LEXICAL ACCESS EXPERIMENTS WITH CONTEXT-DEPENDENT ARTICULATORY FEATURE-BASED MODELS

Preethi Jyothi[1]*, Karen Livescu*[2]*, Eric Fosler-Lussier*[1]

[1]Department of Computer Science and Engineering, Ohio State University, Columbus, Ohio
[2]Toyota Technological Institute at Chicago, Chicago, Illinois
{jyothi,fosler}@cse.ohio-state.edu, klivescu@ttic.edu

ABSTRACT

We address the problem of pronunciation variation in conversational speech with a context-dependent articulatory feature-based model. The model is an extension of previous work using dynamic Bayesian networks, which allow for easy factorization of a state into multiple variables representing the articulatory features. We build context-dependent decision trees for the articulatory feature distributions, which are incorporated into the dynamic Bayesian networks, and experiment with different sets of context variables. We evaluate our models on a lexical access task using a phonetically transcribed subset of the Switchboard corpus. We find that our models outperform a context-dependent phonetic baseline.

Index Terms— Lexical access, articulatory features, dynamic Bayesian networks

1. INTRODUCTION

Conversational speech is characterized by a large amount of pronunciation variability. Words in spontaneous speech often have multiple pronunciations that do not conform to the canonical forms, which has been argued to be a cause of the deterioration in speech recognition performance relative to read speech [1, 2, 3]. Most pronunciation models account for this variability by adding alternate phonetic pronunciations to the baseform dictionary, often using phonological rules represented as decision trees (e.g., [4]). An alternative approach is to model speech as multiple streams of sub-phonetic features, rather than a single sequence of phones, which is finer-grained and may be more effective at avoiding word confusability [5, 6]. Recent models in this category have been inspired by ideas from articulatory phonology [7].

We consider an extension of the models in [5], in which a dynamic Bayesian network (DBN) is used to represent multiple streams of variables related to the states of articulatory features (AFs), along with probabilistic constraints on their asynchrony. We replace the context-independent (CI) surface feature distributions of [5] with context-dependent (CD) ones using decision trees, which represent the surface feature distributions given their context such as previous and future feature values.

We evaluate our models against phone-based ones using measures that attempt to isolate the strengths and limitations of pronunciation models from those of the acoustic or language models. One way of evaluating a pronunciation model is to compute its perplexity on unseen test data. This was done for phone-based models in [4] and for articulatory feature-based models in [8]. However, perplexities do not directly predict word classification performance, and also cannot be compared exactly for phone-based and feature-based models. Here we evaluate the models more directly in a lexical access task, as in [5]. In this task, a set of potential word candidates is scored according to the likelihood of the word being predicted, given a fine phonetic transcription representing the actual feature values produced by a speaker. As a test-bed, we use data from Switchboard Transcription Project (STP) [9], a set of conversational speech data that has been manually transcribed at a fine phonetic level.

2. MODELS

2.1. Context-dependent phone baseline model

For a fair comparison against our CD feature-based models, we built a CD phone baseline system based on the model specification in [4]. The hand-labeled phonetic transcriptions from STP were aligned with a phonemic dictionary to give phoneme-to-phone transformations, which were then used to build decision trees for each phoneme. Each phoneme was represented as a six-element feature vector (type of phoneme (vowel/consonant/silence), consonant-manner, consonant-place, vowel-manner, vowel-place, nasal/non-nasal). As in [4], we also allowed deletion of phonemes in context. The context variables included the identity of the phoneme to be mapped as well as three neighboring phonemes on either side and the distance of the phoneme from the word boundary on either side, as per the description in [4]. This phonetic baseline was implemented with finite-state transducers using the OpenFST toolkit [10].

This research was supported by the NSF grants IIS-0905633 and IIS-0905420. The opinions expressed in this work are those of the authors and do not necessarily reflect the views of the funding agency.

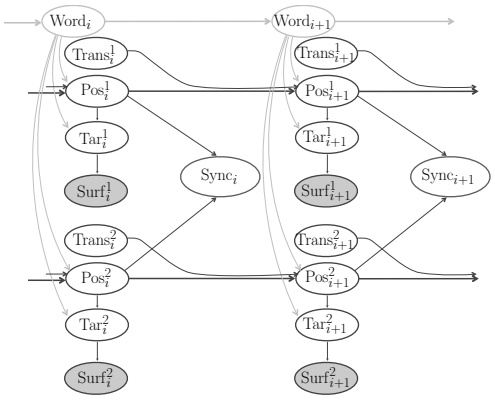

Fig. 1. Context-independent AF model.

