LEXICAL MODELING OF NON-NATIVE SPEECH FOR AUTOMATIC SPEECH RECOGNITION

Karen Livescu and James Glass

Spoken Language Systems Group
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139 USA

ABSTRACT

This paper examines the recognition of non-native speech in JUPITER, a speaker-independent, spontaneous-speech conversational system. Because the non-native speech in this domain is limited and varied, speaker- and accent-specific methods are impractical. We therefore chose to model all of the non-native data with a single model. In particular, this paper describes an attempt to better model non-native lexical patterns. These patterns are incorporated by applying context-independent phonetic confusion rules, whose probabilities are estimated from training data. Using this approach, the word error rate on a non-native test set is reduced from 20.9% to 18.8%.

1. INTRODUCTION

Speech recognition accuracy has been observed to be drastically lower for non-native speakers of the target language than for native speakers [3, 13, 14]. Research on both non-native accent modeling and dialect-specific modeling shows that large gains in performance can be achieved when the acoustics [1, 9, 14] and pronunciation [5, 7, 13] of a new accent or dialect are taken into account. Non-native accents are more problematic than dialects because there is a larger number of non-native accents for any given language and because the variability among speakers of the same non-native accent is potentially much greater than among speakers of the same dialect due to different levels of familiarity with the target language and individual tendencies.

Previous work with non-native speech has involved modeling either a particular accent or a particular speaker. The work described in this paper deals with the speech recognizer used in JUPITER [2], a speaker-independent, spontaneous-speech conversational system that interacts with non-native speakers from many diverse backgrounds. Furthermore, there is a relatively small amount of non-native JUPITER data—the corpus used here contains 5,146 non-native utterances (4,339 of which contain only in-vocabulary words), compared to 62,324 native utterances (46,036 in-vocabulary). In this situation, it is impractical to model each non-native accent separately or classify the many accents. The possibilities for speaker adaptation are also limited, as each speaker's interaction with the system is typically only a few utterances long.

The main goal of this work, therefore, is to discover what performance gains can be obtained by modeling all non-native speech with a single model. The following sections describe the methods we have used to automatically discover non-native pronunciation variations, as well as the results of recognition experiments applying these methods.

This work was supported by the National Science Foundation through a Graduate Research Fellowship and under Grant No. IRI-9618731.

2. LEXICAL MODELING

In this section we explore modifications to the lexicon to incorporate non-native pronunciations. Ideally, we would like to collect entire word pronunciations and train their probabilities from real non-native data. However, since there are not enough instances of each word in the training set, we instead attempt to derive rules from the data, which we then apply to the baseline lexicon. We present our methods and findings for a very simple type of rule, namely context-independent phonetic confusions. Although context-dependent rules would contain more information, the larger required number of parameters would be difficult to train from the limited amount of training data.

2.1. Modeling Pronunciation Patterns Using Finite-State Transducers

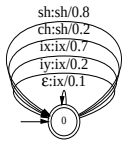
The recognition engine used in this work is a finite-state transducer-based version of the SUMMIT segment-based recognition system [2]. The baseline recognizer search space can be represented as the composition of several FST's:

$$R = P \circ L \circ G,$$

where P is a phonetic graph with associated acoustic scores; L is the lexicon; and G is the language model. Non-native pronunciation rules are incorporated by introducing an additional FST, C, between the lexicon and phonetic graph, so that the modified search space can be represented as

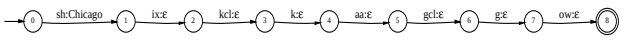
$$R_C = P \circ C \circ L \circ G$$

C can represent any number of phenomena, and its arc weights represent the probabilities of these phenomena. In the case of context-independent confusion rules, C consists of a single state, with self-loops representing allowed confusions. A portion of C may look like the following:

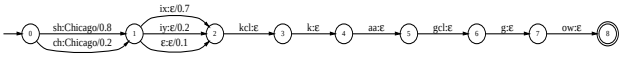


An arc labeled $(x : y)/p$ indicates that the *lexical label* y can be realized as the *surface label* x with probability p. This FST indicates, for example, that a lexical [sh] can be realized as a surface [sh] with probability 0.8 or as a [ch] with probability 0.2. Insertions and deletions are represented as transitions to or from ε, the null unit. A full C would contain at least one arc for every lexical label, so that each lexical pronunciation has at least one surface realization. We refer to C as a *confusion FST* or *CFST*. In the present work, the arc probabilities are estimated from training data.

