PHONE CLUSTERING USING THE BHATTACHARYYA DISTANCE

Brian Mak   *Etienne Barnard*

Center for Spoken Language Understanding
Oregon Graduate Institute of Science and Technology
20000 N.W. Walker Road, Portland, OR 97006
{mak, barnard}@cse.ogi.edu

ABSTRACT

In this paper we study using the classification-based Bhattacharyya distance measure to guide biphone clustering. The Bhattacharyya distance is a theoretical distance measure between two Gaussian distributions which is equivalent to an upper bound on the optimal Bayesian classification error probability. It also has the desirable properties of being computationally simple and extensible to more Gaussian mixtures. Using the Bhattacharyya distance measure in a data-driven approach together with a novel 2-Level Agglomerative Hierarchical Biphone Clustering algorithm, generalized left/right biphones (BGBs) are derived. A neural-net based phone recognizer trained on the BGBs is found to have better frame-level phone recognition than one trained on generalized biphones (BCGBs) derived from a set of commonly-used broad categories. We further evaluate the new BGBs on an isolated-word recognition task of perplexity 40 and obtain a 16.2% error reduction over the broad-category generalized biphones (BCGBs) and a 41.8% error reduction over the monophones.

1. INTRODUCTION

Perhaps the most significant advance in speech recognition during the past decade has been the incorporation of context-dependent phonetic units into frame-based systems. Currently, the most popular choices of subword units are the generalized biphones (GB) or generalized triphones (GT).

There are two basic approaches to derive these generalized phones: (1) The knowledge-driven approach employs linguistic knowledge about the coarticulatory influences between neighboring phones. For example, L. Deng *et al.* [3] defined contexts based on broad phonetic categories and classified the articulatory effects on vowels and consonants each to 5 types and derived 25 generalized contexts for each phone. A. Ljolje [8] used more detailed contextual effects to derive a set of 19 left-context classes and 18 right-context classes. (2) The data-driven approach evaluates all contexts in the training data, and uses some distance measure with a clustering algorithm to split or merge the contexts to a specified number of generalized contexts. This usually uses an information-theoretic distance measure commonly employed with Hidden Markov models. Examples are works from D'Orta *et al.* [4], Juang and Rabiner [10], and Lee [6]. Lee *et al.* [7] later suggested another tree-based allophone clustering method. All allophones are placed in the root of the decision tree and each node of the tree is associated with a binary question, which is selected from a set derived by a linguistic expert. The "best" question is assigned to a node if it results in a binary split with minimal loss of entropy. Though some of the linguistically-motivated phone units derived using the first approach work quite well, the second approach is more popular because (a) it is difficult and tedious for even a linguistic expert to come up with broad phonetic classes in the case of biphones and triphones; and, (b) more importantly, most of contemporary frame-based recognizers are acoustically motivated; the second method fits well into their working paradigm by making full use of the acoustic information from the data. It is the data-driven approach we adopt in this paper.

As we have been working on speech recognition using the neural-network approach in which no HMMs are built, it is natural for us to explore distance measures other than the information-theoretic ones to do phone (context) clustering. Speech recognition is, after all, a classification problem; thus we study using the classification-based Bhattacharyya distance measure to guide phone clustering. The Bhattacharyya distance is a theoretical distance measure between two Gaussian distributions which is equivalent to an upper bound on the optimal Bayesian classification error probability. It also has the desirable properties of being computationally simple while at the same time being extensible to more Gaussian mixtures.

2. BHATTACHARYYA DISTANCE

The Bhattacharyya distance is covered in many texts on statistical pattern recognition (for example, [5]). We will give a brief review here with the following notations:

ω_i	: class $i, i = 1, 2$
P_i	: a priori probability of ω_i
M_i	: mean vector of class ω_i
Σ_i	: covariance matrix of class ω_i