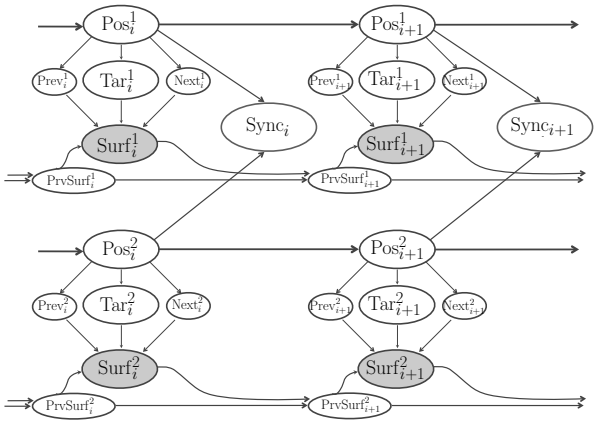

Fig. 2. (Basic CD + Previous Surface value) AF model showing two feature streams. Word and Trans_i^j variables are not shown for visual clarity.

2.2. Context-dependent AF-based models

Fig. 1 shows the structure of the AF-based model of [5] for two feature streams over two (10 ms) frames (in the actual models, there is a larger number of feature streams). Pos_i^j is an index into the underlying phonetic pronunciation of the word for feature j at time frame i. Pos variables range from 0 to $N-1$, where N is the number of phones in the word's pronunciation. Tar_i^j is the corresponding underlying (target) feature value and Surf_i^j is the surface feature value, which is observed in our experiments. In this CI model, the surface value $Surf_i^j$ depends only on the current target value for that feature. In each frame, each feature can either remain in the same state as in the last frame or transition to the next state with some transition probability: $\mathrm{Pos}_{i+1}^j = \mathrm{Pos}_i^j$ if Trans_i^j=0; $\mathrm{Pos}_{i+1}^j = \mathrm{Pos}_i^j + 1$ if Trans_i^j=1. This leads to the possibility of asynchrony between the feature streams, which is constrained by the Sync variables in Fig. 1. The asynchrony probabilities between these feature streams and transition probabilities of the features can be set by hand based on linguistic knowledge, or they can be learned from data. We use both of these options in our experiments, as described in the following section.

The main new feature of our CD model, a version of which is shown in Fig. 2, is that each surface feature value no longer depends only on the target feature value but also on other context variables. In our most basic model, the dependence is on the previous and next distinct target values (Prev_i^j and Next_i^j, respectively [1]. We also experiment with adding the previous distinct surface value, $\mathrm{PrvSurf}_i^j$, as a context variable. In Fig. 2, $\mathrm{PrvSurf}_i^j$ changes to the value of Surf_i^j on a feature transition and retains the value of $\mathrm{PrvSurf}_i^j$ from the previous frame when there is none (the word and transition variables have been excluded from Fig. 2 for visual clarity). Analogously to the phone-based models, we use decision trees to learn the CD surface feature distributions. The interpretable nature of decision trees allows us to verify that the CD substitutions make intuitive sense (for example, nasalization of vowels between nasal consonants), and they can be easily integrated into our DBNs. These models were implemented using the GMTK toolkit [11, 12].

[1]The CI DBN model in [5] maintains some amount of non-determinism in the phone to target feature value mapping. In the CD models, the inclusion of the Next_i^j variables complicates this; for the current work, therefore, we assume a deterministic mapping from target phones to target feature values.

3. EXPERIMENTS AND RESULTS

3.1. Construction of AF decision trees

To generate the articulatory-feature decision trees, we use a subset of STP data consisting of roughly 90,000 10 ms frames of speech. We use 60% of the data to train the decision trees, 20% to tune the decision tree parameters, and the remaining 20% for testing, as in [8].[2] We use Weka's [13] J48 class (a variant of the C4.5 pruned decision tree algorithm) to implement the decision trees. The probability distributions at the leaves are smoothed using Lidstone's law of smoothing, which adds a small value λ to the counts at the leaves. The minimum number of instances at the leaves of the trees and λ were tuned on the development set. As for the phone-based decision trees, an alignment is needed to associate feature values with context variables; for this purpose, we performed a forced alignment using the context-independent DBN model of Fig. 1. We use seven features derived from the vocal tract variables defined in [5, 7] – lip aperture, tongue tip position and aperture, tongue body position and aperture, velum position, and glottis aperture – bundled into three streams as in [8]: all tongue features, glottis/velum and lip aperture.

[2]Our decision tree code is based on code from Sam Bowman. We gratefully acknowledge his code and assistance.

3.2. Perplexity results

For both AF-based and phone-based decision trees, we test several sets of context variables:

1. *Context-independent (CI)*: The context is only the target value for the phone/feature at the current frame.
2. *Basic CD (CD-Basic)*: Same as 1 + previous and next distinct target values.
3. *Context-dependent + previous surface value (CD-Basic + prevSurf)*: Same as 2 + previous distinct surface value of the phone/feature.
4. *Context-dependent + previous surface value + distance (CD-Basic + prevSurf + Dist)*: Same as 3 + distance in frames from the previous/next distinct target value of the phone/feature.

