable nuclei. Finally, a binary feature is created for each time frame. These features equal 1 in the frames where a maximum occurs, and they equal 0 in all other frames.

We also present results using *Oracle* syllable features. There are several different forms of oracle features that came about as the experiments evolved. In all of the feature types there is one syllable for every vowel in the lexicon (including successive vowels; this is discussed in Section 3.5). The first feature type is *Word Aligned (W.A.) Oracle*. The word boundaries are obtained from the time-aligned transcriptions from [Deshmukh et al., 1998], and word pronunciations are chosen using a forced alignment. The syllable nuclei for a word are evenly spaced within its boundaries. The feature value for the frame closest to each nucleus is given a value of 1, and all other frames are assigned a 0. An example is shown in Figure 1 (d).

The second type of oracle feature is *Syllable Aligned (S.A.) Oracle*. Generating these features required time-aligned phone transcriptions. These were generated from a forced alignment using the same process that generated the ANN targets. Recall that the choice of pronunciation is given by the baseline recognizer as part of generating the alignments, and word boundaries from [Deshmukh et al., 1998] were used but the alignment process was allowed two frames of "slack". Given a set of time-aligned phone transcriptions, the beginnings and ends of all words are marked as syllable boundaries. A heuristic is used to determine the boundaries within words that have more than one syllable. This heuristic splits strings of consonants that occur between vowels by placing the first consonant in the coda of the first syllable and subsequent consonants in the onset of the second syllable. The binary features are then created in a similar manner to W.A., except there is one nucleus placed in the center between each syllable boundary.

The third type of oracle feature is *Vowel Aligned (V.A.) Oracle*. These make use of the same time-aligned phonetic transcriptions obtained from the same forced alignments used for the Syllable Aligned Oracle features. In this case a nucleus is placed in the time center of each vowel. Finally, we have *Oracle Stressed* syllable features. These are three-valued features indicating a stressed nucleus, an unstressed nucleus, or no nucleus. The nuclei locations are the same as in the S.A. Oracle features. The features indicate lexical stress which is determined directly from the

[4]QuickNet is available at: `http://www.icsi.berkeley.edu/Speech/qn.html`

pronunciation. A summary of all of the nuclei feature types is given in Table 1.

5.2 SVitchboard Baseline

The first set of experiments were performed on the 500 word task of SVitchboard [King, Bartels, and Bilmes, 2005]. The observation vectors are 13 dimensional PLPs normalized on a per conversation side basis along with their deltas and double-deltas. It uses state clustered within-word triphones and implements a three state left-to-right topology. There are 455 Gaussian mixture models with a maximum of 32 components per mixture. The phone set and dictionary was based on dictionaries from the Spoken Language Systems group at MIT. Except where specified, all the DBNs were trained and decoded using the Graphical Models Toolkit (GMTK) [Bilmes, 2002, Bilmes and Zweig, 2002]. PLP features were created using the Hidden Markov Model Toolkit (HTK) [Young et al., 2005].

The baseline model was originally used in Livescu et al. [2007] where the evaluation set was reported as having a 59.2% word error rate (WER). The improved baseline result of 58.6% reported in [Bartels and Bilmes, 2007] and Table 3 is due to a larger beam size. There were two differences that gave the improved result of 52.3% reported in [Bartels and Bilmes, 2008]. After normalization, the PLP features are re-scaled to give the features a global mean and variance that is similar to the unnormalized features. The features used in [Livescu et al., 2007, Bartels and Bilmes, 2007] had a similar scaling, but the values were numerically smaller and were not actually reflective of the unnormalized features. Second, the minimum occupancy count for the creation of a triphone cluster was increased and triphones that are never seen in training were completely removed from the clustering process. Finally, a WER of 51.9% is reported here as the baseline result (Table 4). This improvement came from optimizing the language model scale and word insertion penalty on the entire D fold rather than the smaller D_short fold.

A summary of the models used on SVitchboard is given in Table 2. This table compares the baseline version each experiment was built upon and gives the number of parameters needed for each model.

5.3 Utterance-Level Versus Word-Level

The first experiment compares Utterance-Level to Word-Level using oracle syllable features. The goal of this experiment is to determine how much improvement we can gain in the best of cases by knowing the number of syllables in each utterance and how much (if any) additional information is in the nuclei locations.

The results are given in Table 3. The Utterance-Level oracle DBN gives a 7% relative WER improvement over the baseline. The word error rate improvement comes in the form of a reduction in deletions and insertions but with a rise in substitutions. The increased substitutions are the result of cases where the baseline hypothesis has a deletion and the oracle constraint forces the addition of a word which is incorrect.

Word-Level gives an 18% relative improvement over Utterance-Level. In comparison to Word-Level, the cardinality of the *Syllable Count* variable in Utterance-Level needs to be quite large which makes decoding slower, increases the memory requirements, and makes it more susceptible to search errors. Given this result, making use of the locations of the syllable nuclei appears to be of much more use than having only the syllable count. This experiment also shows that when using the oracle syllable features Word-Level gives a 24% relative WER improvement over the baseline. This substantial improvement is obtained simply from having syllable marks that are accurate in number and approximate in location, and it motivates further investigation of the model. The relative performance of Utterance-Level both in terms of WER and speed motivates no further investigation of this particular model.

5.4 Word, Syllable, Nucleus-Level Performance

Results using oracle syllable features are given in rows (b), (c), and (d) of Table 4. Using the oracle features with Nucleus-Level we achieve a 21% relative improvement in word error rate over the baseline. Nucleus-Level is the most accurate of the three graphs, followed by Syllable-Level and then Word-Level. This is likely because it can make the best use of the nuclei locations. Note that the Word-Level results differ from Table 3. There are two reasons for this. First, the results in Table 3 use W.A. Oracle features and the results in Table 4 use V.A Oracle features. Second, as described in Section 5.2 the word recognition portion of the graph was built from a different, improved baseline. The results in Table 3 are from an earlier experiment before both of these improvements were implemented.

The results using the estimated syllable locations are given in rows (e), (f), and (g) of Table 4. Using Word-Level we achieve a 2.5% relative improvement in WER which is significant at the 0.005 level according to a difference of proportions significance test. Unlike the oracle features, Word-Level, Syllable-Level, and Nucleus-Level all perform

Table 2: Table comparing the models used in the various SVitchboard experiments. The **Table** column references which table of results the particular model's results are originally given in. The **Syllable Features** column gives the type of nuclei features used for the experiment (if any). *W.A. Oracle* are the word aligned oracle features, *V.A. Oracle* are vowel aligned, and *S.A. Oracle* are syllable aligned. The **Baseline Version** indicates which sets of experiments used the same Gaussian parameters and transition probabilities. **LM Param.** gives the number of parameters in the bigram language model (the same language model was used for all experiments). **AC Param.** gives the number of trained parameters making up the Gaussian mixture models and transition probabilities. **Syllable Param.** gives the number of trained parameters in the count matching CPT (only needed when using estimated nuclei features). The **Total Param.** column gives the total number of trained parameters.

Model	**Table**	**Syllable Features**	**Baseline Version**	**LM Param.**	**AC Param.**	**Syllable Param.**	**Total Param.**
Baseline, SVB 1	3		SVB 1	3339	1055201	0	1058540
Utterance-Level	3	*W.A. Oracle*	SVB 1	3339	1055201	0	1058540
Word-Level	3	*W.A. Oracle*	SVB 1	3339	1055201	0	1058540
Baseline, SVB 2	4		SVB 2	3339	1150238	0	1153577
Word-Level	4	*V.A. Oracle*	SVB 2	3339	1150238	0	1153577
Syllable-Level	4	*V.A. Oracle*	SVB 2	3339	1150238	0	1153577
Nucleus-Level	4	*V.A. Oracle*	SVB 2	3339	1150238	0	1153577
Word-Level	4	*Estimated*	SVB 2	3339	1150238	35	1153612
Syllable-Level	4	*Estimated*	SVB 2	3339	1150238	9	1153586
Nucleus-Level	4	*Estimated*	SVB 2	3339	1150238	15	1153592
Skip / Rewind	5		Skip/Rewind	3339	1150398	0	1153737
Skip / Rewind+Word Lev.	5	*Estimated*	Skip/Rewind	3339	1150398	35	1153772
Extra Components	5		Extra Comp.	3339	1233662	0	1237001
Extra Comp.+Word Lev.	5	*Estimated*	Extra Comp.	3339	1233662	35	1237036
Syllable-Level	6	*S.A. Oracle*	SVB 2	3339	1150238	0	1153577
2 Stream Syl.-Level	6	*S.A. Stress*	SVB 2	3339	1150238	0	1153577

