PRONUNCIATION AND SILENCE PROBABILITY MODELING FOR ASR

Guoguo Chen[1]*, Hainan Xu*[1]*, Minhua Wu*[1]*, Daniel Povey*[1,2]*, Sanjeev Khudanpur*[1,2]

[1]Center for Language and Speech Processing
[2]Human Language Technology Center of Excellence
The Johns Hopkins University, Baltimore, MD 21218, USA

guoguo@jhu.edu, hxu31@jhu.edu, mwu56@jhu.edu, dpovey@gmail.edu, khudanpur@jhu.edu

Abstract

In this paper we evaluate the WER improvement from modeling pronunciation probabilities and word-specific silence probabilities in speech recognition. We do this in the context of Finite State Transducer (FST)-based decoding, where pronunciation and silence probabilities are encoded in the lexicon (L) transducer. We describe a novel way to model word-dependent silence probabilities, where in addition to modeling the probability of silence following each individual word, we also model the probability of each word appearing after silence. All of these probabilities are estimated from aligned training data, with suitable smoothing. We conduct our experiments on four commonly used automatic speech recognition datasets, namely Wall Street Journal, Switchboard, TED-LIUM, and Librispeech. The improvement from modeling pronunciation and silence probabilities is small but fairly consistent across datasets.

Index Terms: automatic speech recognition, pronunciation probability, silence probability

1. Pronunciations and Inter-Word Silence

A key component of an automatic speech recognition (ASR) system is the pronunciation lexicon, which rewrites each word in a potentially large vocabulary in terms of a relatively small number of phonetic units. Such lexicons contain a single pronunciation for most words, even though it is widely recognized that everyday speech contains significant deviations from canonical pronunciations [1]. Some, including [2], have argued for *explicit* pronunciation modeling through a data-driven expansion of the lexicon. Others have argued that such pronunciation variation should be modeled *implicitly* via context-dependent acoustic models [3].

Explicit pronunciation modeling entails creating multiple pronunciations for each (or most) word(s) in the lexicon. In [4], the acoustic model training corpus is decoded with an automatic phone recognizer to obtain frequent alternative pronunciations of frequent words. In [5] and [6], phonological rules are used to generate alternative pronunciations of words, while in [7] and [8], statistical decision trees for the same purpose. Another aspect of explicit pronunciation modeling is the estimation of the probabilities of alternative pronunciations. In [9] and [10] pronunciation probabilities are estimated according to their relative frequency in the training data, while in [11] pronunciation probabilities depend on dynamic features such as speaking rate, segment durations, pitch, etc.

This work was partially supported by NSF Grant No IIS 0963898, DARPA BOLT Contract No HR0011-12-C-0015 and an unrestricted gift from Google Inc. (No 2012-R2-106).

Implicit pronunciation modeling relies on the underlying acoustic-phonetic models to account for pronunciation variations, and therefore removes the necessity to explicitly determine and represent them in the lexicon. In some methods, acoustic model parameters of a phoneme (e.g., Gaussian densities) are tied with those of its alternative realizations, thus capturing alternative pronunciations [3, 12, 13]. Others view pronunciations as a bundle of features, and pronunciation variation is viewed as feature-change or asynchrony [14, 15].

While variability in the pronunciation of individual words has been studied extensively, relatively little has been studied about inter-word silence and its dependence on the prosodic and syntactic structure of the utterance. In [3, 10], for instance, three types of silence are permitted following each pronunciation in the lexicon: a zero-silence, a short pause and a long silence. It is demonstrated empirically that permitting such options for inter-word silence improves ASR performance. But there is no mechanism to predict, for instance, that zero-silence is less likely to follow the word *White* in "*Gandalf the White said*," than in "*The White House said*."

We propose to explicitly model the probability of inter-word silence. As a first approximation, we will ignore syntactic and prosodic structure, and condition the probability of inter-word silence on only the identities of the surrounding words. We will estimate this probability from data.

We make three contributions through this paper. We re-visit the use of pronunciation probabilities in the lexicon [3, 9, 10], and demonstrate empirically on multiple datasets that it is consistently beneficial. We then propose a novel word-dependent estimate of the probability of inter-word silence. Finally, we propose a method to incorporate the inter-word silence probabilities in a finite state transducer (FST) framework, permitting easy implementation in an FST-based decoder, and demonstrate further improvement in ASR performance from it.

