Abstract

Classifier combination has long been a staple for improving robustness of ASR systems; we present an experiment where introducing phonological feature scores from another lab’s system [1] into our system gives a statistically significant improvement in Conditional Random Field-based TIMIT phone recognition, despite a standalone system based on their features performing significantly worse. The second part of the paper explores the reasons for this improvement by examining different representations of phonological attribute classifiers, in terms of what they are classifying (binary versus n-ary features) and representation of scoring functions. The analysis leads to the conclusions that while binary phonological feature estimates usually are worse than n-ary features, the combination of the two can be quite good if there are also differences in the feature definitions or training paradigm.

Index Terms: Speech recognition, Feature Combination, Binary Features

1. Introduction

To improve accuracy and performance of systems, automatic speech recognition (ASR) systems often combine several feature representations. This is motivated by the assumption that some characteristics that are present in a particular feature representation are absent in another feature representation, and that the discrepancies between representations can help improve robustness. In general, different feature extraction algorithms can reveal complementary aspects of the original acoustic signal [2].

In the Automatic Speech Attribute Transcription (ASAT) project, a joint venture between Georgia Tech, Rutgers University, and Ohio State [3], a prevailing goal has been to develop platforms that allow for combination of different representations in a plug-and-play framework; the particular focus of the project has been integrating information from detectors of phonological events. In one thrust of this project, we have been experimenting with log-linear modeling of sequences (via Conditional Random Fields [4]) as combiners of different types of linguistic event estimators [5]. The CRF framework has been utilized previously within the project to combine systems from Rutgers, OSU, and GaTech [6].

To this point, we have not attempted to integrate systems external to the ASAT project, which is critical for determining the plug-and-playability of the framework. Ore and Slyh [1], from the Air Force Research Lab (AFRL), presented a system at Interspeech 2007 for improving phonological feature estimators for TIMIT by combining Gaussian Mixture Model (GMM) and Multi-Layer Perceptron (MLP) estimators. They have very

<table>
<thead>
<tr>
<th>Class</th>
<th>Feature Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SONORITY</td>
<td>Vowel, Obstruent, Sonorant, Syllabic, Silence</td>
</tr>
<tr>
<td>VOICE</td>
<td>Voiced, Unvoiced, N/A</td>
</tr>
<tr>
<td>MANNER</td>
<td>Plosive, Nasal, Flap, Fricative, Approximant, Lateral Approximant, Affricate</td>
</tr>
<tr>
<td>PLACE</td>
<td>Bilabial, Labiodental, Labiovelar, Dental, Alveolar, Postalveolar, Retroflex, Palatal, Velar, Uvular, Glottal</td>
</tr>
<tr>
<td>HEIGHT</td>
<td>Close, Open-close, Close-mid, Open-mid, Near-open, Near-close, Near-mid, Near-back, Near-front, Central, Back</td>
</tr>
<tr>
<td>ROUND</td>
<td>Round, Nonround, Round, Nonround, Round, Nonround</td>
</tr>
<tr>
<td>TENSE</td>
<td>Tense, Lax, N/A</td>
</tr>
</tbody>
</table>

Table 1: OSU definitions: 44 n-ary phonological features across 8 classes.

<table>
<thead>
<tr>
<th>Class</th>
<th>Feature Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>VOICE</td>
<td>Voiced, Unvoiced</td>
</tr>
<tr>
<td>MANNER</td>
<td>Plosive, Nasal, Flap, Fricative, Approximant, Lateral Approximant, Affricate</td>
</tr>
<tr>
<td>PLACE</td>
<td>Bilabial, Labiodental, Labiovelar, Dental, Alveolar, Postalveolar, Retroflex, Palatal, Velar, Uvular, Glottal</td>
</tr>
<tr>
<td>HEIGHT</td>
<td>Close, Open-close, Close-mid, Open-mid, Near-open, Open</td>
</tr>
<tr>
<td>FRONT</td>
<td>Front, Near-front, Central, Near-back, Back</td>
</tr>
<tr>
<td>ROUND</td>
<td>Rounded, Unrounded</td>
</tr>
</tbody>
</table>

Table 2: AFRL definitions: 36 binary phonological features across 6 classes.

graciously given us features to experiment with in combination with our system.

In the next section, we describe an initial experiment in combining the OSU and AFRL systems, in which we found somewhat unexpected results. The following section then describes some diagnostic experiments that we have run with our own system to try to tease apart how representations of phonological features may be interacting within the CRF system.