The frame-level perplexity of a data set with T frames is:

$$perp(\mathrm{Surf}_1, \ldots, \mathrm{Surf}_T) = 2^{\frac{-1}{T}\sum_{i=1}^{T} \log_2 p(\mathrm{Surf}_i|\mathrm{Context}_i)}$$

where $p(\mathrm{Surf}_i|\mathrm{Context}_i)$ is given by the phone/feature decision trees. In the case of phonetic trees, Surf_i refers to the surface phone in frame i; for AF trees, Surf_i is shorthand for $\mathrm{Surf}_i^1, \ldots, \mathrm{Surf}_i^F$, where F is the number of articulatory feature streams. In both cases, $\mathrm{Context}_i$ is shorthand for all of the context variable values at time i. Table 1 shows perplexity values on the test set for the four sets of context variables.[3] Adding the previous distinct phone/feature surface value to the context substantially improves the perplexity scores for both feature-based and phone-based models, but it is not clear if the distance in frames to the previous/next distinct values is beneficial. Thus, in the experiments in the next section, we only build context-dependent AF models using the *CD-Basic* and *CD-Basic + prevSurf* trees.

3.3. Lexical access experiments using AF decision trees

For the lexical access experiments, we use the same experimental setup as in [5]. We use a subset of the STP data, post-processed and divided into training (2942-word), development (165-word), and test (236-word) sets. The training set corresponds to the entire set used for the perplexity experiments above. The words are excised from continuous utterances and are treated as isolated for our purposes. The vocabulary contains about 3300 words.

The question being addressed is the following: Given the surface realization of a word, how accurately can we recognize the word?[4] For each word in the vocabulary, we compute $P(\mathrm{Word}|\mathrm{Surf}_{1:T}^{1:F})$ where Word is T frames long, and choose the word that maximizes this probability. For the AF-based models, these probabilities are computed via inference on the DBNs. For phone-based models, they are computed as the probability of the best alignment between each vocabulary word and the given surface phone sequence, with the phoneme-to-phone mapping given by the phonetic decision trees. We train the decision trees (and, later, other model parameters) on the training sets, tune on the development set, and do final testing on the test set.

Set of context variables	Phone-based	Feature-based
CI	3.51	2.56
CD-Basic	2.57	2.17
CD-Basic + prevSurf	1.72	1.80
CD-Basic + prevSurf + Dist	1.76	1.68

Table 1. Test set perplexities for four sets of context variables.

We compute the error rate (ER), the percentage of incorrectly classified words, for several of the systems described earlier. Results are given in Tables 2 and 3. *CI* is the CI AF-based model of [5].[5] *CD-Phone* is the context-dependent phone baseline model, which is very similar to the model of [4] as described in Section 2.1.

For the basic CD AF model *CD-BasicFeat*, the feature asynchrony and feature transition probabilities were set by hand based on linguistic judgments (for instance, probability of asynchrony decreases for increasing values of asynchrony between the feature streams). We also learned these probability values using the Expectation Maximization (EM) algorithm [14] given the observed surface feature values and word observations (*CD-BasicFeat+afterEM*). It took about 8 iterations to converge with a 0.2% difference in the total log probability on the training set. *CD-BasicFeat+prevSurf* also includes the previous distinct surface value as a context variable (as in Fig. 2).

Since these pronunciation models are intended to ultimately be used in complete speech recognizers, the goal is not only to identify the single correct word but also to ensure that the correct word is close to the top of the list of hypotheses when it is not correctly identified. We therefore also look at the "oracle" error rates of the 3-best and 5-best lists, i.e. the fraction of examples where the correct word is not in the 3- or 5-best list.

From Tables 2 and 3, we see that the *CD-BasicFeat* system performs better than the *CD-Phone* system on both the development set and the test set. The difference between the systems is statistically significant at $p < 0.01$ according to McNemar's test for both sets. This is an encouraging result in that the basic context dependent AF system is performing significantly better than a context-based phone system that is using more context surrounding each phoneme. We also note that the *CD-BasicFeat+afterEM* system is not very different

[3] Our perplexity values differ from the numbers reported in [8] as we reimplemented the decision tree construction using the WEKA toolkit [13] for a significant speedup in running time. However, we observe the same trends in perplexities as reported in [8].

[4] We are not using ground truth surface feature values, but rather the manual fine phonetic transcriptions converted via a deterministic mapping from phones to features. Performance may be improved with actual feature input, but the setup here shows that our models can be used, in principle, with an otherwise phone-based recognizer.

[5] Note that the numbers are different from those in [5] because homophones (whether/weather) were penalized in that work.

