社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

Asynchronous Articulatory Feature Recognition Using Dynamic Bayesian Networks

Mirjam WESTER†, Joe FRANKEL†, and Simon KING†

† Centre for Speech Technology Research, University of Edinburgh, United Kingdom.

Abstract This paper builds on previous work where dynamic Bayesian networks (DBN) were proposed as a model for articulatory feature recognition. Using DBNs makes it possible to model the dependencies between features, an addition to previous approaches which was found to improve feature recognition performance. The DBN results were promising, giving close to the accuracy of artificial neural nets (ANNs). However, the system was trained on canonical labels, leading to an overly strong set of constraints on feature co-occurrence. In this study, we describe an embedded training scheme which learns a set of data-driven asynchronous feature changes where supported in the data. Using a subset of the OGI Numbers corpus, we describe articulatory feature recognition experiments using both canonically-trained and asynchronous-feature DBNs. Performance using DBNs is found to exceed that of ANNs trained on an identical task, giving a higher recognition accuracy. Furthermore, inter-feature dependencies result in a more structured model, giving rise to fewer feature combinations in the recognition output. In addition to an empirical evaluation of this modeling approach, we give a qualitative analysis, investigating the asynchrony found through our data-driven method and interpreting it using linguistic knowledge.

Key words Articulatory feature recognition, dynamic Bayesian networks

1. Introduction

The majority of ASR systems describe the parameterized speech signal in terms of phones: words are simply concatenations of phone sequences. Modeling a word as a sequence of phone segments, i.e. the "beads-on-a-string" paradigm [1], ignores the *source* of the variation present in spontaneous, conversational speech, describing the resulting modifications using context-dependent models. The variation in natural speech arises from the overlapping, asynchronous nature of speech production, along with effects such as co-articulation and assimilation. Given that these are articulatory phenomena, we believe that the variation encountered by an ASR system can be modeled in a principled manner using articulatory features (AF) as a representational basis.

Previous work, reported in [2], proposed dynamic Bayesian networks (DBN) as a model for articulatory feature recognition. For related work on feature models and DBNs see [3], [4]. The motivations for our approach are two-fold: firstly, dependencies between features can be modeled, and secondly, DBNs offer a framework in which the various components of a feature-based recognizer can readily be combined. Adding dependencies between the AFs was shown to improve feature recognition performance. The DBN results were promising, giving close to the accuracy of artificial neural nets (ANNs). However, the system was trained on canonical labels, leading to an overly strong set of constraints on feature co-occurrence. In this study, we describe an embedded training scheme with the goal of learning a set of data-driven asynchronous feature changes.

2. Data

Experimental work uses a subset of the Numbers corpus [5], a collection of naturally spoken numbers collected at the Center for Spoken Language Understanding (CSLU) at OGI. The utterances were taken from other CSLU telephone speech data collections, and include isolated digit strings, continuous digit strings, and ordinal/cardinal numbers. Each file in the Numbers corpus has been orthographically and phonetically transcribed following the CSLU Labeling Conventions [6].

The subset used in this study was selected at IDIAP to contain only the 30 most frequent words and no utterances with truncated words [7]. To ensure acceptable experiment turnaround times, we further reduced the amount of data by using only the first half of the training set, and splitting the validation set into two parts. The first was used for intermediate evaluation during training, and the second as an independent test set. Table 1 shows the number of utterances, phones and minutes of speech contained in each of the data sets. In all experiments, the acoustic waveforms are parameterized as 12 MFCCs and energy with 1^{st} and 2^{nd}

derivatives appended.

set	utterances	phones	minutes
train	5000	94,578	145
validation	1750	33,439	51
test	1768	39,258	75

Table 1 *Statistics of the OGI Numbers data selection.*

Frame-level feature labels were obtained in much the same way as in previous work [8], by mapping from phones to articulatory-acoustic features. The feature specifications are similar to those used in [8], with differences in the place of articulation and front-back groups. There are a number of phones which do not occur in spoken numbers, and therefore appear very rarely in the OGI Numbers corpus (e.g. /b/, /m/, and /h/). As a consequence, some places of articulation occur very infrequently in the data. To avoid problems of data sparsity, labial frames were mapped to labiodental and all glottal frames relabeled velar. The front-back feature group has been augmented to include a central value. The feature groups, their values and cardinalities are listed in Table 2.