Table 3: Comparison of Utterance-Level to Word-Level using W.A. (word aligned) oracle syllable features on SVitchboard. S, D, and I are counts of substitutions, deletions, and insertions. WER is percent word error rate. [Bartels and Bilmes, 2007]

Graph	**Syllable Features**	**Development**				**Evaluation**			
		S	**D**	**I**	**WER**	**S**	**D**	**I**	**WER**
Baseline		585	234	157	53.2%	6815	3122	1803	**58.6%**
Utterance Level	*W.A. Oracle*	655	140	87	48.1%	6927	2935	1046	**54.5%**
Word Level	*W.A. Oracle*	628	50	41	39.2%	7418	913	603	**44.6%**

Table 4: SVitchboard results of Word, Syllable, and Nucleus-Level using *V.A. Orac.* (vowel aligned oracle) syllable features and estimated syllable features. Row (h) uses ROVER to combine the results from Word-Level, Syllable-Level, and Nucleus-Level using estimated syllable features. Some of these results are also run in combination with a silence oracle (abbreviated S.O.). S, D, I are counts of substitutions, deletions, insertions. Is is an estimation of how many of the insertions occur during silence regions. WER is percent word error rate. [Bartels, 2008]

		Syllable	**Development**					**Evaluation**				
	Graph	**Features**	**S**	**D**	**I**	**Is**	**WER**	**S**	**D**	**I**	**Is**	**WER**
(a)	Baseline		5833	2297	1108	704	51.1%	6676	2732	983	532	**51.9%**
(b)	Word-Level	*V.A. Orac.*	6454	503	588	16	41.7%	7344	669	738	11	**43.7%**
(c)	Syllable-Level	*V.A. Orac.*	6314	752	404	9	41.3%	7132	846	558	5	**42.6%**
(d)	Nucleus-Level	*V.A. Orac.*	6116	527	441	12	39.2%	6967	705	552	4	**41.1%**
(e)	Word-Level	*Estimated*	5898	2068	869	335	48.9%	6792	2510	838	233	**50.6%**
(f)	Syllable-Level	*Estimated*	5804	2327	705	251	48.9%	6693	2789	692	168	**50.8%**
(g)	Nucleus-Level	*Estimated*	5845	2281	756	264	49.1%	6758	2707	699	183	**50.8%**
(h)	ROVER (e), (f), (g)	*Estimated*	5828	2216	782	278	48.8%	6726	2655	736	197	**50.5%**
(i)	Baseline + S.O.		5933	2038	636	0	47.6%	6739	2478	726	0	**49.7%**
(j)	Word-Level + S.O.	*Estimated*	5972	1940	711	0	47.7%	6810	2339	812	0	**49.8%**
(k)	Syl.-Level + S.O.	*Estimated*	6059	1807	728	0	47.5%	6929	2208	793	0	**49.6%**

at a similar level. Row (h) uses ROVER [Fiscus, 1997] scored by frequency of occurrence to combine the Word-Level, Syllable-Level, and Nucleus-Level results.

5.5 Performance in Speech and Silence

When inspecting the results using the estimated syllable nuclei, it can be seen that many of the improvements result from removing insertions due to burst noises during silence regions. Figure 7 gives an example of the Syllable-Level graph removing such an error. The baseline model correctly decodes the word "right", but then incorrectly inserts a second "right" during a breath noise. The breath noise does not sound like the word "right", but it has an even worse acoustic match to the silence model. The syllable detector correctly identifies the syllable nucleus for the word "right", but it has five false detections during the breath noise. The Syllable-Level graph gives a low probability to decoding "right" during the noise because it is not a five syllable word. There is also a false detection during the initial silence, but it does not affect the result. This improvement is obtained by exploiting a mismatch between the syllable detector and the baseline recognizer and is discussed in more detail in Section 7.

To further investigate the theory that these improvements are from removing insertions due to burst noise, the number of insertions the models make during silence regions was calculated. This is the column labeled **Is** in Table 4. This value was determined by comparing the decoded word boundaries with the transcriptions from [Deshmukh et al., 1998]. If a decoded word lies entirely within a region labeled as silence in the transcriptions, then it is considered a silence insertion. If it only overlaps with speech in its first two frames or its last two frames but otherwise lies within a silence region, it is still considered a silence insertion. Note that the oracle feature experiments have non-zero silence insertions. Recall that the oracle features were created from forced alignments that allowed the word boundaries to be placed plus or minus two frames from the boundaries defined in [Deshmukh et al., 1998]. In a few cases when the alignment word boundary deviates from the transcription word boundary, a word can both agree with the oracle nuclei features and be a silence insertion. Word-Level, Syllable-Level, and Nucleus-Level yield significant reductions in silence insertions compared to the Baseline. These reductions are 56%, 68%, and 66%, respectively. This gives empirical evidence to support the anecdotal inspection given above.

This hypothesis was tested again by an additional experiment. Rows (i), (j), and (k) of Table 4 give results for the Baseline, Word-Level, and Syllable-Level with a frame level silence oracle. In these experiments the time-aligned transcriptions are used to place "hard" constraints on the model's hypothesis space, and all hypotheses that do not agree with the labeled silence/non-silence regions are pruned away. Note that the "silence" regions also include the non-speech noise, and with the help of the oracle information neither system will ever hypothesize speech in a noise region.

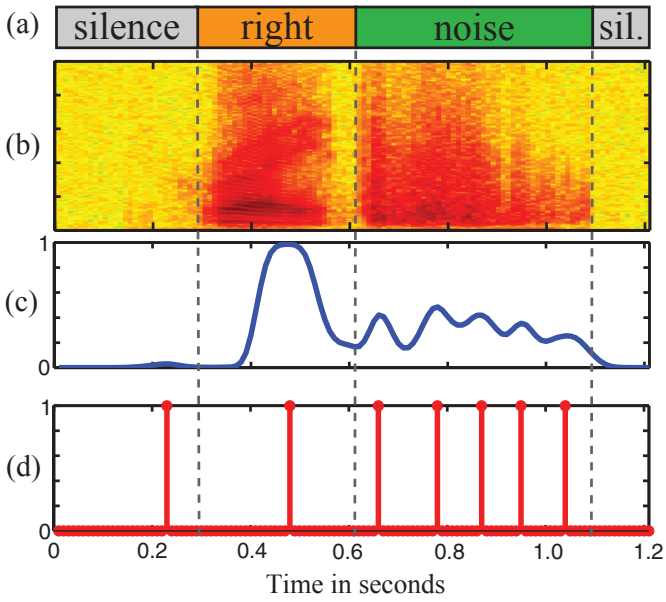


Figure 7: Example to illustrate a noise detection. (a) Gives the transcription. Note that the baseline recognizer decodes the noise as an additional "right", but Syllable-Level correctly decodes the noise as "silence". (b) Spectrogram of the audio signal. (c) Smoothed posterior from the Fisher trained neural network vowel detector (d) Maxima in the vowel posterior are interpreted as syllable nuclei. The one syllable in "right" is detected correctly. There are false detections in the silence and noise portions. This example is SVitchboard utterance sw4057B-0027-1. [Bartels and Bilmes, 2008]

Using the silence oracle, the baseline and the two syllable models give similar results. This, again, indicates that the improvement seen using the Word-Level and Syllable-Level models comes primarily from the reduction of insertions due to noise. It also shows that the models are quite effective at this task. The absolute improvement of 1.3% using Word-Level is 59% of the possible improvement of 2.2% given by the silence oracle.

Using the oracle syllable nuclei the performance improves as the syllable information becomes more fine-grained (Nucleus-Level outperforms Syllable-Level which outperforms Word-Level). With no oracle information Word-Level, Syllable-Level, and Nucleus-Level all perform at a similar level. Combining these scores using ROVER gives a small improvement, but this improvement is not statistically significant. This implies that the mistakes made by three of the models are similar. One possible explanation for the similar results is the following. There is evidence that the improvement is obtained during silence regions when speech hypotheses from the baseline recognizer do not match the syllable detector (for example, Figure 7). When this occurs the mismatch is significant to make a difference in all three models, and seldom is the mismatch subtle enough to, for example, be reasonable on Word-Level but not Nucleus-Level.