Model	ER	3-B ER	5-B ER
CI	26.1	21.2	21.2
CD-Phone	23.6	18.2	15.7
CD-BasicFeat	20.6	12.7	11.5
CD-BasicFeat+afterEM	20.0	12.1	11.5
CD-BasicFeat+prevSurf	17.6	10.9	9.1

Table 2. Error rates on the development set.

Model	ER	3-B ER	5-B ER
CI	40.5	35.9	33.7
CD-Phone	32.1	23.2	19.0
CD-BasicFeat	31.2	21.9	18.6
CD-BasicFeat+afterEM	30.8	21.5	18.1
CD-BasicFeat+prevSurf	29.1	18.6	16.0

Table 3. Error rates on the test set.

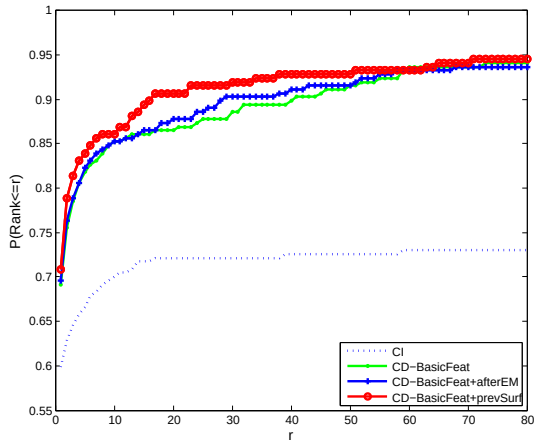

Fig. 3. Cumulative distribution of ranks of the correct word on the test set.

from *CD-BasicFeat*, implying that training the parameters of a system with a linguistically motivated initialization of these parameters does not significantly affect the error rates.

For a more detailed look at the effect of the various AF-based context variables, we also compute the cumulative distributions of the correct word's rank, shown in Fig. 3. The context-dependent models not only outperform the context-independent one in terms of accuracy, but also outperform it by increasing margins for increasing rank threshold r; and the best-performing model concentrates about 90% of the correct words within the top $r = 15$ hypotheses.

4. CONCLUSIONS AND FUTURE WORK

On a lexical access task, our context-dependent articulatory feature-based models perform significantly better than a context-dependent phone baseline. We also observe that with the inclusion of context, the best feature-based model almost always ranks the correct word within the top 15 or so hypotheses. Future work includes incorporating this model in a complete end-to-end speech recognizer, as well as improving our current models by adding more cross-word context, stress and other prosody-related context, and context-dependent asynchrony.

5. REFERENCES

[1] D. Jurafsky, W. Ward, Z. Banping, K. Herold, Y. Xiuyang, and Z. Sen, "What kind of pronunciation variation is hard for triphones to model?," in *Proc. ICASSP*, 2001.

[2] D. McAllaster, L. Gillick, F. Scattone, and M. Newman, "Fabricating conversational speech data with acoustic models: A program to examine model-data mismatch," in *Proc. ICSLP*, 1998.

[3] M. Saraçlar and S. Khudanpur, "Pronunciation change in conversational speech and its implications for automatic speech recognition," *Computer Speech and Language*, vol. 18, no. 4, pp. 375–395, 2004.

[4] M. Riley, W. Byrne, M. Finke, S. Khudanpur, A. Ljolje, J. McDonough, H. Nock, M. Saraçlar, C. Wooters, and G. Zavaliagkos, "Stochastic pronunciation modelling from hand-labelled phonetic corpora," *Speech Communication*, vol. 29, no. 2-4, pp. 209–224, 1999.

[5] K. Livescu and J. Glass, "Feature-based pronunciation modeling with trainable asynchrony probabilities," in *Proc. ICSLP*, 2004.

[6] L. Deng and D.X. Sun, "A statistical approach to automatic speech recognition using the atomic speech units constructed from overlapping articulatory features," *The Journal of the Acoustical Society of America*, vol. 95, no. 5, pp. 2702–2719, 1994.

[7] C.P. Browman and L. Goldstein, "Articulatory phonology: An overview," *Phonetica*, vol. 49, no. 3-4, pp. 155–180, 1992.

[8] S. Bowman and K. Livescu, "Modeling pronunciation variation with context-dependent articulatory feature decision trees," in *Proc. Interspeech*, 2010.

[9] S. Greenberg, J. Hollenback, and D. Ellis, "Insights into spoken language gleaned from phonetic transcription of the Switchboard corpus," in *Proc. ICSLP*, 1996.

[10] "OpenFST Library," http://www.openfst.org/.

[11] J. Bilmes and G. Zweig, "The Graphical Models Toolkit: An open source software system for speech and time-series processing," in *Proc. ICASSP*, 2002.

[12] "The Graphical Models Toolkit (GMTK)," http://ssli.ee.washington.edu/~bilmes/gmtk/.

[13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[14] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.