feature	values	cardinality
manner	approximant, fricative, nasal, stop, vowel, silence	6
place	labiodental, dental, alveolar, velar, high, mid, low, silence	8
voicing	voiced, voiceless, silence	3
rounding	rounded, unrounded, nil, silence	4
front-back	front, central, back, nil, silence	5
static	static, dynamic, silence	3

Table 2 *Specification of the multi-leveled articulatory features used in this work. The right-hand column gives the cardinality of each feature.*

3. Dynamic Bayesian Networks

A Bayesian network (BN) provides a means of encoding the dependencies between a set of random variables (RV). The RVs and dependencies are represented as the nodes and edges of a directed acyclic graph. A Bayesian network exploits missing edges (implying conditional independence) to factor the joint distribution of all RVs into a set of simpler probability distributions. A dynamic Bayesian network (DBN) consists of instances of a Bayesian network repeated over time, with dependencies across time.

3.1 AF recognition model topology

Previous work derived a set of inter-feature dependencies for the task of articulatory feature recognition [8]. The same model topology is used in this work, and is shown in Figure 1. The Graphical Models Toolkit (GMTK) [9] was used to implement all models. As before, the observation process is a product of Gaussian mixture models (GMM), such that for f_k denoting the value of feature F_k, the probability of an observation $\mathbf{y}$ is given as the product of the probabilities of $\mathbf{y}$ given the individual features:

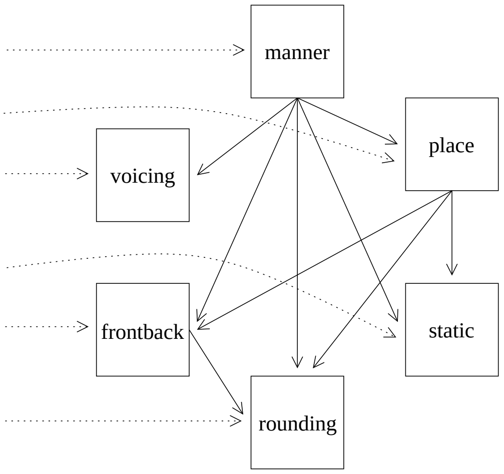

Figure 1 *Graph depicting the dependencies between features. Each feature is also conditioned on its value in the previous frame (implied by the dotted arrows) and a silence/non-silence node which, along with the observation process, has been omitted for clarity.*

$$p(\mathbf{y}|f_1, \ldots, f_6) = \prod_{k=1}^{6} p(\mathbf{y}|f_k) \qquad (1)$$

The sparse structure of the conditional probability tables (CPT) which describe the dependencies between features dictates which feature values can co-occur. Training on canonically-derived labels leads to a strong set of constraints, in effect re-encoding the phone labels to give a model resembling a monophone hidden Markov model (HMM). In the absence of labels which give the level of detail required to train a set of asynchronous feature labels, we chose to build an asynchronous model in a data-driven manner.

3.2 Asynchronous model training

Our goal is to derive a set of CPTs which allow asynchronous change where supported by the data, whilst retaining sufficient sparsity to limit the number of allowable feature combinations and give a workable model. The essence of the training scheme is as follows: zero values in CPTs trained on canonical labels are raised to some small value, and embedded training follows to allow feature combinations with strong acoustic likelihood to accumulate probability mass.

These give rise to non-zero entries in the CPTs, whilst combinations with low acoustic likelihood continue to result in zero or very low probabilities.

Simultaneous training of all 6 features in this manner would be computationally infeasible, and so asynchronous CPTs are trained for each feature in turn, with parent node CPTs trained before those of their children. The full scheme proceeds as follows:

1. Observation process and feature CPTs initialized by training on canonical feature labels and acoustic parameters.
2. For each feature F_k, such that all parents of F_k have already had asynchronous CPTs estimated:
   a. all zero probability values in the CPT replaced with $1/(\alpha\,\mathrm{card}(F_k))$.
   b. embedded training with feature sequence, but not timing, enforced. CPT for F_k trained until convergence with no other parameters updated.
   c. CPT cells containing values less than $1/(\alpha\,\mathrm{card}(F_k))$ set to zero to restrict the size of state space.
3. Embedded training until convergence of all feature CPTs and observation GMMs together.