The effect of composing C with L is to add to each arc in L a set of parallel arcs corresponding to the possible realizations of the lexical phone on that arc. For example, a lexicon containing only the word *Chicago* with one pronunciation may look like the following:



If C contains only the confusions shown above, plus self-mappings for all of the other lexical labels, then $C \circ L$ is:



This approach is similar in some respects to previous work in lexical modeling. In [6], Levinson *et al.* obtained word hypotheses by aligning the outputs of a phonetic recognizer with the lexicon and grammar. However, the confusion weights were determined using an acoustic similarity measure. The work of Teixeira *et al.* [13] is more similar to the current approach, in that the pronunciation weights are estimated from training data. In [11], Riley and Ljolje train probabilistic, context-dependent phoneme-to-phone mappings to obtain a phonetic lexicon for native speakers. Finally, our approach is similar to that of ANGIE [12], a subword lexical modeling framework in which phonological rules have trainable probabilities.

2.2. Estimation of Confusion Probabilities

In order to estimate the confusion probabilities in C, we need a phonetic transcription for each utterance in the training set, aligned with the corresponding pronunciation of each word as it appears in the lexicon. In our approach, transcriptions are generated automatically (see Section 3) and aligned with the lexicon using an automatic string alignment procedure. Finally, the maximum likelihood (ML) estimates of the confusion probabilities are computed from the frequencies of confusions in the alignments.

For substitutions and deletions, if the lexical label l occurs n_l times, and the confusion $(s : l)$ occurs $n_{s:l}$ times, the ML estimate of the probability of $(s : l)$ is

$$\hat{P}(s : l) \equiv \hat{P}(s|l) = \frac{n_{s:l}}{n_l}$$

In the case of insertions, on the other hand, we need to take the *a priori* probability of an insertion into account. The ML estimate of this *a priori* probability is

$$\hat{P}(ins) = \frac{n_{ins}}{n_{tot}},$$

where n_{ins} is the number of insertions and n_{tot} is the total number of aligned phones. Given that an insertion occurs, the estimated probability of the inserted phone being s is

$$\hat{P}(s|ins) = \frac{n_{s,ins}}{n_{ins}},$$

where $n_{s,ins}$ is the number of s insertions. The total estimated probability of an s insertion is then

$$\begin{aligned} \hat{P}(s : \varepsilon) \equiv \hat{P}(s, ins) &= \hat{P}(s|ins)\hat{P}(ins) \\ &= \frac{n_{s,ins}}{n_{tot}} \end{aligned}$$

2.3. Computational Details

The composition of a lexicon and a CFST can be larger than the lexicon by a factor of up to N_s, the number of surface labels. This can make both the time and the space requirements of the recognizer prohibitively large. Therefore, instead of precomputing the composition $C \circ L \circ G$, we precompute $L \circ G$ and compose the result with C dynamically during recognition. In dynamic composition, portions of the search space R_C are created as necessary to expand the hypotheses being considered.

In order to further limit the size of R_C, we prune C by including only those arcs whose probabilities are above a given threshold. In addition, in some of the experiments, we reduce time and space requirements by using a narrower beam in the recognition search than the baseline recognizer does. While this initially increases the error rate, it allows us to experiment with a larger range of CFST sizes.

Due to memory and computational constraints, we do not smooth the probabilities to account for sparse training data. However, to ensure that the baseline pronunciation of each word is allowed, we include all of the self-confusions $(l : l)$ with some minimum probability.

3. EXPERIMENTS

This section describes recognition experiments performed with CFST's. For these experiments, the 4,339 in-vocabulary non-native utterances were divided into a 2,717-utterance training set, a 609-utterance development set for parameter tuning, and a 1,013-utterance test set.

The recognizer uses diphone acoustic models, using as features the first 14 MFCC's averaged over 8 regions near

each boundary in the segmentation. The diphones are modeled using diagonal Gaussian mixtures with up to 50 components per model, trained on a set containing 33,692 native utterances and the non-native training set. The basic lexical units consist of 61 phone labels. The lexicon contains 1,956 words, many with several alternative pronunciations. The language model is a word trigram. This configuration is similar to the one in [2].

The baseline recognizer achieves a word error rate (WER) of 20.9% on the non-native test set and 10.5% on a native test set.

3.1. Probability Estimation from Phonetic Recognition Hypotheses

In the first set of experiments, the transcription of each training utterance is simply the best hypothesis produced by a phonetic recognizer. The phonetic recognizer uses the same acoustic models as the word recognizer. In order to constrain the transcriptions as little as possible, while minimizing obvious transcription errors, a phone bigram language model is used for this task. The alignments are performed using equal weights for all substitutions, deletions, and insertions. We refer to this as the "phonetic recognition (PR) method." It is similar to methods used by others to derive transcriptions for training of pronunciation rules [4].

Using this method on the non-native training set, we obtain a CFST, C_{PR}, containing 2188 confusions (out of a maximum of $62^2 - 1 = 3843$, since there are 62 phone labels including the null label).