The primary difference between the results from the oracle features and the estimated features appears to be during the speech regions. The results using oracle features have a dramatic drop in deletions. There is also a small increase in substitutions that occurs because of cases where the baseline hypothesis has a deletion and the oracle constraint forces the addition of a word which is incorrect. The estimated syllable nuclei appear to help in some speech regions, but detection errors cause ASR recognition errors in others. The current syllable detection accuracy for speech regions is at a level that works well enough to not hurt the average performance.

5.6 Alternative Silence Models

Since the evidence suggests that the improvements when using estimated syllable nuclei is primarily during the silence regions, the next step is to see how the method compares to other more complicated silence models.

The results given in Table 4 all use a 3 state left to right model for both silence and short pause. The first alternative silence model that was tried is referred to as **Skip / Rewind**. In this model, the first state of silence has the option of

Table 5: Alternative silence model results on SVitchboard. S, D, I, Is are counts of substitutions, deletions, insertions. Is is an estimation of how many of the insertions occur during silence regions. WER is percent word error rate. [Bartels, 2008]

Graph	Syllable Features	Development					Evaluation				
		S	D	I	Is	WER	S	D	I	Is	WER
Baseline		5833	2297	1108	704	51.1%	6676	2732	983	532	**51.9%**
Skip / Rewind		5773	2364	929	526	50.2%	6627	2809	824	389	**51.2%**
Extra Components		5762	2320	816	399	49.2%	6622	2795	735	279	**50.7%**
Skip/Rewind + Word-Level	*Estimated*	5802	2313	746	269	49.0%	6693	2666	725	185	**50.4%**
Extra Comp. + Word-Level	*Estimated*	5875	2190	730	201	48.7%	6593	2694	742	152	**50.1%**
Extra Comp. + Sil. Orac.		5934	1996	680	0	47.6%	6702	2442	733	0	**49.3%**

Table 6: Two Stream Syllable-Level results on SVitchboard. The result for Syllable-Level uses syllable aligned oracle (*S.A. Oracle*) features, and the result for Two Stream Syllable-Level (2 Stream S.-L.) uses syllable aligned oracle stress (*S.A. Stress*) features. S, D, and I are counts of substitutions, deletions, and insertions. WER is percent word error rate. [Bartels, 2008]

Graph	Syllable Features	Development				Evaluation			
		S	D	I	WER	S	D	I	WER
Baseline		5833	2297	1108	51.1%	6676	2732	983	**51.9%**
Syllable-Level	*S.A. Oracle*	6362	571	479	41.0%	7183	750	571	**42.5%**
2 Stream S.-L.	*S.A. Stress*	5826	172	158	34.1%	6660	284	198	**35.7%**

transitioning directly to the third state (a skip) in addition to its usual options of staying in the state or transitioning to the second state. The third state has the option of transitioning back to the first state (a rewind) in addition to its usual options of staying in the state or leaving the model.

The silence models used for the results in Table 4 have 32 Gaussian components per state (as mentioned in section 5.2), which is the same number of components as the mixtures that model speech. In **Extra Components** more components are added to the silence mixtures. The reason for this is that they can better model the large variety of noises that do appear in the training data. The number of components per silence state was optimized on the development set to be 384.

The results for the alternative silence models are given in Table 5. Skip / Rewind gives a relative improvement of 1.3% and Extra Components gives a relative improvement of 2.3%. The Extra Components WER is approximately the same as the original Word-Level result.

Both of the alternative silence models can be combined with the syllable models. Word-Level is chosen here as it gave the best result using estimated nuclei in the experiments from Table 4. Skip / Rewind + Word-Level gives a 3.0% relative improvement over the original baseline, and Extra Components + Word-Level gives a 3.6% relative improvement. Also, Extra Components + Word-Level gives a relative improvement of 1.2% over Extra Components alone. This is 43% of the possible improvement of 2.8% relative (from Extra Components to Extra Components + Silence Oracle). It is interesting to note that Extra Components has fewer insertions than Extra Components + Word-Level, but the number of silence insertions (Is column) indicates that the improvement comes during silence regions. Also, if one compares the Word-Level results in Table 4 (seen in rows (b), (e), and (j)) to the Syllable-Level and Nucleus-Level results, Word-Level has comparably more insertions and fewer deletions.

It is also noteworthy that Extra Components + Silence Oracle performs 0.4% better than Baseline + Silence Oracle (Table 4, row(i)). A possible reason for this is that it is providing a better model for end of word short pauses, and this improved the training of the parameters for the phonetic GMMs.

Additionally, it would also be possible to combine Skip / Rewind with Extra Components. One could then combine all three of Skip / Rewind, Extra Components, and Word-Level. These combinations are deferred to future work.

Table 7: Large vocabulary results. The data are the Hub5 2001 development and evaluation sets. S, D, and I are counts of substitutions, deletions, and insertions. WER is percent word error rate. Other similarly configured first pass systems have reported results better than the baseline results presented here, such as the 45% WER on the SWB-II dev portion reported in Hain et al. [1999]. [Bartels, 2008]

		Syllable	Development				Evaluation			
Graph	**Data**	**Features**	**S**	**D**	**I**	**WER**	**S**	**D**	**I**	**WER**
Baseline	SWB-I		5992	2776	714	44.3%	5842	2733	659	**45.0%**
	SWB-II		7558	3187	947	57.1%	6242	2994	876	**51.1%**
	Cell		7776	3222	904	56.4%	7697	3581	968	**55.3%**
	Total		21326	9185	2565	52.5%	19781	9308	2503	**50.6%**
Word-Level	SWB-I	*Estimated (Fisher)*	5992	2589	736	43.5%	5900	2595	691	**44.8%**
	SWB-II	*Estimated (Fisher)*	7487	3096	917	56.1%	6301	2809	891	**50.6%**
	Cell	*Estimated (Fisher)*	7852	3016	951	56.0%	7826	3303	1013	**54.9%**
	Total	*Estimated (Fisher)*	21331	8701	2604	51.8%	20027	8707	2595	**50.2%**

5.7 Two Stream Syllable-Level

The results of the Two Stream Syllable-Level model from Section 3.4 can be seen in Table 6. Using oracle syllable stress features the model achieves a 16% relative WER improvement over Syllable-Level alone and a 31% relative improvement over the baseline. This is a promising result and motivates further investigation into integrating stress information. Note that the Syllable-Level results differ slightly from Table 4 because the S.A (syllable aligned) Oracle features are used here. Recall from Section 2 that the syllable aligned features place the oracle nuclei in the time center of the entire syllable, and the V.A. (vowel aligned) Oracle features used in Table 4 place the oracle nuclei in the time center of the vowel.

5.8 Large Vocabulary Baseline System

The 2001 Hub5 large vocabulary baseline system will now be described. The features are PLP cepstra plus energy along with their deltas and double deltas. They are calculated every 10ms with 25ms windows. The features are mean and variance normalized on a per conversation side basis with the normalization calculated only on speech utterances and omitting inter-segment gaps. A 3 state silence model with skip and rewind states was used, and the short pause is a single state that shares its Gaussian mixture parameters with the center state of silence. All other models are 3 state left to right. The system uses state clustered within-word triphones, and there are 8006 Gaussian mixture models with a total of 364k total components. Training was done on all of Switchboard-I [Godfrey et al., 1992]. Training made use of the time aligned transcriptions from [Deshmukh et al., 1998], but it allowed for soft word boundaries 3 frames to either side of each fixed word boundary specified by the transcriptions. This was done by specifying the transcriptions as a lattice (with only one path) and using the "slack" mechanism from Ji et al. [2006]. The phone set and dictionary was provided by SRI. The acoustic model parameter training was performed using EM to maximize the likelihood of the models using GMTK. No discriminative training or adaptation was performed.

The language model is a bigram trained using approximately 3 million tokens from Switchboard and 22 million words from Fisher. A vocabulary of the 61615 most common words was used. Decoding and lattice generation was done using HTK, and the lattice error rate is 13.0%. The language model scale and penalty were optimized on the development set resulting in a scale of 15 and a WIP of +1. The one-best results are given in Table 7.