The remainder of this paper is organized as follows. We briefly describe the generation of training data for pronunciation and silence modeling in Section 2. We then explain how we estimate pronunciation probabilities in Section 3 and word-dependent inter-word silence probabilities in Section 4. We describe how we encode pronunciation and silence probabilities via an FST in Section 5. The experimental setup is detailed in Section 6, and results are provided in Section 7. Finally we reiterate our main claims in Section 8.

2. Training Data Alignment

We estimate the pronunciation probabilities and word-dependent silence probabilities using the acoustic model training data. The original lexicon (without pronunciation probabilities) is used to train an early stage triphone system, with which

we align all the data. Bigram-like counts are collected for word-pronunciation pairs, including word-specific counts of (an optional) silence following each pronunciation. These will be used to estimate pronunciation probabilities and word-dependent silence probabilities, as described in Sections 3 and 4.

3. Pronunciation Probability Estimation

3.1. Pronunciation probabilities

We estimate the pronunciation probabilities for a word with multiple pronunciations via simple relative frequency [3, 9, 10]. Let $w.p_i$ be the i^{th} pronunciation of word w, $1 \leq i \leq N_w$, and N_w is the number of different *baseform* pronunciations of word w in the lexicon. Let $C(w, w.p_i)$ be the count of "$w\ w.p_i$" pairs in the aligned training data. The probability of a pronunciation $w.p_i$ given the word w is simply

$$\pi(w.p_i|w) = \frac{C(w, w.p_i) + \lambda_1}{\sum_{i=1}^{N_w}(C(w, w.p_i) + \lambda_1)}, \quad (1)$$

where λ_1 is a smoothing constant that we typically set to 1.

3.2. Max-normalization

An undesirable consequence of (1) is that a word with several equiprobable pronunciations is unfairly handicapped w.r.t words that have a single pronunciation: e.g. the past tense of "read" w.r.t the color read "red." Max-normalization, whereby the pronunciation probabilities are scaled so that the most likely pronunciation of each word has "probability" 1, has been found helpful in speech recognition [7]. This suggests using

$$\pi'(w.p_i|w) = \frac{\pi(w.p_i|w)}{\max_{1\leq i\leq N_w} \pi(w.p_i|w)}. \quad (2)$$

We do max-normalization for pronunciation probabilities in all our experiments. The quantity $\pi'(w.p_i|w)$ is of course not a well defined probability any more. It will later be encoded into the lexicon FST in the form of cost, as described in Section 5.

4. Silence Probability Estimation

We next explain how we model the probability of inter-word silence. Recall that inter-word silence is not handled by the language model, because language model training data largely comes from text sources. So the model must be estimated from acoustic model training data, as noted in Section 2.

Let <s>=$w_1\ w_2\ \ldots\ w_{N-2}$=</s> denote an N-length word sequence, including the utterance boundary markers, <s> and </s> . We will take the view that either a silence token s or non-silence token n is stochastically produced by speakers between each pair of consecutive words. Furthermore, we will assume that speakers first generate the word sequence, and then decide where to place inter-word silence, so that the probability of observing an s between w_i and w_{i+1} may be conditioned on the utterance context. However, rather than dwell on prosodic and syntactic phrasing, we will (as a practical first approximation) model inter-word silence as a local decision that depends on only the two surrounding words. For example, the probability of $w_1\ n\ w_2\ s\ w_3\ n\ w_4$ given the word sequence $w_1\ w_2\ w_3\ w_4$ will be $P(n \mid w_1, w_2) \times P(s \mid w_2, w_3) \times P(n \mid w_3, w_4)$.

Furthermore, for words that admit multiple pronunciations, we permit the probability of silence to be dependent on the specific pronunciation that is chosen. Therefore, the generative process above is further refined whereby we assume that speakers first generate the word sequence, then decide which pronunciation to use for each word, and then decide whether to place inter-word silence between each pair of words: with probability $P(s \mid w.p_i, w'.p_j)$.

In a final practical approximation, we implement $P(s \mid w.p_i, w'.p_j)$ as a product of the probability (i) of silence s_r following $w.p_i$, and (ii) of silence s_l preceding $w'.p_j$ given $w.p_i$, and we marginalize the latter probability over $w.p_i$. Much less training data is needed to estimate $P(s_r \mid w.p_i)$ and $P(s_l \mid w'.p_j)$ separately than $P(s \mid w.p_i, w'.p_j)$ jointly. *Crucially, this final approximation enables encoding inter-word silence probabilities directly into the lexicon.*