We tested CFST's derived from C_{PR} with varying pruning thresholds. The threshold `cprune` is expressed as a negative log probability, so that the higher the threshold, the larger the CFST. Figure 1 shows results obtained on the non-native test set for `cprune` $\in [0, 6]$. At `cprune` $= 0$, the CFST is an identity mapping; at `cprune` $= 6$, it contains about half the confusions in C_{PR}. The figure shows two series of results, using different beam widths (`vprune`) in the recognition (Viterbi) search. We tested with both beam widths because, on the development set, WER reductions were obtained with both (from a baseline of 20.2% to 18.9% at `vprune` $= 15$ and 18.2% at `vprune` $= 20$); the same improvements were not found on the test set, however.

For both `vprune` $= 15$ and `vprune` $= 20$, there is a minimum in WER at `cprune` $= 4$. The increase in WER for `cprune` > 4 may indicate that the low-probability arcs are not well-trained, so that adding them increases the confusability between words. The lowest WER obtained in this series of experiments is 20.3% at `vprune` $= 20$, `cprune` $= 4$. This difference, however, is not significant according to a matched-pair sentence-segment test [10].

3.2. Probability Estimation from Forced Paths

The PR method results in large CFST's whose accuracy is limited by that of the alignments. Furthermore, there is reason to believe that the CFST's created this way are *unnecessarily* large, since the phonetic recognizer does not use all of the information that the word recognizer has available to it. In the PR method, we build the CFST that the word recognizer would need if the word sequence were constrained to match a *particular* string of phones. However, the word

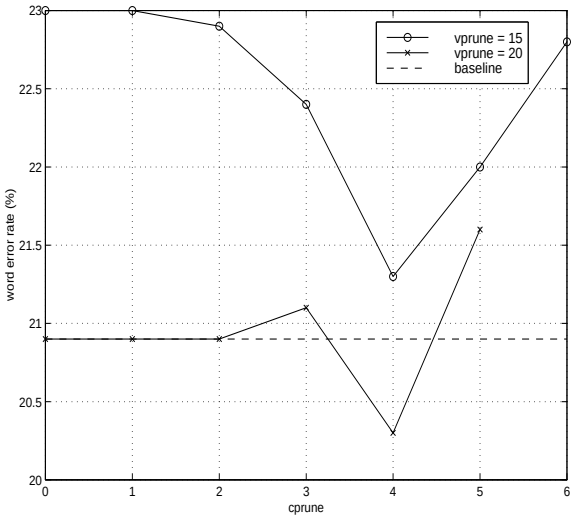


Figure 1: Word error rate on the non-native test set as a function of CFST pruning threshold, using the PR method.

recognizer has the entire phonetic graph at its disposal, and can search for alternate phones that better match the lexicon when necessary. For this reason, we may not need to expand the lexicon to such a great extent.

An alternate approach that uses the entire phone graph is to generate *forced* transcriptions using a lexicon consisting of the known word string for each utterance, expanded with a pre-existing CFST C_0. In other words, each transcription is the best path through the FST

$$R_{FP} = P \circ C_0 \circ W,$$

where P is the phone graph and W represents the known word transcription of the utterance and its baseline pronunciations. For C_0, we used C_{PR}, pruned with a threshold of 6.5 and padded with a minimum probability for each self-confusion. The pruning threshold was chosen so that transcriptions were computed in a reasonable time (i.e., less than 10x real-time), while allowing most of the confusions in C_{PR}. We refer to this as the "forced path (FP) method."

Using this method, we obtain a CFST, C_{FP}, containing 840 confusions. This is a large decrease relative to the PR method. From a visual inspection of the C_{FP}, the confusion statistics appear to conform better with our expectations: the probability mass is more concentrated in the more likely confusions, and many of the expected non-native confusions, such as $(iy : ih)$ and $(uw : uh)$, receive higher probability estimates than with the PR method.

Figure 2 shows the WER's obtained on the non-native test set using CFST's with `cprune` $\in [0, 12]$. At `cprune` $=$ 12, the CFST contains all of the confusions in C_{FP}. In this case, only the narrower search beam (`vprune` $= 15$) was used. This is because, in experiments on the development set, recognition with the wider beam took a prohibitively long time at `cprune` ≥ 8, and the WER was lower with a narrower beam and larger `cprune`. For this reason, the WER's are higher than baseline at the low `cprune`'s. The

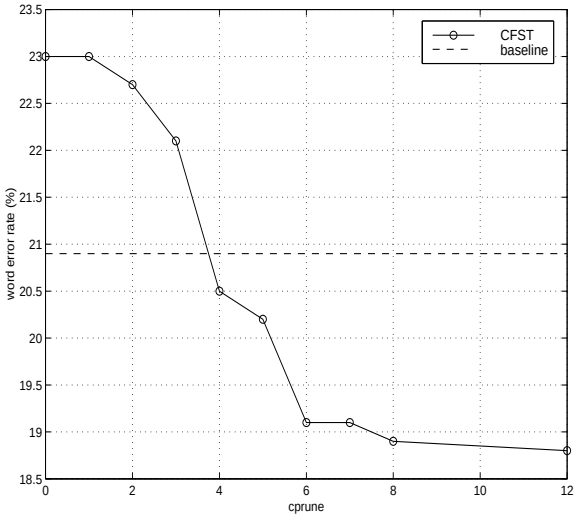


Figure 2: Word error rate on the non-native test set as a function of CFST pruning threshold, using the FP method.

lowest WER is 18.8% at `cprune` = 12. The difference from baseline is significant at the 0.001 level according to the matched-pair sentence-segment test.