2. Combining AFRL and OSU systems

In previous work in our lab [5], we had carried out experiments where we had trained a baseline MLP system on a target phone set of 61 labels and utilized the posterior estimates $P(\text{phone}|x)$, where $x$ is the acoustic input (in this case, 13-dimensional PLP vectors with double-deltas), and utilized the estimates as stand-in acoustic features. We term these features as OSU/phones. Phonological feature posterior estimators were also instantiated
by mapping phones to feature classes. A set of eight n-ary neural networks were trained to classify individual phonological features as shown in Table 1. The phonological features used traditional phonetic categories such as manner, place, etc., and were based on the definition of the International Phonetic Association (IPA) phonetic chart. The posterior probabilities were taken at the output of each model’s MLP classifier, which we refer to as the OSU.phono.features.

The system by Ore and Slyh [1] utilizes a slightly different feature set (Table 2); they perform phonological feature detection using Gaussian Mixture Models (GMMs) and MLPs, where the scores from the GMM and MLP-based feature detector were combined using a second MLP for each feature treated in a binary fashion (36 MLPs in total). The first layer of the system was trained on the Wall Street Journal (WSJ1) data, whereas the second, combining MLP was trained on TIMIT.

Apart from the slightly different training paradigm between the OSU and AFRL systems, there are three classes of differences between the systems. First, comparing Tables 1 and 2, the major differences are the addition of a sonority class in the OSU features, which gives a first-level broad class categorization of the data, as well as a tense/lax distinction; the AFRL features give a more fine-grained categorization of the articulatory space (separating, for example, labial into labial and labiodental, among other distinctions).

A second distinction is the use of binary classifiers (AFRL) versus n-ary classifiers (OSU). The latter representation necessitates a not-applicable (N/A) entry in every field; we have found previously that these N/A entries actually work well as broad-class detectors – for example, N/A for the place feature usually fires when encountering a vowel or silence [7].

Finally, a third difference between the systems lies in the representation of detection scores. The OSU system uses the posterior estimate of each class (whether phone or phonological feature) as the representation of the input to the CRF. The AFRL system linearizes the output of the final MLP and takes the difference between the positive and negative linear scores. As described in Section 3, this can be thought of as a log posterior ratio (LPR).

## 2.1. Experimental design

We carried out an experiment where we combined the OSU.phones posterior estimates and the OSU.phono.features posterior estimates generated by different feature detectors with the AFRL features. These baseline MLP systems and their various combinations with the AFRL features were used as input for training the CRF recognizer. We have found CRFs to be effective combiners of this type of information since they are less sensitive to redundant, overlapping information [7].

CRF combination techniques are described more fully in [5], but briefly, for an observed data sequence $X$ and a set of random variables over corresponding label sequences $Q$, Conditional Random Fields provide a probabilistic framework for calculating the probability of $Q$ globally conditioned on $X$ [8]. For ASR, the input sequence $X$ corresponds to representations of a series of frames of speech data (acoustic input), while the label sequence $Q$ corresponds to the phone sequence assigned to the input sequence. The probabilistic model of a phone sequence $Q$, given some acoustic input $X$, is mediated in this model through a set of weighted functions $f_i$:

$$P(Q|X) = \frac{\exp(\sum_i \sum_k \lambda_i f_i(q_{i-1}, q_i, X, t))}{Z(X)}$$

(1)

where the $\lambda_i$ are the weights assigned by the learning algorithm. The main issue is how to define the observation functions $f_i$ that relate $Q$ to $X$; in these experiments we variously use the OSU posterior features or AFRL scores as feature functions.

In all experiments, we trained on the TIMIT si/sx training utterances and tuned learning rates on a 400 utterance development set. For the OSU features, MLP posterior estimators were trained from an input representation of 13 PLP coefficients with deltas and double-deltas, a context window of 9 frames, and 2000 hidden units. The AFRL features are identical to those described in [1]. For each system, a CRF was trained over the relevant inputs to predict 48 phone classes, which were reduced to 39 following standard TIMIT evaluation procedures [9].

## 2.2. Results

Table 3 shows the phone accuracy comparison of the different features and their various combinations for the TIMIT task on the 400-sentence development set, the 192-sentence core test set, and the 944-sentence “enhanced” set, which corresponds to all test sentences not in the development set. Statistical significance at the $p \leq 0.05$ level between systems is approximately 0.9% for the dev set, 1.4% for the core test set, and 0.6% for the enhanced set using a one-tailed z-test.

The AFRL features performed remarkably poorly in our initial test (System 1a), where they were fed directly into the CRF as feature functions. However, it was soon apparent that the large dynamic range of the features interacted poorly with the stochastic gradient algorithm, and we found that variance normalizing the AFRL scores produced a result (System 1b) that was much closer to that obtained by the OSU systems (2,3), although a statistically significant performance difference remained.

However, the best results come from throwing all the features into the CRF phone recognition system. When all the features are combined together (Systems 1+2+3), the test set accuracy increases by about 2% absolute versus the best single system across all of the test sets. This means that combined feature representations are better than any single feature representation.