4.1. Probability of silence to the right of a word

We use the $P(s_r|w.p)$ to denote the probability of inter-word silence following the pronunciation $w.p$, and $P(n_r|w.p)$ to mean the complementary probability of non-silence following $w.p$. We compute $P(s_r|w.p)$ from training data counts as

$$P(s_r|w.p) = \frac{C(w.p\,s) + \lambda_2 P(s)}{C(w.p) + \lambda_2}, \quad (3)$$

where $C(w.p\,s)$ is the count of the sequence "$w.p\,s$" in the training data, $C(w.p)$ is the count of pronunciation $w.p$ in the training data, $P(s) = C(s)/(C(s)+C(n))$ is the overall probability of inter-word silence, and λ_2 is a smoothing constant that we set to 2 for experiments reported here.

4.2. Probability of silence to the left of a word

Our analysis of English conversational speech alignments suggests that the identity of the following word is often a better predictor of inter-word silence (i.e. silence before that word) than of the word preceding the potential inter word silence. Rather than use "trigram" counts $C(w\ s\ w')$ to estimate a joint model of inter-word silence, which cannot be encoded into a lexicon via pronunciation probabilities, we propose a *corrective* model $F(s_l \mid w'.p_j)$ that makes $P(s_r \mid w.p_i) \times F(s_l \mid w'.p_j)$ a good approximation to the joint model $P(s \mid w.p_i, w'.p_j)$.

Specifically, a simple product of the separate empirical estimates of the type $\hat{P}(s_r \mid w.p_i) \times \hat{P}(s_l \mid w'.p_j)$ will *double-count* the occurrence (or lack thereof) of inter-word silence. Yet we still wish to capture via $w'.p_j$ whatever effect is not already modeled by $w.p_i$. To this end, we compute the smoothed correction terms

$$F(s_l|w'.p) = \frac{C(s\ w'.p) + \lambda_3}{\widetilde{C}(s\ w'.p) + \lambda_3}, \text{ and} \quad (4)$$

$$F(n_l|w'.p) = \frac{C(n\ w'.p) + \lambda_3}{\widetilde{C}(n\ w'.p) + \lambda_3}, \quad (5)$$

where $\widetilde{C}(s\ w'.p)$ and $\widetilde{C}(n\ w'.p)$ are the "mean" counts of silence or non-silence preceding $w'.p$, estimated according to

$$\widetilde{C}(s\ w'.p) = \sum_v C(v\ *\ w'.p) P(s_r|v), \quad (6)$$

where the sum is over all pronunciations v in the lexicon, the symbol $*$ in $C(v\ *\ w'.p)$ denotes either s or n, and $P(s_r|v)$ is computed using Equation (3). λ_3 is a smoothing constant that we set to 2 for experiments reported here.

4.3. Putting it all together

The net result of the steps described above are the two context-dependent estimates of inter word silence (or lack thereof):

$$\begin{aligned} P(s \,|\, w.p_i, w'.p_j) &\approx P(s_r \,|\, w.p_i) \times F(s_l \,|\, w'.p_j), \text{ and} \\ P(n \,|\, w.p_i, w'.p_j) &\approx P(n_r \,|\, w.p_i) \times F(n_l \,|\, w'.p_j). \end{aligned}$$

Note that these may not sum to unity. We accept that as the price of being able to easily incorporate inter-word silence probabilities in an FST implementation of a lexicon, as described next.

5. Lexicon Finite-State Transducer

In the context of weighted finite-state transducer (WFST)-based speech recognition, lexicons are represented as FSTs, which map a sequence of phones to a sequence of words. Figure 1 gives a basic lexicon FST (L_1) that allows two vocabulary words: *yes* → [y eh s] and *am* → [ae m].

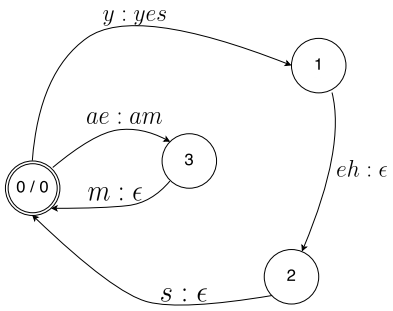

Figure 1: Basic lexicon FST (L_1)

Incorporating pronunciation probabilities into the lexicon FST is trivial, as they can be encoded as arc weights in the FST. Figure 2 illustrates the simplified version of lexicon FST (L_2) used in our pronunciation probability modeling experiments, where negated log-probabilities of pronunciations are added as weights. Note that L_2 also allows optional silence between words. Comparing L_2 with L_1, three more states are added: a start state 0 that represents the beginning of the sentence, a silence state 2 that inserts silence between words, and a disambiguation state 1 adding disambiguation symbol ($\#0$) after inserting silence, which keeps the lexicon FST determinizable [16].