5.9 Large Vocabulary Lattice Rescoring

The Word-Level lattice graph was used to rescore the large vocabulary lattices. The distribution $p(\mathcal{C}^m = c|\mathcal{S}^{str} = s)$ was trained on all of Switchboard-I. The syllable detection features were derived from ANNs trained on 2000 hours of Fisher (these were discussed in detail in Section 2). Results for rescoring the large vocabulary lattices using Word-Level are in Table 7. The improvement in WER is 0.4% absolute. This is significant at the 0.1 level according to a difference of proportions (strict) significance test. Looking at the three data subsets, the largest improvement was on the Switchboard-II data followed by the Cellular and then the Switchboard-I data. There are several possible

explanations of why the improvement was small. First, the larger training set allows a larger variety of noises to be included in the silence model. Unlike the SVitchboard system, many of the noises are marked in the training data and these can be mapped to a special "reject" phone. Note that, outside of Switchboard-I, explicitly marked noises are not typical in large vocabulary corpora.

Word-Level was also tried on lattices for the RT-03 development set. These lattices were generated by one of the final decoding passes in an LVCSR system from SRI which used a cross-word triphone acoustic model and a multi-word trigram language model. The resulting lattices were then rescored with a 4-gram language model which was used to generate the final lattices. The one-best word error rate is 23.4%. The same Switcboard-I trained $p(\mathcal{C}^m = c|\mathcal{S}^{str} = s)$ and syllable detection features from the previous lattice rescoring experiment were used. Word-Level was not able to improve on the development set's lattice one-best, so the evaluation set was not scored. The system that generated these lattices was already highly tuned so it is not entirely surprising that no improvement was found. In addition, the rescoring process limits the recognizer to the hypotheses expressed in the lattice. Even if the syllable model would have proposed the correct sentence in an unconstrained setting, if it is not in the lattice it will never be decoded.

The lattice rescoring experiments used a Word-Level model with estimated nuclei locations. This was the best performing model on SVB using estimated locations, but Syllable-Level and Nuclei-Level models could have also been tried. In addition, forced alignments could have been used to create oracle nuclei features for the large vocabulary data sets. Oracle nuclei gave large improvements on SVB and would likely give improvements on these tasks as well.

6 Previous Work

The use of syllable information in automatic speech recognizers has been a topic of research in the past. The first and most similar approach to the work presented in this article is to use syllable location information to constrain a recognizer. There is also related work that integrates syllabic information using a DBN. Another topic is syllable detection and is also an integral part of this work. A differing but related approach is to employ the syllable as the basic unit of recognition instead of the widely adopted path of using phonetic units. Finally, other ways of dealing with noise are presented. Each of these areas will be discussed in turn.

6.1 Syllable Based Segmentations for ASR

Early work in ASR required speech to be segmented into short units because of the computing constraints at the time. Phonetic segmentations were commonly used, but syllabic segmentations were also explored. Mermelstein [1975] proposed segmenting speech into syllable length segments for use in a system that would decode the segments using phone based models. In Davis and Mermelstein [1978] the performance of various feature types were compared on manual syllable segmentations. The experiments in Zwicker et al. [1979] automatically segmented speech by half syllables by detecting both syllable boundaries and peaks of nuclei. Speech was automatically segmented into whole syllables using onset estimations in Hunt et al. [1980]. All of these methods made hard decisions about syllable locations.

The work from Wu et al. [1997], Wu [1998] that was discussed in Section 1 is also an example of making use of syllable segmentations.

The first advantage of the models presented here over previous work is the degree to which the syllable segmentation information is provided to the recognizer in a "soft" manner. The DBN allows the syllabic information to be used to modify hypothesis scores in both first pass decoding and lattice rescoring. This combining of information can be seen as an on-line re-ranking of the hypotheses. A related difference of this article is that the DBNs presented here align syllable nuclei locations with word hypotheses using varying degrees of asynchrony giving a "soft" interpretation of the location information. A third distinctive point is that syllable nuclei are used exclusively, and boundary information is not incorporated. Syllable boundaries themselves are imprecise for two reasons. First, when two nuclei within a word are separated by consonants it is not always obvious which of the two syllables the consonants belong to. For example, consider the word bottle, "b aa dx ax l". The "dx" could be assigned to either the first or second syllable. Other analyses would assign it to *both* syllables (known as ambisyllabicity). The second reason syllable boundaries are imprecise is coarticulation. A human might move some of their articulators to begin pronouncing a syllable before finishing the previous one. Although coarticulation also exists in vowels, we only need to locate a single point somewhere within the vowel region as opposed to locating a boundary between two consonants.

6.2 DBNs using Syllabic Information

A related paper was given by [Basu, 2005] where a DBN was used to jointly detect speech and voicing. This work was notable because it characterized speech as a signal that alternates between voiced and unvoiced regions (which is a syllabic pattern) and made use of a DBN. However, no speech recognition was performed.

The research in Çetin [2004], Çetin and Ostendorf [2005] that was discussed in Section 1 is another example of using a DBN to model syllabic information.

6.3 Syllable Detection

There has been a number of different efforts in syllable detection. Many of these efforts concentrated on the problem of detection, but in others the detection was part of a recognition system or a precursor to a different analysis. There have been two primary ways that syllable locations have been identified. The first is to use signal processing methods to identify peaks in energy and correlation, and the second is to use a discriminative classifier.

Early work in syllable detection concentrated on finding peaks and valleys in speech energy followed methods of removing spurious detections. In Mermelstein [1975] the energy in the frequency range of speech was measured, and this signal was low pass filtered in time. The minima of the energy curve were considered potential syllable boundaries. Minima that had a small distance in the energy curve from nearby minima were removed from consideration by an algorithm that built a convex hull around each syllable. An energy curve was also used in Zwicker et al. [1979], but in that detector some frequency bands outside the range of vowels were subtracted from the energy to bias it towards vowel energy. Both maxima and minima of the modified energy curve were used to segment the speech into half syllable units. False boundary detections were rejected based on a measure of performance from the downstream pattern matching. In Pfitzinger et al. [1996] nuclei were again estimated from peaks in a smoothed energy contour of a bandpass filtered speech signal. False alarms were reduced by rejecting peaks lower than a threshold and by rejecting all but one peak from sets of peaks that are too close in time. In Pfau and Ruske [1998] syllable nuclei were detected using a loudness function similar to Zwicker et al. [1979], and false peaks were filtered in two ways. First, peaks that were not sufficiently close to a neighboring decline in the energy curve were removed. Second, peaks in speech regions that have sufficiently high zero-crossing rates were removed.

Instead of an energy curve, Morgan and Fosler-Lussier [1998] used the sum over the product of all pairs of sub-band energy envelopes. This gave a measure of the correlation between sub-band energies (peak counting of this measure gave one of the 3 measures in the speaking rate estimate they called *mrate*) . This idea was expanded on in Wang and Narayanan [2005, 2007]. First, a 19 band filter was applied to the waveform, and the 5 bands with the most energy were selected. The selected sub-band energy curves then underwent a weighting window followed by a temporal correlation function. The sub-bands were then combined using the sub-band correlation measure from Morgan and Fosler-Lussier [1998] and smoothed in time. Peaks that are lower than a threshold are rejected and all but one peak from sets of peaks that are too close in time are rejected. This method also employed a voicing detector and rejected peaks in regions the detector marked as unvoiced.

As mentioned, other work uses a discriminative classifier to locate syllable boundaries or nuclei. In Wu et al. [1997], Shire [1997], Wu [1998] a neural network was used to find syllable onsets. The input features to the neural network were RASTA-PLP and a set of specially designed spectral onset features. To calculate the spectral onset features the speech was first converted to an energy compressed spectrogram. The spectrogram was then convolved with a temporal filter that enhanced changes in energy on the order of 150 milliseconds. Then, the spectrum was smoothed across frequency channels using a Gaussian filter to enhance regions where adjacent channels were changing concurrently. The signals were then half-wave rectified and averaged over a set of nine human auditory inspired critical bands. The ANN was trained to discriminate between frames that do not have a syllable transition and frames that were within a five frame window of a labeled syllable transition. The targets for training were derived from hand-transcribed phonetic transcriptions. Peaks in the ANN posterior were interpreted as possible syllable boundaries, and false positives were reduced by thresholding by the posterior height. The work in Shire [1997] also described an HMM that removes some onset detections based on their duration.

Temporal Flow Model (TFM) neural networks were used in Shastri et al. [1999] to find syllable boundaries. This network had a single continuous valued output node that was trained to have a low value near syllable boundaries and a high value near the nucleus. An onset was detected whenever the output of the TFM was increasing and crossed a threshold, and this threshold was chosen dynamically. The systems used features derived from a modulation-filtered spectrogram, and the target outputs for training were derived from Syllable-level transcriptions.