Interestingly, we find that the performance of combining the AFRL and OSU phonological feature systems (Systems 1+3) has two properties: (i) it provides about the same level of performance as combining the OSU phone and phonological

<table>
<thead>
<tr>
<th>Model</th>
<th>Feature Space</th>
<th>dev acc</th>
<th>core acc</th>
<th>enh acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a) AFRL (no norm.)</td>
<td>36</td>
<td>55.0</td>
<td>52.6</td>
<td>52.9</td>
</tr>
<tr>
<td>1b) AFRL (norm.)</td>
<td>36</td>
<td>68.9</td>
<td>66.8</td>
<td>67.8</td>
</tr>
<tr>
<td>2) OSU.phone</td>
<td>61</td>
<td>71.3</td>
<td>69.6</td>
<td>70.4</td>
</tr>
<tr>
<td>3) OSU.phono.features</td>
<td>44</td>
<td>70.6</td>
<td>67.8</td>
<td>68.9</td>
</tr>
<tr>
<td>Systems 2+3</td>
<td>105</td>
<td>72.7</td>
<td>70.3</td>
<td>71.4</td>
</tr>
<tr>
<td>Systems 1+3</td>
<td>80</td>
<td>72.5</td>
<td>70.9</td>
<td>71.4</td>
</tr>
<tr>
<td>Systems 1+2+3</td>
<td>141</td>
<td>73.5</td>
<td>71.5</td>
<td>72.2</td>
</tr>
</tbody>
</table>

Table 3: Percent phone accuracies on TIMIT for development, core test, and enhanced test sets for the baseline n-ary features and their combinations. Significance at the $p \leq 0.05$ level is approximately 0.9%, 1.4% and 0.6% for these data sets, respectively.

\(^1\)The state-of-the-art results for TIMIT phone recognition are 75% on the Core test set and 79.04% on the full test set [5].
feature systems (Systems 2+3), and (ii) the combination is significantly better than either phonological feature system alone. This suggests that the differences in phonological feature systems are significant enough to provide additional robustness in the phone recognition process.

With the many differences between the systems, it is interesting to know whether the improvement came from the use of binary features, from the use of the log-posteriors, or other factors such as the training paradigm for each system. The remainder of this paper tries to tease apart this issue by addressing the following questions: (1) what is the difference between binary and n-ary representations of the same definition of the phonological feature set using the same training paradigm, and (2) when binary classifiers are used, what should be the method of computing scores for input to the CRF?

To answer these questions, we carried out a second set of experiments where we ran the binary features over our original feature set. The next section describes this method in detail, and illustrates how we extracted the log-linear binary features.

3. Investigating representations for the OSU phonological feature set

The second set of experiments represents speech sounds with binary valued phonological features – every phone is marked as either having a feature (eg. [+voice]) or not having the feature (eg. [-voice]) [10]. The new phonological feature set, defined by the non-N/A features in Table 1, consisted of 37 binary phonological features, $C_i$ (i = 1, ..., 37). One MLP was trained for each feature to predict the posterior probabilities $P(C_i|\mathbf{x})$.

We implemented four different score representations for each phonological feature: posterior probability, linearized MLP output, log-posterior ratio and log-likelihood ratio. Linearized MLP outputs, often used in Tandem-style ASR systems, amount to discarding the softmax normalization and exponentiation, and represent an unscaled log posterior. The latter two methods are described below.

In traditional methods of binary classification, the Log Likelihood Ratio (LLR) expresses class confidence as a function of the ratio between the likelihood of $P(\mathbf{x}|C_i)$ and $P(\mathbf{x}|\neg C_i)$. Similarly, we can define a posterior ratio (PR) between classes as:

$$PR = \frac{P(C_i|\mathbf{x})}{P(\neg C_i|\mathbf{x})}$$  \hspace{1cm} (2)

Taking the log of PR we can turn the posterior ratio into a difference of log-posteriors (log posterior ratio, or LPR).

$$LPR = \log \frac{P(C_i|\mathbf{x})}{P(\neg C_i|\mathbf{x})} = \log P(C_i|\mathbf{x}) - \log P(\neg C_i|\mathbf{x})$$  \hspace{1cm} (3)