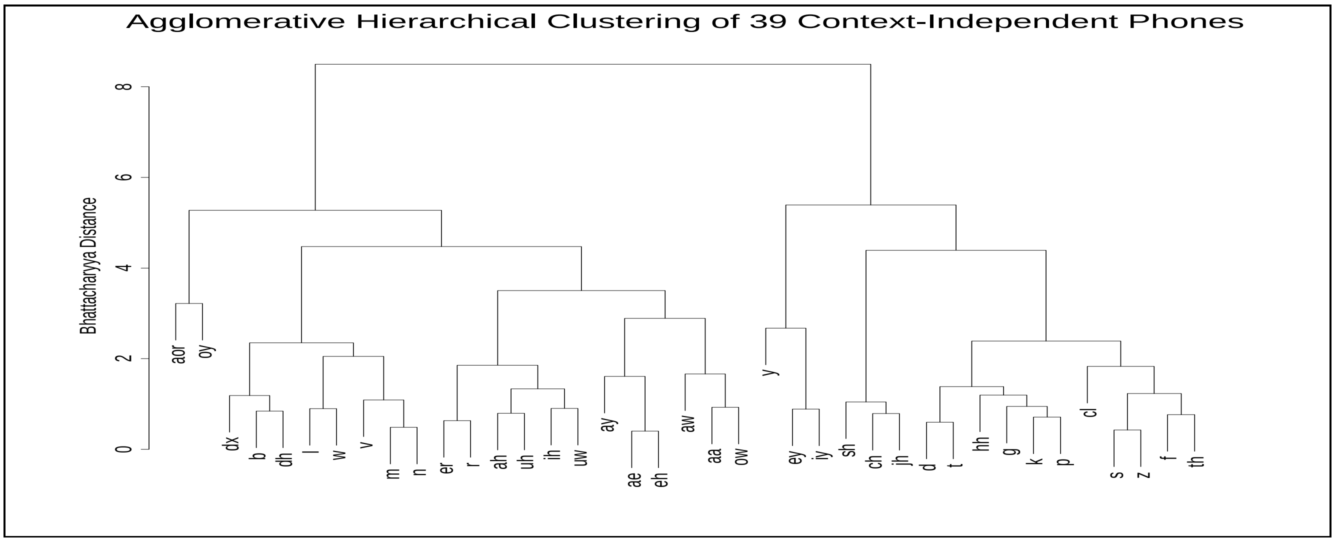

Figure 1: Agglomerative hierarchical clustering of 39 context-independent monophones

The Bhattacharyya distance, D_{bhat}, is a separability measure between two Gaussian distributions and is defined as follows:

$$D_{bhat} = \frac{1}{8}(M_2 - M_1)^T \left[\frac{\Sigma_1 + \Sigma_2}{2}\right]^{-1} (M_2 - M_1) + \frac{1}{2} \ln \frac{\left|\frac{\Sigma_1+\Sigma_2}{2}\right|}{\sqrt{|\Sigma_1||\Sigma_2|}} \qquad (1)$$

The first term of eqn.(1) gives the class separability due to the difference between class means, while the second term gives the class separability due to the difference between class covariance matrices. Furthermore, the optimal Bayes classification error between the two classes is bounded by the following expression:

$$\varepsilon \ \leq \sqrt{P_1 P_2} \ \exp\left(-D_{bhat}\right) \qquad (2)$$

We will refer to the upper bound of the error probability evaluated from the inequality(2) with $P_1 = P_2 = 0.5$, as the *Bhattacharyya error*, ε_{bhat} . That is,

$$\varepsilon_{bhat} \ = 0.5 \ \exp\left(-D_{bhat}\right) \qquad (3)$$

By setting the two prior probabilities equal, the two terms ε_{bhat} and D_{bhat} are equivalent in that both indicate the "intrinsic" separability of the two distributions, regardless of their prior probabilities. In summary, advantages of using the Bhattacharyya distance are that

- it is computationally very simple; and,
- by deriving it from an error bound rather than from an exact solution, it provides a "smoothed" distance between the two classes in study, which is more appropriate since we do not believe our data to be truly normally distributed.

3. BHATTACHARYYA DISTANCE BETWEEN PHONES

As a first step in our study of using the Bhattacharyya distance to measure phone separability, the following simple procedure is used to derive a Gaussian acoustic model for each phone:

Step 1. Each phone utterance is first divided into frames of equal duration.

Step 2. Speech features of dimension N are computed from each frame.

Step 3. Each phone utterance is also divided into three equal segments, each consisting of an equal number of frames.

Step 4. Each segment of the phone is then represented by the centroid of its constituent frame vectors.

Step 5. Finally the three centroid feature vectors are concatenated together to form a single $3N$-dimensional vector to represent a phone utterance.

Step 6. The mean and covariance matrix of the $3N$-dimensional utterance vectors from all utterances of a phone are computed.

Specifically in this paper, each utterance is pre-emphasized with a filter whose transfer function is $1 - 0.97z^{-1}$. Then for every $10ms$ frame, a $25ms$ Hamming window is applied, and the first twelve 24th-order LPC cepstral coefficients (LPCC) are extracted. Thus, each phone utterance is represented by a 36-dimensional feature vector.