The value of α was set to be 10^{-5}, an order of magnitude lower than the smallest CPT cell found after training on canonical labels.

The asynchronous-feature models derived from the intermediate parameters were used to realign the training set by decoding whilst enforcing the correct (according to the canonical transcription) sequence of features. In the following section, an analysis of the feature realignment is given.

4. Analysis of feature realignment

This section investigates the changes in feature boundaries by comparing the new asynchronous transcriptions to the canonical transcriptions. The goal is to ascertain how many changes occur, where they occur, and whether the changes are linguistically plausible, or simply a side effect of the model's preference for a slightly different labeling, or possibly due to errors in the canonically derived feature labels.

4.1 Overall boundary shifts

Table 3 shows the percentage of frames that are different in the canonical and asynchronous feature transcriptions. These results indicate sufficient movement is taking place to warrant further investigation.

Figure 2 shows the percentage of switches in the asynchronous feature transcriptions that have been placed $1-5$ frames left or right of the canonical boundary. For example, for voicing 36% of the feature switches occur at the same frame in the canonical and realigned data. 19% of voicing feature switches take place one frame before the canonical boundary, and 20% occur one frame later. About $70-75\%$ of boundaries are either the same or differ by only one frame in the two labelings. The number of switches within each feature group are given in Table 4.

feature	frames changed
manner	10.41%
place	5.93%
voicing	5.35%
rounding	5.36%
front-back	5.65%
static	5.36%

Table 3 *Percentage frames changed per feature group in training set. Total number of frames is 911,003.*

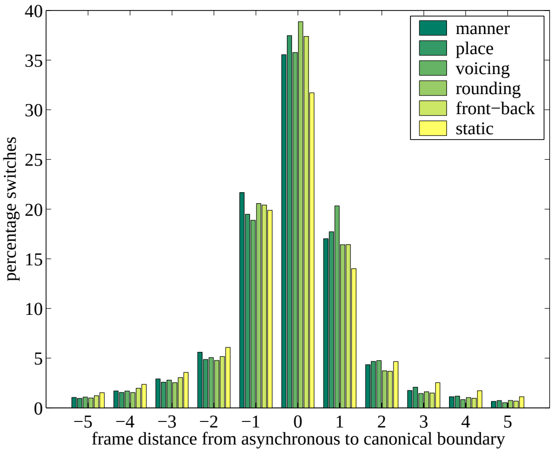

Figure 2 *Percentage switches that move N frame distance from the canonical boundary in the asynchronous transcriptions. Results given here are limited to five frames either side of the boundary.*

Table 4 gives the mean overall frame deviation from the canonical boundary for each feature group. The results in the "overall mean" column seems to indicate that there is a half frame bias in our canonical feature labeling. To check whether this is indeed the case, the canonical phone-derived labeling was generated anew after first subtracting 5 ms from all time stamps. The resulting half frame corrected labeling is compared to the asynchronous feature labeling. The third column in Table 4 shows that applying the half frame shift indeed corrects for the bias in our canonical labeling.

A side effect of subtracting 5 ms is that slight changes in the feature sequences occur. A number of features (and therefore switches) is deleted or inserted because they last only half a frame. Utterances in which the canonical feature switch sequence changed after the 5 ms correction was applied were omitted from the comparisons. Table 4 shows the original number of feature switches and the number of

feature	overall mean	-1/2 frame mean	original switches	-1/2 frame switches
manner	-0.4806	0.0560	84,292	65,383
place	-0.3787	0.1337	83,992	65,516
voicing	-0.4256	0.1914	51,909	45,658
rounding	-0.4276	0.1189	78,804	63,685
front-back	-0.5446	-0.0316	78,873	63,811
static	-0.4805	0.0654	58,543	49,023
overall	-0.4564	0.0847	436,413	353,076

Table 4 *Overall mean deviation from canonical boundaries and from half frame adjusted canonical boundaries. Number of feature switches for each feature group present in the training data.*

feature switches after removing the utterances that do not match the original data.