In Schutte and Glass [2005] a support vector machine was used to classify speech as sonorant or non-sonorant. Training made use of phonetic-level transcriptions. The SVM output was smoothed in time, and a system of dynamically thresholding the smoothed output was developed

The work in Dashiell et al. [2008] combined nuclei estimations from a HMM based manner classifier, an energy curve method from Pfau and Ruske [1998], and the correlation based method from Wang and Narayanan Wang and Narayanan [2005, 2007]. It also estimated syllable boundaries by combining the manner classifier, the energy curve method from Pfau and Ruske [1998], and using minima of the spectral discontinuity features from Wu et al. [1997], Shire [1997].

Syllable detection is used in this article, but finding an optimal detection method was not a major focus. The detector used for the experimental results in this article use PLP features as input to a neural network, and nuclei are chosen as the peaks of this curve. This method is similar to Wu et al. [1997], Shastri et al. [1999], Schutte and Glass [2005], and was discussed in detail in Section 2.

The vowel posteriors shown in Figures 1 and 3 correlate well with a simple energy envelope and one might wonder why the more complex neural network based approach was used. The input features to the neural network include energy as well as 12 additional PLP coefficients along with their delta and double deltas. The neural network has the potential to use this extra information to increase its accuracy, though no direct comparison was done here. The work mentioned above indicates that an energy envelope is not likely to be optimal. In Morgan and Fosler-Lussier [1998] peak finding in an energy curve was found to be less accurate for estimating speaking rate than a sub-band correlation based method, and most of the other signal processing based syllable detection methods mentioned above make use of frequency sub-band energy rather than the full energy envelope. The most accurate syllable peak detector in Dashiell et al. [2008] was obtained by combining a modified energy curve with correlation and classifier based methods.

6.4 Syllables as a Recognition Unit

There has been a number of authors who have used the syllable as the basic unit of recognition (as opposed to the widely adopted phonetic units). The primary motivation for this is the hypothesis that pronunciation variation is better modeled with syllables than with phones. Using the syllable as the basic recognition unit was proposed very early on in Fujimura [1975]. This work proposed that speech should be characterized in terms of syllable onset, nucleus, and coda along with sub-classes within each of these sub-syllable units. This was actually deployed in a dynamic template matching system in Zwicker et al. [1979] where template matching was used to recognize half-syllable units, and in Hunt et al. [1980] where a similar approach was used on whole-syllable units. Syllables were employed as the basic recognition unit in an HMM based system in Green et al. [1993].

Syllables were combined with an articulatory approach in Kirchhoff [1996]. There, a set of articulatory features were recognized independently and these feature values were then matched to syllable templates.

Syllables as a basic recognition unit were again analyzed by a team of researchers in Ganapathiraju et al. [1997]. This work created a set of models for the 200 most common monosyllabic words, and a second set of syllable models for the syllables in the remaining monosyllabic words and multiple syllable words. Phone models were also created and used on any syllable with inadequate training data. This system gave a modest improvement over a triphone baseline.

A number of papers have reported improvements when combining phone and syllable models. The work of Dupont et al. [1997] used a standard phone based recognizer combined with a syllable based recognizer using asynchronous HMMs that are fused together at dynamically located "recombination states". Syllable time-length information was incorporated at a number of different levels in Wu et al. [1998b], Wu [1998]. One system used phones as a recognition unit, but used modulation spectrogram features that emphasized syllabic frequency modulations and used a syllable length context window as input to its ANN based phone classifier. A second system additionally used half-syllables as a recognition unit. The system with syllable-based signal processing and phonetic recognition units outperformed the baseline in reverberant conditions. Combining the system with syllable units with the phonetic baseline system outperformed all other systems. These same authors in Wu et al. [1998a], Wu [1998] combined the phonetic baseline and the system with syllable units by rescoring N-best lists. Individually the baseline system outperformed the syllable system, but the combined system outperformed both.

The analysis of the pronunciations of spontaneous speech in Greenberg [1998], Fosler-Lussier et al. [1999], Greenberg [1999], Chang [2002] gave arguments that pronunciation variation should be modeled with syllables. They showed that 85% of onsets are pronounced canonically but only 65% of nuclei and 63% of codas are. They also argued that prosodic prominence and lexical stress also have a large influence on pronunciation variation. The work

in Chang [2002] used this as a basis for a multi-tier recognition system that integrated articulatory features and stress accent into syllable based recognition units.

The DBNs presented in this article are motivated by the integration of long term segmentation and duration information, and they do not attempt to use syllables as a basic recognition unit. The ideas presented here are compatible with a syllabic recognition unit, and it would be straightforward to apply them to such a system.

6.5 Previous Work in Syllable Stress for ASR

There has also been previous work in using syllable stress in speech recognition. Research on improving recognition using stress and other prosodic aspects of speech is a significant body of work and will not be fully covered here, but a number of relevant papers will be mentioned.

Aull and Zue [1985] analyzed how much the vocabulary size can be constrained by knowing the stress pattern. The work in Hieronymus et al. [1992] marked stressed and unstressed vowels in a phoneme lattice as a precursor to a lexical search. In Wang and Seneff [2001], Gaussian mixtures were used to classify syllables as having "reduced", "unstressed", "primary stress", or "secondary stress". The GMM scores were included for vocalic segments and ignored for all others. As described above, Çetin [2004], Çetin and Ostendorf [2005] employed separate models for stressed and unstressed nuclei in a multi-rate DBN. More recently, in Ananthakrishnan and Narayanan [2007] n-best lists are rescored with a model that includes a term for prosodic scores with stress being part of this score.

6.6 Handling Noise

The syllable-based DBNs presented in this article achieve improvements in word error rate by reducing spurious insertions due to burst noises. This section will discuss previous efforts in speech/noise discrimination and noise robust ASR.

A large number of ASR techniques include improved noise robustness as one of their goals, and this subsection is not meant as a comprehensive list. Most any adaptation technique will improve recognition performance in noisy environments. There are also a number of feature types designed with noise in mind. RASTA-PLP feature processing assumes environmental noise varies slowly compared to the linguistic content, and the spectral estimates are passed through a filter that removes steady state and slowly varying components [Hermansky and Morgan, 1994]. In Chen et al. [2002, 2005], it is argued that there is also an upper bound in the rate that linguistic information is conveyed in speech, and a low pass filter is applied in the feature domain. Such approaches tend to be more effective on stationary noise than on bursts. A TRAP feature uses a vector that represents the temporal behavior of 1 second of speech from a single frequency band [Hermansky and Sharma, 1998, 1999], and it is argued that relative temporal trajectories are more noise robust than methods based on cross sections of the auditory spectrum.

Another related research area is the development of voice activity detectors (VADs). VADs typically output binary speech/non-speech decisions and are designed for use in variable rate speech coding. The VAD described by the International Telecommunication Union recommendation G.729 [Benyassine et al., 1997] makes use of four features: a line spectral frequency representation of linear prediction coefficients (LPCs), full-band energy, low-band energy, and the zero-crossing rate. The algorithm calculates averages of these features for silence regions and updates these averages with new data as it comes in. The difference between the silence averages and the feature values at the current frame are compared using a linear decision boundary that was manually tuned.

More recent VADs use statistical approaches. In Sohn et al. [1999], Gaussian models were crated for background noise and speech plus background noise. A Kalman filter based VAD was created in Fujimoto and Ishizuka [2008]. This VAD updates its noise model in an unsupervised manner.

There has also been some work integrating VAD results into speech recognition. The output of a neural network based VAD was integrated into ASR in Beaufays et al. [2003]. This VAD made binary speech/non-speech decisions but was integrated into recognition in a soft manner using a penalty factor. In Fujimoto et al. [2008] VAD results were used as hard constraints for a recognizer.

The model proposed here makes use of temporal dynamics at a syllable-length time scale. This is typically not done in a VAD (in part because their applications typically require real-time frame level classification). Another distinctive point of the model presented here is that it makes the speech/non-speech decision in a soft manner at recognition time. Finally, the DBNs presented here have the potential to label noises as non-speech even when the noise is unlike the surrounding test data and never appear in the training data. The DBNs also make use of a trained silence model that includes the noises that do exist in training. Methods such as Benyassine et al. [1997], Fujimoto and Ishizuka [2008]

can deal with noises not seen in training by building a silence model from the test data, but they are not particularly adept at dealing with burst noises, especially if they are not seen elsewhere in the test data.