Although this work is aimed at non-native speakers, we have tested the recognizer using C_{FP} with `cprune` = 12 on a native test set as well. Interestingly, this configuration yields the same WER on native speakers as the baseline recognizer. This is an encouraging sign for future work combining native and non-native models.

4. CONCLUSIONS AND FUTURE WORK

The work described in this paper demonstrates that some of the pronunciation patterns of non-native speakers as a group can be modeled with automatically-trained, context-independent phone confusions, represented by a simple finite-state transducer. In order to make the methods more practical, it would be necessary to make computational improvements to reduce the running time and memory requirements.

There are many possible extensions to this work. For example, it may be possible to improve performance by iteratively training [4] or smoothing the confusion probabilities. Our initial attempts at both iterative training and smoothing have not yielded improvements, however [8]. It would also be interesting to train context-dependent rules; this may be feasible as more training data become available or by grouping phone labels into classes in the existing data.

We have emphasized that our immediate goal was to improve the recognition of non-native speakers as a group. Although the possibilities for accent- or speaker-specific modeling are limited in a domain such as JUPITER, some additional gains may be obtained using instantaneous or incremental adaptation during a user interaction or speaker clustering during training.

Finally, an obvious avenue for future work is the combination of native and non-native models in a single recognizer such that performance on each population is as close as possible to that of the population-optimized models.

5. ACKNOWLEDGMENTS

Lee Hetherington and Issam Bazzi provided many helpful suggestions on FST's and confusion probabilities. Sally Lee's initial tagging of accented JUPITER callers made it feasible to divide the data into accent categories.

REFERENCES

[1] V. Diakoloukas, V. Digalakis, L. Neumeyer, and J. Kaja. Development of dialect-specific speech recognizers using adaptation methods. In *Proc. ICASSP*, 1997.

[2] J. R. Glass, T. J. Hazen, and I. L. Hetherington. Real-time telephone-based speech recognition in the JUPITER domain. In *Proc. ICASSP*, 1999.

[3] J. R. Glass and T. J. Hazen. Telephone-based conversational speech recognition in the JUPITER domain. In *Proc. ICSLP*, 1998.

[4] P. Hanna, D. Stewart, and J. Ming. The application of an improved DP match for automatic lexicon generation. In *Proc. Eurospeech*, 1999.

[5] J. Humphries, P. Woodland, and D. Pearce. Using accent-specific pronunciation modelling for robust speech recognition. In *Proc. ICSLP*, 1996.

[6] S. E. Levinson, A. Ljolje, and L. G. Miller. Continuous speech recognition from a phonetic transcription. In *Proc. ICASSP*, 1990.

[7] W. K. Liu and P. Fung. Fast accent identification and accented speech recognition. In *Proc. ICASSP*, 1999.

[8] K. Livescu. Analysis and modeling of non-native speech for automatic speech recognition. Master's thesis, MIT, August 1999.

[9] P. Nguyen, Ph. Gelin, J.-C. Junqua, and J.-T. Chien. N-best based supervised and unsupervised adaptation for native and non-native speakers. In *Proc. ICASSP*, 1999.

[10] D. S. Pallet, W. M. Fisher, and J. G. Fiscus. Tools for the analysis of benchmark speech recognition tests. In *Proc. ICASSP*, 1990.

[11] M. D. Riley and A. Ljolje. Automatic generation of detailed pronunciation lexicons. In C.-H. Lee, F. K. Soong, and K. K. Paliwal, editors, *Automatic Speech and Speaker Recognition*. Kluwer Academic Publishers, Boston, 1996.

[12] S. Seneff, R. Lau, and H. Meng. ANGIE: A new framework for speech analysis based on morpho-phonological modelling. In *Proc. ICSLP*, 1996.

[13] C. Teixeira, I. Trancoso, and A. Serralheiro. Recognition of non-native accents. In *Proc. Eurospeech*, 1997.

[14] G. Zavaliagkos, R. Schwartz, and J. Makhoul. Adaptation algorithms for BBN's phonetically tied mixture system. In *Proc. ARPA Spoken Language Systems Technology Workshop*, 1995.