In future work, we will apply the half frame shift to the OGI Numbers' time stamps prior to generating the frame-level feature labels. In the remainder of this analysis however, the original canonical labeling is used. It would not be a fair comparison if we changed the canonical labeling but did not retrain the the asynchronous model CPTs to reflect this. In addition, the number of switches lost due to omitting utterances for which the feature sequences do not match the canonical feature sequences is quite substantial (see differences between columns 4 and 5 in Table 4) and we did not want to base the current analysis on a reduced data set.

4. 2 Specific boundary shifts

Overall deviation from feature boundaries only gives a general indication of whether the system as a whole is behaving as expected. To find out whether linguistically plausible processes are being captured, we need to look at individual feature switches. Therefore, Table 5 gives details of specific feature switches. Only feature switches for which the mean deviates more than 2 frames from the canonical boundary are given.

feature	mean	feature switch	count
manner	2.36	approximant → silence	1319
place	-2.51	silence → labiodental	1920
	-11.40	silence → high	290
	3.74	dental → mid	204
	-2.59	high → mid	502
	-2.79	high → velar	489
voicing	-2.25	silence → voiced	5718
rounding	-2.48	silence → unrounded	1344
	-4.31	unrounded → rounded	134
front-back	-2.58	silence→ front	1317
	6.98	central →silence	402
	-3.58	front→ back	130
	-2.93	front→ central	56

Table 5 *Mean deviation from canonical boundary for specific feature switches >2 or <-2 frames.*

Figure 3 illustrates two of the place of articulation feature switches in context "dental → mid" and "silence → high". Note that in our feature mapping, closures are treated as silence.

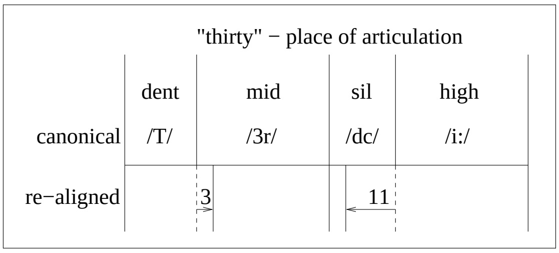

Figure 3 *Example of frame shifts for place of articulation.*

Another example from Table 5 which is interesting to highlight is the feature switch from front to back. The mean deviation for this switch is -3.6, it occurs 130 times in the data. In 87% of the cases the context for this switch is /i:/ to /oU/ which occurs for example in the word sequence "three oh". /i:/ is a front vowel, and /oU/ is a diphthong which has been canonically labeled as back. The data however indicate that more often than not the start of /oU/ is more front than back. One could definitely argue that labeling /oU/ as back is incorrect to start with. Furthermore, the data corroborate this and point us in the direction of labeling the diphthongs as diphthongs. Thus in future work, the canonical labeling for the diphthong /oU/ will consist of a $50 - 50$ split from central to back. In addition, all other diphthongs will be relabeled in the light of this finding.

4. 3 Linguistically expected shifts

As we restricted our asynchronous-feature models by enforcing the canonical sequence of features and did not allow for deletions, insertions and/or substitutions of features there are only a few linguistic processes we can investigate within this analysis.

One of the linguistically motivated feature boundary shifts we can examine is vowel nasalization. This can occur when a vowel is followed by a nasal consonant for example in the words "nine", "one", "and". The expectation is that the boundary will move to the left, i.e. the vowel becoming nasalized. From the data we find that the overall mean deviation for the feature switch "vowel → nasal" is -0.61, indicating a slight movement of the nasal feature into the vowel feature. Thus, the data indicate there is some nasal spread into preceding vowels but it is slight. In addition, these data may not be giving the full picture as nasal spread is already partially included in our canonical labeling through the nasalization diacritic.

Even though linguistically expected shifts are not obvious in this data for the above mentioned reasons, the analysis highlights a number of issues related to the canonical labeling which can be resolved, and will possibly lead to better initialization of our DBNs and consequently improved asynchronous models.

One final point to address in this analysis is to what degree we can speak of *asynchronous* feature shifts. In the approach described here the features are not independent of each other but adhere to the conditional dependencies depicted in Figure 1. Consequently, the question arises whether the boundaries for the various feature groups move synchronously or asynchronously. An estimate of the degree of asynchrony can be given by the number of feature combinations. The realigned data contains 351 combinations which, compared to the 62 feature combinations in the canonical data, shows that there is indeed asynchronous feature boundary movement. Future work will further investigate the relationships between the various feature groups.