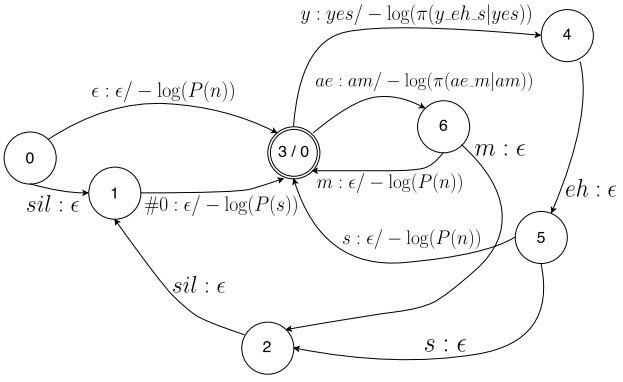

Figure 2: Lexicon FST with optional silence and pronunciation probabilities (L_2)

Figure 3 shows the simplified version of lexicon FST (L_3) used in our silence probability modeling experiments. In Figure 3, optional silence is inserted between words by introducing two special states designated for silence (state 1) and non-silence (state 2), and word-dependent silence probabilities described in Section 4 are encoded as weights (using negated logarithm) of arcs that go into and out from these two special states. Note that in L_3 we add pronunciation cost (negated log probability of pronunciation) in addition to silence cost when we transit from these two special states to words, so that L_3 also models the pronunciation probabilities computed in Section 3. Generally, if the current state is the beginning of the sentence, or the end of some words, then the lexicon FST has to transit either to silence state 1 or to non-silence state 2 first, with the cost that corresponds to the silence probabilities estimated in Section 4.1. It then has to transit back to another word, or to the end of the sentence, with the cost that corresponds to the silence probabilities described in Section 4.2 as well as the pronunciation probabilities described in Section 3. It worth mentioning that in L_3 we avoid the necessity of the disambiguation state by inserting disambiguation symbol ($\#0$) whenever we transit back to the non-silence state 2.

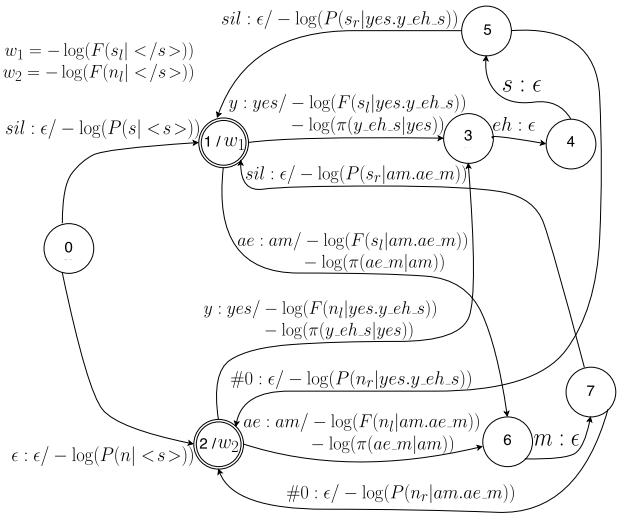

Figure 3: Lexicon FST with word-dependent silence probabilities (L_3)

6. Experiment Setup

This section describes how we conduct our experiments. We use the open source speech recognition toolkit Kaldi [17] for all our implementation and experiments.

6.1. Baseline System

We use Kaldi's online version of time delay neural network (TDNN) recipe [18, 19] as our baselines. We start off by training a GMM system, where 13 dimensional mel-frequency cepstral coefficient (MFCC) [20] features are extracted, followed by a typical maximum likelihood acoustic training recipe that begins with a flat-start initialization of context-independent phonetic HMMs, and ends with speaker adaptive training (SAT). This GMM system is used to align training data and generate labels for neural network training.

We then extract 40 dimensional MFCC features to train an i-vector extractor. We train the i-vector extractor on mean normalized MFCC features, but extract them from MFCC features without mean normalization. This way the i-vectors can capture the mean shifts along input dimensions, which will be available to the neural network so that it can learn necessary normalizations. The i-vector extractor computes 100 dimensional i-vector for each speaker, which will then be appended to the 40 dimensional MFCC features to form features for neural network training.