7 Discussion

When a noise occurs, often both the baseline speech recognizer and the neural network give a high posterior probability to speech. The syllable DBNs are able to correct mistakes in the baseline because they recognize that a mismatch between the baseline hypothesis and the detected number of syllables is indicative of noise. An example of this was given in Figure 7 and discussed in Section 5.5. The neural network posterior given in this example is from the network that was trained on 2000 hours of speech data. Even with such a large amount of training data this is a difficult task.

Many of the false detections are clearly not syllable nuclei, and one could easily suppress many of these based on the vowel or silence posteriors at these points. The primary reason that this is not done is that the frequency of their occurrence is informative, and including them improves the speech detection performance. One technique that was tried during the development of the graphs was to not include peaks with posteriors below a threshold and/or not include peaks when the corresponding silence posterior is above a threshold, but including all of the maxima as nuclei gave a better word error rate. Assuming that high posteriors indicate speech can be problematic because the neural network is often fooled by the same noises that fool the baseline recognizer. In Figure 7, the five false detection points during the breath noise have a vowel posterior that is larger than the silence or consonant posteriors. Although these posteriors are smaller than the one correct detection in the figure, in other speech segments it is not uncommon to see true vowels with similar posteriors.

The corrections mentioned above (and similar such corrections) are not a result of a particularly good acoustic match between training and test data. Instead, they are a result of the two portions of the model being inconsistent and thereby precluding ”speech” as being hypothesized at those points. Furthermore, theoretically there do not need to be any examples of a specific noise in the training data for a mismatch to occur and cause it to be properly decoded as non-speech. (This hypothesis has not been directly tested, though. Noises are not labeled in the training and test sets that were used here.)

This method can also be seen as classifier combination. Typical classifier combination approaches concentrate on ways of choosing between a set of alternative hypotheses. The effect seen here is different in that neither hypothesis is correct; instead, the mismatch is used as an information source.

Another important aspect to the work in this article is its ability to incorporate information over time spans longer than the frame rate. This is done in two ways. First, the long analysis window of the ANN along with locating peaks in the smoothed ANN posterior analyzes the signal over syllable length time scales. Second, the syllable counting portion of the model adds additional state that is related to the duration of each hypothesized syllable. Other work that makes use of longer time scales is typically in the form of a feature that is appended to the standard observation vector, such as [Hermansky and Sharma, 1998]. The method here is similar to the focused evidence methods given in Subramanya et al. [2005], Bartels and Bilmes [2005]. It is similar in that the syllable features do not need to “pass through” a Gaussian mixture that is modeling phonetic properties of the signal. Also, using discrete peaks in the posterior provides segmentation at the resolution of syllable nuclei. Using long term features directly, such as a long analysis window or a moving average over the estimated nuclei, would smear information across syllable and word boundaries.

Finally, an important part of all of the experiments is the mapping from a phonetic word pronunciation to syllables. In this system, there is one syllable for every vowel in the pronunciation. It would be desirable to know the number of errors that are caused by discrepancies between the number of syllables in the lexicon and what was actually said and the number of errors that are caused by mistakes made by the syllable detector. In addition, in the oracle features the number of syllables is based on the pronunciations in the lexicon. If what was actually spoken deviates significantly from the available pronunciations the oracle features will not reflect this. The analysis of human labeled phonetic transcriptions in [Greenberg, 1999] found that deviations from the canonical pronunciation of the nucleus are almost always substitutions to other vowels. This would lead one to believe that the number of syllable errors caused by differences between the actual speech and the lexicon to be small in comparison to detection errors. An examination of a small sample of the SVitchboard utterances used here was consistent with this, but an in-depth analysis was not performed.

8 Future Work

The DBNs presented here for integrating syllable nuclei successfully discriminate between speech and non-speech noise which results in a significant improvement in word error rate. Currently, the models do not appear to provide any advantages inside sub-segments that the baseline model correctly hypothesizes as speech. Given the large improvement using the oracle syllable nuclei, future work will examine how to make the syllable detection more robust. In particular, the vowel detection ANN could potentially benefit from additional feature types that are typically not used in ASR. The current set of PLP features could be supplemented with information about voicing. Examples include the output of a voicing detection algorithm, time and frequency correlation, zero crossings, modulation spectrum, and spectral entropy. Also, combining multiple estimators is typically a good path for improvement and could be employed here to improve syllable detection as well.

Furthermore, for any of these improvements to the syllable detection method, the detector could be evaluated independently of the recognizer by comparing its output to human transcriptions, such as from [Greenberg et al., 1996]. This would also allow the neural network based detector to be directly compared to the signal processing and energy envelope based methods described in Section 6.3.

Another direction for improvement is to integrate syllable detections in a soft manner. This was implemented in an earlier version of the Word-Level graph and results were reported in Bartels and Bilmes [2007]. The soft integration was removed because using all of the detections was more effective, but certainly not all possibilities were exhausted and this concept could be revisited. In particular, the previous work focused on the height of the maxima. Additional information, such as curvature, distance in time from nearest maxima, and classifier entropy could also be used. Also, neural networks are not the only option for a classifier. Ratio Semi-definite Classifiers [Malkin and Bilmes, 2008, 2009] might provide a posterior estimate that is less confident in regions that are not being classified accurately.

Future work could also use linguistic knowledge to help better model when to expect syllables to be accurately detected. Considering stress in the models is the logical first step. This could be done simply by giving stressed and unstressed syllables in the lexicon different *Count Matching* CPTs. One would expect stressed syllables to have a higher probability of detection, and unstressed syllables to have a lower probability. The context of a vowel might be another important indicator of when the syllable is independently detectable, and again could be used as a conditioning variable for *Count Matching*. Further work could use acoustic information to automatically detects syllable stress, and the Two Stream Syllable Level graph could be altered to make use of this information.

Here only graph applied to lattice rescoring was Word-level. It would be possible to create Syllable-level and Nucleus-level graphs to rescore lattices. Note that these graphs would need a recognition section capable of giving the syllable boundaries for the word hypotheses.

The syllable DBNs could also be applied to phone recognition, which is an important part of many spoken term detection systems, such as in Lin et al. [2008]. The motivation is that a phone recognizer does not have the help of a language model, and this weaker classifier might benefit more from information about segmentation and syllabic structure more than a word recognizer.

The syllable DBNs could also be used to integrate many other types of information. The DBN can be seen as a general framework for incorporating any speech events that are better thought of as a long-term detection than as frame-level features. Potential applications are tone detections for tonal languages and disfluencies.

Another potential improvement in syllable detection could come from improved training labels. The existing neural networks are derived from training labels from a forced alignment that is based on a pronunciation lexicon. If the spoken pronunciations differ from the canonical pronunciations in the lexicon, this might not be optimal. An improvement might be obtained by re-training the vowel detector using labels derived from the initial version of the vowel detector.

A neural network was used here as part of the syllable detection step, but there are many ways that neural network posteriors can be utilized in a speech recognition system. Hybrid systems replace the GMM acoustic model with ANN posteriors. Alternatively, tandem features are derived from ANN posteriors Hermansky et al. [2000] and are typically appended to the standard feature vector. Future work could perform direct comparisons between these approaches and the syllable based graphs presented here.

Finally, the syllable nuclei based graphs are already successful in speech / noise discrimination as part of an ASR task. It might be fruitful to use the graph directly as an advanced voice activity detector. Since recognition accuracy would not be a concern, the vocabulary size would not need to be increased or could possibly even be reduced.

9 Synopsis

This article presented a number of motivations and accomplishments of the syllable counting DBNs. These are summarized below:

- Speech has a rhythmic modulation at 3 to 6 Hertz that is not typically found in noise. The time scales of many HMM based systems are too short to take advantage of this, but the syllable DBNs show demonstrable improvement using this information.
- Syllables provide segmentation and duration information that is available without knowing word or phonetic identities. The oracle feature experiments show that this information can potentially give large improvements in WER.
- A straightforward extension to the framework provides information to the recognizer about lexical stress, and the oracle experiment gave large improvements.
- The syllable DBNs can, in theory, handle noises that are not seen anywhere else in the training or test data.
- The reduction in insertions from burst noises can also be seen as gaining information from disagreement between the syllable detection and word recognition portions of the models.
- Unlike previous methods, DBNs allow the segmentation to be specified in a soft manner.
- The information about syllable segmentation is provided to the model in a "focused" manner and does not need to be included in the portion of the model describing phonetic identities.
- The discrete nuclei detections provide segmentation information at the resolution of syllables. Using features with long analysis windows lack temporal resolution and may not be accurate near word and syllable boundaries.
- The DBNs provide a general framework that could be extended to integrate any speech event that is better thought of as a long-term detection than a frame-level feature.