5. Articulatory feature recognition experiments

5. 1 Presentation of results

One problem with the task of articulatory feature recognition is how to evaluate performance. Previous work compared AF recognition results with % frames correct averaged over all features, and % frames in which all features are correct together. These measures compare recognition output with canonically-derived labels and so have the drawback of penalizing asynchrony. However, we do present results computed in this way, along with the standard word recognition measure %accuracy:

$$100 \times (n(\text{correct}) - n(\text{insertions}))\,/n(\text{total labels}) \qquad (2)$$

calculated using the HTK tool HResults [10]. The %accuracy measure disregards timing and allows asynchronous feature change, but still has the capacity to penalize some of the events we would wish to capture, such as where assimilation leads to the deletion of a feature change.

5. 2 ANN feature recognition

A separate ANN mapping from acoustics to feature value was trained for each feature group using the NICO Toolkit [11]. Further details can be found in [8]. The phone to feature mapping used to generate the canonical labels included the diacritics listed in Section 2.. Recognition was implemented using a hybrid ANN/HMM approach. The feature posterior probabilities obtained using NICO [11] were used as input to NOWAY [12], a start-synchronous decoder designed for use with hybrid ANN/HMM systems. A feature insertion penalty was included during decoding to control the number of insertions and deletions. Each of the feature groups was decoded in isolation, and all features weighted equally when compiling overall results. The results of AF recognition using ANNs are given in table 8 in the following section. Comparing against canonical labels, an average of 85.1% frames were correctly identified across the 6 features, with all features correct together in 65.5% of frames. The overall recognition accuracy was found to be 78.9%, with 3473 distinct feature combinations found in the decoded output.

5. 3 DBN feature recognition

Training on canonical labels and MFCCs involves a regime of splitting and vanishing Gaussian mixture components. The final model set gave results of 83.8% frames correct across all features and 76.5% frames all correct together, with an observation process comprising 89742 Gaussian components. Results of 83.0% average and 74.7% frames correct together were found on an intermediate model set using just over a tenth of the parameters. These results along with the numbers of Gaussian components are shown in table 6. For efficiency, the intermediate model parameters were used as a basis in training the asynchronous CPTs.

model	average correct	all correct together	# Gaussian components
intermediate	83.0%	74.7%	9519
final	83.8%	76.5%	89742

Table 6 *Close to highest AF recognition results are given by an intermediate model set using substantially fewer Gaussians. Validation data set results, trained on canonical labels.*

Initial feature recognition accuracy results revealed numerous insertion errors. A transition penalty was therefore included to balance insertions and deletions, and its value set on held-out validation data. Table 7 shows that AF recognition accuracy decreases using the new asynchronous CPTs where evaluation is based on canonically-derived labels. However, with recognition accuracy used to make comparisons, asynchronous CPTs lead to improved performance, with the largest difference found prior to the final all-parameter embedded training step. Recognition with the asynchronous models results in larger numbers of feature combinations occurring in the output, 288 compared to 79 after the final embedded training step.

Table 8 gives AF recognition results for a system where the final model observation GMMs are combined with the asynchronous feature CPTs developed using intermediate parameters, and then all-parameter embedded training performed until convergence. Also shown are the ANN results of section 5. 2. Comparison based on canonical labels shows that the ANNs give a slightly higher average frame-wise accuracy, 85.1% compared to 84.7%, though a substantially lower percentage of frames in which all features are correct together.

model	average correct	all correct together	recognition accuracy	# combinations
intermediate GMM parameters, only CPT embedded training				
canonical	84.5%	77.0%	79.2%	67
asynch	84.4%	75.9%	80.2%	193
intermediate GMM parameters, full embedded training				
canonical	84.7%	77.4%	80.6%	79
asynch	83.6%	73.0%	80.8%	288

Table 7 *AF recognition with and without asynchronous feature changes, before and after final all-parameter embedded training, validation data set.*

The DBNs also give a higher recognition accuracy, 81.2% compared to 78.9%.

model	average correct	all correct together	recognition accuracy	# combinations
ANN	85.1%	65.5%	78.9%	3473
DBN	84.7%	77.2%	81.2%	186

Table 8 *DBN and ANN AF recognition compared on test set data. DBN system is built on final model parameters, asynchronous CPTs and all-parameter training.*

6. Discussion and future

Previous work on AF recognition has for the most part relied upon canonical transcripts during training. With features derived from phone labels, the resulting models inevitably carry exactly the limitations which we wish to circumvent using articulatory features as a representational basis.