Combining the training labels and features generated from the above two steps, we create examples for neural network, and perform TDNN training. In the TDNN architecture, splicing over time does not only happen at the input layer of neural network, but also at other layers such as the hidden layers. Besides, the splicing is usually over nonadjacent frames. For more details readers are referred to [18, 19].

6.2. Datasets

Since previous work on pronunciation modeling usually yields modest gains [12], we decide to run our experiments over multiple datasets, to minimize the affect of noise. We conduct our experiments on four commonly used speech recognition datasets, namely the Wall Street Journal (WSJ) corpus [21], the Switchboard (SWBD) corpus [22], the TED-LIUM corpus [23] and the Librispeech corpus [24]. Note that for Librispeech only the 100 hour subset is used in our experiments.

dataset	pron / word ratio	% of multi-pron words
WSJ	1.077	7.2
SWBD	1.024	1.7
TED-LIUM	1.051	4.5
Librispeech	1.033	3.0

Table 1: Lexicon statistics for different datasets

We use the default lexicon in Kaldi recipes for each dataset. The pronunciation statistics of the lexicons are shown in Table 1. From the above table we can see that all lexicons have words with more than one pronunciations, especially for the WSJ lexicon, where 7% of the words have multiple pronunciations. None of the default lexicons comes with pronunciation probabilities.

7. Results

We evaluate the performance using word error rate (WER) in percentage. For all the results presented below, we search for the best word insertion-penalty as well as the best acoustic scale. This is important to show improvements from silence modeling, because our previous model assigned a probability of 0.5 to silence and 0.5 to non-silence between each pair of words, which acted as a crude insertion penalty.

7.1. Impact of retraining

The updated lexicon FST encoded with pronunciation probabilities or word-dependent silence probabilities are typically used for decoding, as done in [19]. They of course can also be used in training, which may improve the alignment and thus reduce WER. Before we run into the full experiments, we first conduct an analysis experiment on SWBD SAT system. Results in Table 2 suggest that retraining or not retraining the acoustic models will not make much a difference. Therefore, in the rest of our experiments, we only use pronunciation probabilities and word-dependent silence probabilities for decoding.

		baseline	+pron model	+sil model
no-retrain	swbd	20.1	20.0	19.6
	eval2000	26.9	26.9	26.7
retrain	swbd	20.1	19.9	19.5
	eval2000	26.9	27.0	26.7

Table 2: WER performance of not retraining and retraining the acoustic models with the updated lexicon FST (SWBD SAT system)

7.2. WER performance on TDNN systems

Table 3 shows the WER performance of using pronunciation probabilities and word-dependent silence probabilities on all the four corpora. As we can see from the table, modeling pronunciation probabilities generally helps to reduce the WER, but at a very modest amount (usually $0.1-0.2\%$, absolute). This is consistent with the conclusions from previous work [12]. Modeling word-dependent inter-word silence probabilities in addition to pronunciation probabilities, which is new, usually brings further improvements. It improves on top of pronunciation probabilities across datasets, except for TED-LIUM and the clean evaluation condition of Librispeech.

		baseline	+pron model	+sil model
WSJ	dev93	6.90	6.68	**6.49**
	eval92	4.11	3.99	**3.95**
SWBD	swbd	13.7	13.6	**13.1**
	eval2000	20.5	20.4	**20.0**
TED-LIUM	dev	20.0	**19.8**	**19.8**
	test	18.1	**17.9**	**17.9**
Librispeech	dev_clean	5.9	**5.8**	**5.8**
	dev_other	21.3	21.3	**21.0**
	test_clean	**6.6**	**6.6**	**6.6**
	test_other	22.9	22.7	**22.5**

Table 3: WER performance of using pronunciation probabilities and word-dependent silence probabilities (TDNN system)

8. Conclusion

We have re-visited the modeling of pronunciation probabilities, including the context-dependent probability of (optional) inter-word silence between specific pronunciations. Experiments on multiple datasets suggest that explicitly modeling pronunciation probabilities usually improves ASR performance in terms of WER, thought the gain is modest. Empirical results also show that modeling word-dependent silence improves recognition performance further on top of the pronunciation probability modeling, and the improvement is fairly consistent across multiple datasets.

This paper is also a step towards joint acoustic-prosodic modeling: the presence of inter-word silence is connected to prosodic phrasing, which in turn is useful for the syntactic analysis of spoken utterances. Syntax in turn is a bridge between spoken words and semantics.

9. References

[1] S. Greenberg, J. Hollenback, and D. Ellis, "Insights into spoken language gleaned from phonetic transcription of the switchboard corpus," in *Proceedings of the International Conference on Spoken Language Processing*, Philadelphia, PA, 1996.