When the posterior estimator is a two-class MLP using a softmax output layer, the LPR has a particularly nice form due to the form of the underlying estimator. If

$$P(C_i|\mathbf{x}) = \frac{\exp(W^+f(\mathbf{x}))}{\exp(W^+f(\mathbf{x}))+\exp(W^-f(\mathbf{x}))},$$

$$P(\neg C_i|\mathbf{x}) = \frac{\exp(W^-f(\mathbf{x}))}{\exp(W^+f(\mathbf{x}))+\exp(W^-f(\mathbf{x}))},$$

where $W^+$ ($W^-$) are the output-layer weights associated with the positive (negative) class, and $f(\mathbf{x})$ are the outputs of the penultimate layer of the MLP (a function of the input), then the LPR may be calculated as:

$$LPR = (W^+f(\mathbf{x})) - (W^-f(\mathbf{x})).$$  \hspace{1cm} (6)

This formulation, involving the difference of two linear outputs from an MLP, is exactly the output of the system used by Ore and Sylh [1], and thus their scoring may be interpreted as a log posterior ratio. The more traditional LLR can be computed via a class-specific offset to the LPR for each binary attribute:

$$LLR = \log \frac{P(C_i|\mathbf{x})}{P(\neg C_i|\mathbf{x})} = \log \frac{P(C_i|\mathbf{x})P(\neg C_i)}{P(\neg C_i|\mathbf{x})P(C_i)}$$

$$= \log \frac{P(C_i|\mathbf{x})}{P(\neg C_i|\mathbf{x})} + \log \frac{P(\neg C_i|\mathbf{x})}{P(C_i)}$$  \hspace{1cm} (7)

The probabilities $P(C_i)$ and $P(\neg C_i)$ were estimated from the training data by counting the number of positive instances and the number of negative instances for each feature.

Each of these score representations were calculated for each feature to obtain 74 binary posterior estimates, 74 binary linear outputs, 37 LPR binary posteriors and 37 LLR binary posteriors. Again following [5], we trained a CRF-based recognizer on each of these inputs, as well as various combinations of inputs; the CRFs were all trained to predict the standard reduced 48-label set [9]. Decoding followed on the development, core, and enhanced test sets; in each case the phone set was mapped to 39 labels, again following the standard practice [9].

3.1. Results

The bottom half of Table 4 gives the results of the experimental systems and their various combinations with the OSU posterior systems. From the table, we see that no system works as well as n-ary features, which corresponds with the results comparing systems 1 (AFRL) and 3 (OSU.phono.feats). For the phonological binary features, there is a statistically significant drop in accuracy over the baseline using n-ary features.

The LPR and the LLR based ASR systems (6,7) achieved far better phone accuracy than the posteriors and linear systems (4,5). This suggests that the use of these ratios can potentially provide an advantage over the posterior estimates. Because the LPR/LLR representation explicitly represents differences between positive and negative evidence, cueing the CRF to the relationship between pairs of features can improve performance; the relationship between positive and negative binary posteriors is not explicitly given in the posterior system.

Comparing the results of the baseline systems and the experimental systems, we find that integrating the best binary features (which, like AFRL, use LPR/LLR representations) with OSU systems 2+3 results in an accuracy which is comparable to the original system 2+3. This strongly suggests that LPR-based representations of binary features are not the cause of the increased performance in the initial experiments, and that maybe the difference is due to the different training paradigm of the features or the way the features were defined.

4. Discussion and Conclusions

In bringing systems from outside the Automatic Speech Attribute Transcription (ASAT) project into the ASAT paradigm,
our initial experiments led to an unexpected result: a phonological feature system that was statistically significantly worse than our baseline systems could, in combination, lead to significant improvements in TIMIT phone recognition. Because of the extensive nature of the differences between the systems, we conducted diagnostic studies designed to tease apart the reasons for the improvement.

Our results show that when phonological features are represented as binary attributes using the same feature set definition, they perform significantly worse than n-ary feature representations; this observation is in line with that of King and Taylor [11], who compare two systems, corresponding to Chomsky-Halle binary features and n-ary features. Results on the TIMIT database show a phone recognition accuracy of about 50% to 60%, with the multi-valued feature models achieving the highest accuracy of 63.5%. When binary features are used with CRFs, it is best to find a way to associate positive and negative classes via a variance-normalized log-ratio metric. Binary posteriors were the worst performing features; this suggests that too much evidence is discarded in insisting that a feature either be present or not.

The fact that ratio-based binary features utilizing the OSU feature definitions did not improve performance in combination with the n-ary OSU systems suggests that having different training paradigms and/or feature definitions may be a fruitful direction for exploration. This is good news for bringing other representations into the ASAT framework, including multi-lingual and landmark-style features. However, we have not yet teased apart the real reason for the improvements we obtained by adding in AFRL’s features; we have only eliminated binary-ness and score representations as likely candidates for the boost in performance.

5. Acknowledgments
The authors would like to thank Brian Ore and Raymond Slyh from the AFRL for providing us with features to work with our system. This work was supported by NSF ITR grant IIS-0427413 and a faculty-student fellowship from the Dayton Area Graduate Studies Institute.

6. References