Acknowledgments

This work was supported by ONR MURI grant N000140510388.

References

Sankaranarayanan Ananthakrishnan and Shrikanth Narayanan. Improved speech recognition using acoustic and lexical correlates of pitch accent in a n-best rescoring framework. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages IV–873–IV–876, 2007.

Takayuki Arai, Misha Pavel, Hynek Hermansky, and Carlos Avendano. Intelligibility of speech with filtered time trajectories of spectral envelopes. In *Proc. of ICSLP*, volume 4, pages 2490–2493, 1996.

Ann Marie Aull and Victor Zue. Lexical stress determination and its application to large vocabulary speech recognition. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 10, pages 1549–1552, Apr 1985.

Chris Bartels. *Graphical Models for Large Vocabulary Speech Recognition*. PhD thesis, University of Washington, Seattle, 2008.

Chris Bartels and Jeff Bilmes. Focused state transition information in ASR. In *Proc. of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 191–196, San Juan, Puerto Rico, November/December 2005.

Chris Bartels and Jeff Bilmes. Use of syllable nuclei locations to improve ASR. In *Proc. of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 335 – 340, Kyoto, Japan, December 2007.

Chris Bartels and Jeff Bilmes. Using syllable nuclei locations to improve automatic speech recognition in the presence of burst noise. In *Proc. of Interspeech*, Brisbane, Australia, September 2008.

Sumit Basu. A linked-HMM model for robust voicing and speech detection. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages I–816 – I–819, 2005.

Francçoise Beaufays, Daniel Boies, Mitch Weintraub, and Qifeng Zhu. Using speech/non-speech detection to bias recognition search on noisy data. In *Proc. of ICASSP*, volume 1, pages I–424 – I–427, 2003.

A. Benyassine, E. Shlomot, H.-Y. Su, D. Massaloux, C. Lamblin, and J.-P. Petit. Itu-t recommendation g.729 annex b: a silence compression scheme for use with g.729 optimized for v.70 digital simultaneous voice and data applications. *Communications Magazine, IEEE*, 35(9):64–73, Sep 1997. ISSN 0163-6804.

Jeff Bilmes. Data-driven extensions to HMM statistical dependencies. In *Proc. of Int. Conf. on Spoken Language Processing*, pages 69–72, December 1998.

Jeff Bilmes. Buried markov models for speech recognition. In *Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 713–716, Phoenix, Arizona, March 1999a.

Jeff Bilmes. *Natural Statistical Models for Automatic Speech Recognition*. PhD thesis, University of California, Berkeley, 1999b.

Jeff Bilmes. *GMTK: The Graphical Models Toolkit*, 2002.

Jeff Bilmes. What HMMs can do. *IEICE Transactions in Information and Systems*, E89-D(3):869–891, March 2006.

Jeff Bilmes and Chris Bartels. A review of graphical model architectures for speech recognition. *IEEE Signal Processing Magazine*, 22(5):89–100, September 2005.

Jeff Bilmes and Geoff Zweig. The graphical models toolkit: An open source software system for speech and time-series processing. In *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, volume 4, pages IV–3916–IV–3919, 2002.

Jeff A. Bilmes, Geoff Zweig, Thomas Richardson, Karim Filali, Karen Livescu, Peng Xu, Kirk Jackson, Yigal Brandman, Eric Sandness, Eva Holtz, Jerry Torres, and Bill Byrne. Discriminatively structured graphical models for speech recognition: JHU-WS-2001 final workshop report. Technical report, CLSP, Johns Hopkins University, Baltimore, MD, 2001.

Özgür Çetin. *Multi-rate Modeling, Model Inference, and Estimation for Statistical Classifiers*. PhD thesis, University of Washington, 2004.

Özgür Çetin and Mari Ostendorf. Multi-rate and variable-rate modeling of speech at phone and syllable time scales. *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1:665–668, 18-23 2005.

Shuangyu Chang. *A Syllable, Articulatory-Feature, and Stress-Accent Model of Speech Recognition*. PhD thesis, University of California, Berkeley, 2002.

B. Chen, Q. Zhu, and N. Morgan. Learning long-term temporal features in lvcsr using neural networks. In *Proceedings of International Conference on Spoken Language Processing*, pages 612–615, Jeju, Korea, October 2004.

C. Chen, K. Filali, and J. Bilmes. Frontend post-processing and backend model enhancement on the Aurora 2.0/3.0 databases. In *Intl. Conf. on Spoken Language Proc.*, pages 393–396, 2002.

Chia-Ping Chen, J. Bilmes, and D. Ellis. Speech feature smoothing for robust ASR. In *Proc. of ICASSP*, pages 525–528, 2005.

Amy Dashiell, Brian Hutchinson, Anna Margolis, and Mari Ostendorf. Non-segmental duration feature extraction for prosodic classification. In *Proc. of Interspeech*, Brisbane, Australia, September 2008.

Steven B. Davis and Paul Mermelstein. Evaluation of acoustic parameters for monosyllabic word identification. *The Journal of the Acoustical Society of America*, 64:S180–S181, Nov. 1978.

T. Dean and K. Kanazawa. A model for reasoning about persistence and causation. *Artificial Intelligence*, 93(1-2): 1–27, 1989.

Neeraj Deshmukh, Aravind Ganapathiraju, Andi Gleeson, Jonathan Hamaker, and Joseph Picone. Resegmentation of SWITCHBOARD. In *Proceedings of the International Conference Spoken Language Processing*, volume 4, page 1543, 1998.

Rob Drullman, Joost M. Festen, and Reinier Plomp. Effect of temporal envelope smearing on speech reception. *Journal of the Acoustical Society of America*, 95(2):1053–1064, February 1994.

Stéphane Dupont, Hervé Bourlard, and Christophe Ris. Using multiple time scales in a multi-stream speech recognition system. In *Proc. of the European Conf. on Speech Communication and Technology*, 1997.

Jonathan G. Fiscus. A post-processing system to yield reduced word error rates: recognizer output voting error reduction (ROVER). In *Proc. of the 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 347–354, 1997.

Eric Fosler-Lussier, Steven Greenberg, and Nelson Morgan. Incorporating contextual phonetics into automatic speech recognition. In *Proceedings of the International Congress of Phonetic Sciences*, 1999.

J. Frankel, M. Magimai-Doss, S. King, K. Livescu, and Ö. Çetin. Articulatory feature classifiers trained on 2000 hours of telephone speech. In *Proc. of Interspeech*, Antwerp, Belgium, 2007.

M. Fujimoto, K. Ishizuka, and T. Nakatani. A voice activity detection based on the adaptive integration of multiple speech features and a signal decision scheme. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Las Vegas, NV, April 2008.

Masakiyo Fujimoto and Kentaro Ishizuka. Noise robust voice activity detection based on switching kalman filter. *IEICE Trans. on Information & Systems*, E91-D(3):467–477, 2008.

O. Fujimura. Syllable as a unit of speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 23(1):82–87, Feburary 1975.

A. Ganapathiraju, V. Goel, J. Picone, A. Corrada, G. Doddington, K. Kirchhoff, M. Ordowski, and B. Wheatley. Syllable-a promising recognition unit for LVCSR. In *Proc. of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 207–214, Dec 1997.

Dan Geiger and David Heckerman. Knowledge representation and inference in similarity networks and bayesian multinets. *Artif. Intell.*, 82(1-2):45–74, 1996. ISSN 0004-3702.

J. Godfrey, E. Holliman, and J. McDaniel. Switchboard: Telephone speech corpus for research and development. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 517–520, San Francisco, California, March 1992.

P.D. Green, N. R. Kew, and D. A. Miller. Speech representations in the SYLK recognition project. In Martin Cooke, Steve Beet, and Malcolm Crawford, editors, *Visual Representation of Speech Signals*, chapter 26, pages 265–272. John Wiley & Sons, 1993.

S. Greenberg, J. Hollenback, and D. Ellis. Insights into spoken language gleaned from phonetic transcription of the Switchboard corpus. In *Proc of ICSLP*, pages 24–27, Philadelphia, 1996.

S. Greenberg, Takayuki Arai, and Rosaria Silipo. Speech intelligibility derived from exceedingly sparse spectral information. In *Proc. of ICSLP*, pages 74–77, 1998.