[2] M. D. Riley and A. Ljolje, "Automatic generation of detailed pronunciation lexicons," in *Automatic Speech and Speaker Recognition*, C.-H. Lee, F. Soong, and K. Paliwal, Eds. Kluwer Academic Publishers, 1996, pp. 285–301.

[3] T. Hain, "Implicit modelling of pronunciation variation in automatic speech recognition," *Speech Communication*, vol. 46, no. 2, pp. 171–188, 2005.

[4] T. Slobada and A. Waibel, "Dictionary learning for spontaneous speech recognition," in *Proceedings of Fourth International Conference on Spoken Language Processing (ICSLP)*, vol. 4. IEEE, 1996, pp. 2328–2331.

[5] E. Giachin, A. Rosenberg, and C.-H. Lee, "Word juncture modeling using phonological rules for HMM-based continuous speech recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1990, pp. 737–740.

[6] G. Tajchman, E. Foster, and D. Jurafsky, "Building multiple pronunciation models for novel words using exploratory computational phonology," in *Proceedings of EUROSPEECH*, 1995.

[7] E. Fosler, M. Weintraub, S. Wegmann, Y.-H. Kao, S. Khudanpur, C. Galles, and M. Saraclar, "Automatic learning of word pronunciation from data," in *Proceedings of the International Conference on Spoken Language Processing*, 1996.

[8] B. Byrne, M. Finke, S. Khudanpur, J. McDonough, H. Nock, M. Riley, M. Saraclar, C. Wooters, and G. Zavaliagkos, "Pronunciation modelling for conversational speech recognition: A status report from WS97," in *Proceedings of the Automatic Speech Recognition & Understanding (ASRU) Workshop*. IEEE, 1997, pp. 26–33.

[9] B. Peskin, M. Newman, D. McAllaster, V. Nagesha, H. Richards, S. Wegmann, M. Hunt, and L. Gillick, "Improvements in recognition of conversational telephone speech," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1. IEEE, 1999, pp. 53–56.

[10] T. Hain, P. Woodland, G. Evermann, and D. Povey, "The CU-HTK march 2000 Hub5e transcription system," in *Proceedings of Speech Transcription Workshop*, vol. 1, 2000.

[11] M. Finke and A. Waibel, "Speaking mode dependent pronunciation modeling in large vocabulary conversational speech recognition," in *Proceedings of EUROSPEECH*, 1997.

[12] M. Saraclar, H. Nock, and S. Khudanpur, "Pronunciation modeling by sharing Gaussian densities across phonetic models," *Computer Speech & Language*, vol. 14, no. 2, pp. 137–160, 2000.

[13] M. Saraclar and S. Khudanpur, "Pronunciation ambiguity vs. pronunciation variability in speech recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3. IEEE, 2000, pp. 1679–1682.

[14] M. Finke, J. Fritsch, D. Koll, and A. Waibel, "Modeling and efficient decoding of large vocabulary conversational speech." in *Proceedings of Eurospeech*, 1999.

[15] K. Livescu and J. Glass, "Feature-based pronunciation modeling for speech recognition," in *Proceedings of HLT-NAACL: Short Papers*. Association for Computational Linguistics, 2004, pp. 81–84.

[16] M. Mohri, F. Pereira, and M. Riley, "Speech recognition with weighted finite-state transducers," in *Springer Handbook of Speech Processing*. Springer, 2008, pp. 559–584.

[17] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *Proceedings of the Automatic Speech Recognition & Understanding (ASRU) Workshop*, 2011.

[18] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," submitted to INTERSPEECH 2015. [Online]. Available: http://speak.clsp.jhu.edu/uploads/publications/papers/1048_pdf.pdf

[19] V. Peddinti, G. Chen, D. Povey, and S. Khudanpur, "An i-vector based time delay neural network architecture for far field recognition," *submitted to INTERSPEECH 2015*, 2015. [Online]. Available: http://speak.clsp.jhu.edu/uploads/publications/papers/1049_pdf.pdf

[20] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[21] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 357–362.

[22] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "SWITCHBOARD: Telephone speech corpus for research and development," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1. IEEE, 1992, pp. 517–520.

[23] A. Rousseau, P. Deléglise, and Y. Estève, "TED-LIUM: an automatic speech recognition dedicated corpus." in *Proceedings of LREC*, 2012, pp. 125–129.

[24] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015.