Once the Gaussian phone models are in place, it is straightforward to compute the Bhattacharyya distance between any two phones using eqn.(1).

As an example, Fig. 1 shows a clustering tree of 39 context-independent monophones obtained by using a standard agglomerative hierarchical clustering (AHC) procedure and guided by a Bhattacharyya distance matrix computed over the OGI_TS corpus [9]. It can be seen that most clusters at the bottom level are well-known confusable pairs: {l, w}, {m, n}, {er, r}, {ae, eh}, {ch, jh}, {d, t}, {s, z}, and {f, th}.

4. TWO-LEVEL AGGLOMERATIVE HIERARCHICAL BIPHONE CLUSTERING

In theory, we may simply compute the Bhattacharyya distance matrix among biphones containing a particular basis phone, perform the standard agglomerative hierarchical clustering and select the appropriate clusters as our generalized biphones for that basis phone. However, in practice, we are faced with three problems in this data-driven approach:

1. insufficient data — reliable models cannot be estimated for some rare biphones
2. incomplete biphone coverage — some biphones never occur in the training data
3. unbalanced data — it seems reasonable to require similar amount of training data for the two entities during the computation of their Bhattacharyya distance

Here we propose a novel algorithm, the 2-Level Agglomerative Hierarchical Biphone Clustering (AHBC) algorithm as shown in Algorithm 1. The problems 1 and 2 are solved by augmenting the biphone Bhattacharyya distance matrix with the monophone Bhattacharyya distance matrix for those unseen and under-represented biphones. A crude AHC is performed to obtain the first-level generalized biphones. It stops as soon as each first-level generalized biphone has a fair amount of data in the training corpus. Acoustic models are then re-computed for these first-level generalized biphones, and another round of AHC is performed. The third problem of unbalanced data is solved by limiting the growth of cluster size using the thresholds, FREQ_THRESHOLD_1 and FREQ_THRESHOLD_2.

Algorithm 1 2-Level Agglomerative Hierarchical Biphone Clustering (AHBC) Algorithm

Step 1. Compute the Bhattacharyya distance matrix for monophones.

Step 2. Compute the Bhattacharyya distance matrix for all *possible* left(right) contexts of the basis phone; for those contexts with counts less than COUNT_THRESHOLD, back off with the monophones distances.

Step 3. Perform the first-level AHC with compact inter-cluster distances.

Step 4. Initialize pick_distance to INIT_IC_DIST_1.

Step 5. Pick all clusters from the clustering tree with merged distance less than pick_distance *and* with total data counts not exceeding FREQ_THRESHOLD_1.

Step 6. Determine coverage of the resulting "good" generalized biphones on the training data. (A generalized biphone is considered good if it has more than COUNT_THRESHOLD data count on the training data.)

Step 7. If coverage requirement is satisfied or pick_distance is the last available, go to Step 8. Otherwise, update pick_distance to the next greater distance in the clustering tree and go to Step 5.

Step 8. Fix each "bad" generalized biphone, which has too few data, by merging it with a good one based on the minimum distance between any two phones in the two clusters.

Step 9. Compute Gaussian models and then the Bhattacharyya distance matrix from the first-level generalized biphones.

Step 10. Perform the second-level AHC with compact inter-cluster distances.

Step 11. Pick all clusters from the clustering tree with merged distance less than INIT_IC_DIST_2 *and* with total data counts not exceeding FREQ_THRESHOLD_2.

5. RECOGNITION EXPERIMENT

To evaluate the new generalized biphones (BGBs) derived by the Two-Level AHBC algorithm, we train three separate general-purpose phonetic neural nets using stochastic Backpropagation with the OGI_TS corpus. Out of the 208 usable corpus files, 148 are used for training, 30 for cross validation, and the remaining 30 for testing. The ratio of male speakers to female speakers in each of the data sets is roughly 2:1. The neural nets are two-layer MLPs with 50 hidden nodes and 56 inputs which comprises seven 7th-order PLP coefficients plus the normalized energy for the current frame as well as its six neighboring frames. The number of output units depends on the type of phone units used as follows:

- net-MONO has 39 monophone output classes;
- net-BCGB has 575 output classes representing a set of linguistically-derived and commonly-used broad-category generalized biphones (BCGBs); and,
- net-BGB has 424 output classes representing the BGBs derived from the two-Level AHBC algorithm.