S. Greenberg, T. Arai, and K. Grant. The role of temporal dynamics in understanding spoken language. In P. Divenyi, K. Vicsi, and G. Meyer, editors, *Dynamics of Speech Production and Perception*, pages 171–192. Amsterdam: IOS Press, 2006.

Steven Greenberg. Speaking in shorthand – a syllable-centric perspective for understanding pronunciation variation. In *Proceedings of the ESCA Workshop on Modeling Pronunciation Variation for Automatic Speech Recognition*, pages 47–56, 1998.

Steven Greenberg. Speaking in shorthand – a syllable-centric perspective for understanding pronunciation variation. *Speech Communication*, 29:159–176, 1999.

Steven Greenberg and Takayuki Arai. What are the essential cues for understanding spoken language? *IEICE Trans. Inf. & Syst.*, E87:1059–1070, 2004.

T. Hain, P.C. Woodland, T.R. Niesler, and E.W.D. Whittaker. The 1998 HTK system for transcription of conversational telephone speech. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 57–60, March 1999.

H. Hermansky and N. Morgan. RASTA processing of speech. *Speech and Audio Processing, IEEE Transactions on*, 2(4):578–589, Oct 1994.

H. Hermansky and S. Sharma. TRAPs - Classifiers of temporal patterns. In *Proc. of the Int. Conf. on Spoken Language (ICSLP)*, pages 1003–1006, 1998.

H. Hermansky and S. Sharma. Termporal patterns TRAPs in ASR of noisy speech. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 289–292, 1999.

H. Hermansky, D. Ellis, and S. Sharma. Tandem connectionist feature extraction for conventional HMM systems. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 3, pages 1635–1638, Istanbul, June 2000.

J. L. Hieronymus, D. McKelvie, and F. McInnes. Use of acoustic sentence level and lexical stress in HSMM speech recognition. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 225–227, San Francisco, 1992.

Xuedong Huang, Alex Acero, and Hsiao-Wuen Hon. *Spoken Language Processing*. Prentice Hall PTR, 2001.

M.J. Hunt, M. Lennig, and P. Mermelstein. Experiments in syllable-based recognition of continuous speech. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 5, pages 880–883, April 1980.

Gang Ji, Jeff Bilmes, Jeff Michels, Katrin Kirchhoff, and Chris Manning. Graphical model representations of word lattices. In *IEEE/ACL 2006 Workshop on Spoken Language Technology (SLT2006)*, pages 162–165, Palm Beach, Aruba, Dec 2006.

Simon King, Chris Bartels, and Jeff Bilmes. SVitchboard: Small-vocabulary tasks from switchboard. In *Proc. of the European Conf. on Speech Communication and Technology*, pages 3385–3388, 2005.

Katrin Kirchhoff. Syllable-level desynchronisation of phonetic features for speech recognition. In *Proc. ICSLP '96*, volume 4, pages 2274–2276, Philadelphia, PA, 1996.

Xiao Li. *Regularized Adaptation: Theory, Algorithms and Applications*. PhD thesis, University of Washington, 2007.

Hui Lin, Alex Stupakov, and Jeff Bilmes. Spoken keyword spotting via multi-lattice alignment. In *Proceedings of Interspeech*, Brisbane, Australia, September 2008.

Hui Lin, Alex Stupakov, and Jeff Bilmes. Improving multi-lattice-alignment based spoken keyword spotting. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4877–4880, Taipei, Taiwan, 2009.

Karen Livescu, Özgür Çetin, Mark Hasegawa-Johnson, Simon King, Chris Bartels, Nash Borges, Arthur Kantor, Partha Lal, Lisa Yung, Ari Bezman, Stephen Dawson-Haggerty, and Bronwyn Woods. Articulatory feature-based methods for acoustic and audio-visual speech recognition: Summary from the 2006 JHU summer workshop. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 2485–2488, Honolulu, HI, April 2007.

Jon Malkin and Jeff Bilmes. Multi-layer ratio semi-definite classifiers. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4465–4468, Taipei, Taiwan, 2009.

Jonathan Malkin and Jeff Bilmes. Ratio semi-definite classifiers. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4113–4116, Las Vegas, NV, April 2008.

Paul Mermelstein. Automatic segmentation of speech into syllabic units. *The Journal of the Acoustical Society of America*, 58:880–883, Oct. 1975.

Nelson Morgan and Eric Fosler-Lussier. Combining multiple estimators of speaking rate. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 2, pages 729–732, Seattle, WA, 1998.

J.A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.

J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 2nd printing edition, 1988.

T. Pfau and G. Ruske. Estimating the speaking rate by vowel detection. In *Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, volume 2, pages 945–948, 1998.

H. R. Pfitzinger, S. Burger, and S. Heid. Syllable detection in read and spontaneous speech. In *Proc. ICSLP '96*, volume 2, pages 1261–1264, Philadelphia, PA, 1996.

S. Reynolds and Jeff Bilmes. Part-of-speech tagging using virtual evidence and negative training. In *Human Language Technology Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP-2005)*, pages 459–466, Vancouver, B.C., Oct 2005.

Ken Schutte and James Glass. Robust detection of sonorant landmarks. In *9th European Conf. on Speech Communication and Technology (Eurospeech)*, pages 1005–1008, 2005.

L. Shastri, S. Chang, and S. Greenberg. Syllable detection and segmentation using temporal flow neural networks. In *Proc. of the 14th International Congress of Phonetic Sciences*, pages 1721–1724, 1999.

Michael Shire. Syllable onset detection from acoustics. Master's thesis, University of California, Berkeley, May 1997.

J. Sohn, N. Kim, and W. Sung. A statistical model-based voice activity detection. *IEEE Signal Processing Letters*, 6: 1–3, 1999.

A. Subramanya, J. Bilmes, and C. Chen. Focused word segmentation for ASR. In *9th European Conf. on Speech Communication and Technology (Eurospeech)*, pages 393–396, 2005.

Chao Wang and Stephanie Seneff. Lexical stress modeling for improved speech recognition of spontaneous telephone speech in the jupiter domain. In *Proc. of Eurospeech*, pages 353–356, 2001.

Dagen Wang and Shrikanth Narayanan. Speech rate estimation via temporal correlation and selected sub-band correlation. In *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 413–416, Philadelphia, PA, March 2005.

Dagen Wang and Shrikanth Narayanan. Robust speech rate estimation for spontaneous speech. *IEEE Transactions on Speech, Audio and Language Processing*, 2007.

Su-Lin Wu. *Incorporating Information from Syllable-length Time Scales into Automatic Speech Recognition*. PhD thesis, University of California, Berkeley, Spring 1998.

Su-Lin Wu, Michael L. Shire, Steven Greenberg, and Nelson Morgan. Integrating syllable boundary information into speech recognition. In *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 987–990, Munich, 1997.

Su-Lin Wu, Brian E.D. Kingsbury, Nelson Morgan, and Steven Greenberg. Incorporating information from syllable-length time scales into automatic speech recognition. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 721–724, May 1998a.

Su-Lin Wu, Brian E.D. Kingsbury, Nelson Morgan, and Steven Greenberg. Performance improvements through combining phone- and syllable-length information in automatic speech recognition. In *Proceedings of the 4th International Conference on Spoken Language Processing*, pages 854–857, 1998b.

G. S. Ying, L. H. Jamieson, R. Chen, and C. D. Mitchell. Lexical stress detection on stress-minimal word pairs. In *Proc. ICSLP '96*, volume 3, pages 1612–1615, Philadelphia, PA, 1996.

S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland. *The HTK Book, version 3.3*, 2005.

G. Zweig and S. Russell. Speech recognition with dynamic Bayesian networks. *AAAI-98*, 1998.

G. Zweig, J. Bilmes, T. Richardson, K. Filali, K. Livescu, P. Xu, K. Jackson, Y. Brandman, E. Sandness, E. Holtz, J. Torres, and B. Byrne. Structurally discriminative graphical models for automatic speech recognition: Results from the 2001 johns hopkins summer workshop. In *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, June 2002*, volume 1, pages I–93 – I–96, Orlando, Florida, 2002.

Geoffrey G. Zweig. *Speech Recognition with Dynamic Bayesian Networks*. PhD thesis, University of California, Berkeley, Spring 1998.

E. Zwicker, E. Terhardt, and E. Paulus. Automatic speech recognition using psychoacoustic models. *The Journal of the Acoustical Society of America*, 65:487–498, February 1979.