In this study, we have attempted to move away from our dependence on canonical labels by implementing an embedded training scheme which allows asynchronous feature changes where supported in the data. We believe the results to be encouraging: embedding training did not lead to degeneration of the models, in fact giving a slight increase in feature recognition accuracy. Furthermore, the asynchronous DBNs outperformed ANNs using the frames correct together and recognition accuracy measures despite an overly simple observation process. The increased numbers of feature combinations found in the asynchronous model output show that the constraints due to training on canonical labels have been relaxed, though the numbers remain an order of magnitude lower than those found in ANN feature recognizer output. The structure evident from the DBN output is desirable so long as the model remains capable of producing feature recognition which is sufficiently detailed that the limitations of a phone-based representation are avoided.

Analysis of the asynchronous feature changes proved to be illuminating. First of all, the fact that the data can be analyzed in such a manner is an added benefit of the articulatory feature representation. Secondly, it has lead us to revise the labeling of closures and diphthongs in future work, as well as disclosing a half frame bias which was present in our original phone-derived feature labeling.

Future DBN articulatory feature recognition work will include modifying the training process to allow feature insertions, deletions and substitutions. We are currently in the process of implementing an improved observation process which uses combination-specific distributions where possible, and backs off to product models where training data is limited. We also intend to analyze the recognition output in order to compare the asynchrony found through our data-driven approach to the asynchrony which may be expected on the basis of linguistic knowledge.

However, for meaningful evaluation of refinements, the feature recognizer must be incorporated into a word recognizer as this is the domain in which it will ultimately be used.

7. Acknowledgments

Many thanks to Jeff Bilmes for prompt and helpful responses to questions regarding GMTK.

References

[1] M. Ostendorf, “Moving beyond the ‘beads-on-a-string’ model of speech,” in *Proc. of IEEE ASRU Workshop*, Keystone, CO., 1999, pp. 79–84.

[2] J. Frankel, M. Wester, and S. King, “Articulatory feature recognition using dynamic Bayesian networks,” in *Proc of ICSLP-’04*, Jeju, Korea, 2004.

[3] K. Livescu and J. Glass, “Feature-based pronunciation modeling for speech recognition,” in *Proc. of HLT/NAACL*, Boston, 2004.

[4] K. Livescu and J. Glass, “Feature-based pronunciation modeling with trainable asynchrony probabilities,” in *Proc. of ICSLP ’04*, Jeju, Korea, 2004.

[5] CSLU @ OGI, “Numbers v1.3,” Website, 23 August 2002, `http://www.cslu.ogi.edu/corpora/numbers/index.html`.

[6] T. Lander, “The CSLU labeling guide,” Website, 15 May 1997, `http://www.cslu.ogi.edu/corpora/docs/labeling.pdf`.

[7] J. Mariéthoz and S. Bengio, “A new speech recognition baseline system for Numbers 95 Version 1.3 based on Torch,” Tech. Rep. IDIAP-RR 04-16, IDIAP, 2004.

[8] J. Frankel, M. Wester, and S. King, “Articulatory feature recognition using dynamic Bayesian networks,” in *Proc of ICSLP-’04*, Jeju, Korea, 2004.

[9] J. Bilmes and G. Zweig, “The graphical models toolkit: An open source software system for speech and time-series processing,” in *Proc. of ICASSP ’02*, Orlando, 2002.

[10] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.1)*, Cambridge University Engineering Department, 2001.

[11] N. Ström, “Phoneme probability estimation with dynamic sparsely connected artificial neural networks,” *The Free Speech Journal*, vol. Issue #5, 1997.

[12] S. Renals and M. Hochberg, “Decoder technology for connectionist large vocabulary speech recognition,” 1995, Memo CS-95-17, Dept. of Computer Science, University of Sheffield.