Table 1: Frame level phone recognition results on OGI_TS Corpus (Monophone and BGB are evaluated on the same speech samples; and, confusions among GBs of the same basis phone are ignored for BCGB and BGB)

PHONE UNITS	#CLASSES	#TRAINING FRAMES/CLASS	FRAME LEVEL % CORRECT
Monophone	39	5000	30.7
BCGB	424	500	26.6
BGB	575	500	32.8

Table 2: Isolated word recognition results on OGI Names Corpus (perplexity=40)

PHONE UNITS	NAMES (PERPLEXITY=40)	
	% CORRECT (STD. DEV.)	% ERROR REDUCTION BY BGBs
Monophone	75.8 (0.70)	42.1
BCGB	83.3 (0.57)	16.2
BGB	86.0 (0.55)	—

Table 1 gives the frame level phone recognition results of the three phonetic nets.

5.1. Isolated Word Recognition

The comparison at frame level is of limited value because different test data are used for the different phone units. It is more meaningful to test the three phone types on a real-world recognition task. Four thousand phonetically transcribed names are selected from the OGI Names Corpus [2] with balanced genders. One hundred test sets of perplexity 40 are constructed by randomly choosing twenty male speaking names and twenty female speaking names 100 times without replacement. The three phonetic nets (net-MONO, net-BCGB, and net-BGB) are used to recognize each test set of names using Viterbi search with a lexical tree [1]. Recognition results are averaged over the 100 test sets as shown in Table 2. The new generalized biphones (BGBs) obtain an error reduction of 42.1% over the monophones, and 16.2% over the broad-category generalized biphones (BCGBs).

6. DISCUSSION & FUTURE WORK

In this paper, we have demonstrated that the Bhattacharyya distance, which is derived from optimal Bayesian classification theory, is a useful distance measure for speech phones. We also propose a novel data-driven algorithm to derive generalized biphones using the Bhattacharyya distance matrix among the biphones. In an isolated-word recognition task, using the derived generalized biphones results in a 16.2% error reduction over a set of linguistically-motivated broad-category generalized biphones.

We have successfully extended the Bhattacharyya distance to Gaussian mixtures, and we are going to repeat the experiments described in this paper by modelling phone units as Gaussian mixtures; we expect to get better generalized biphones due to more accurate models.

7. ACKNOWLEDGMENT

This work was supported through a grant in the Young Investigator Program of the Office of Naval Research. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

8. REFERENCES

1. E. Barnard, R. A. Cole, M Fanty, and P. Vermeulen. "Real-World Speech Recognition With Neural Networks". In R. J. Alspector and T. X. Brown, editors, *Applications of Neural Networks to Telecommunications (IWANNT95)*, volume 2, pages 186–193. Lawrence Erlbaum Assoc., Hillsdale, New Jersey, 1995.
2. R. A. Cole, M. Noel, T. Lander, and T. Durham. "New Telephone Speech Corpora at CSLU". *Proceedings of Eurospeech*, pages 821–824, Sep 1995.
3. L. Deng, V. Gupta, M. Lennig, P. Kenny, and P. Mermelstein. "Acoustic Recognition Component of an 86,000-Word Speech Recognizer". *Proceedings of IEEE ICASSP*, pages 741–744, 1990.
4. P. D'Orta, M. Ferretti, and S. Scarci. "Phoneme Classification for Real Time Speech Recognition of Italian". *Proceedings of IEEE ICASSP*, pages 81–84, 1987.
5. K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, Inc., 2nd edition, 1990.
6. K. F. Lee. "Context-Dependent Phonetic Hidden Markov Models for Speaker-Independent Continuous Speech Recognition". *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(4):599–609, April 1990.
7. K. F. Lee, S. Hayamizu, H. W. Hon, C. Huang, J. Swartz, and R. Weide. "Allophone Clustering For Continuous Speech Recognition". *Proceedings of IEEE ICASSP*, pages 749–752, 1990.
8. A. Ljolje. "High Accuracy Phone Recognition Using Context Clustering and Quasi-Triphonic Models". *Computer Speech and Language*, 8:129–151, 1994.
9. Y.K. Muthusamy, R.A. Cole, and B. T. Oshika. "The OGI Multi-Language Telephone Speech Corpus". *Proceedings of ICSLP*, II:895–898, Oct 1992.
10. L. R. Rabiner, C. H. Lee, B. H. Juang, and J. G. Wilpon. "HMM Clustering for Connected Word Recognition". *Proceedings of IEEE ICASSP*, pages 405–408, 1989.