DISCRIMINATIVELY TRAINED GAUSSIAN MIXTURE MODELS FOR SENTENCE BOUNDARY DETECTION

M. Tomalin & P.C. Woodland

Cambridge University Engineering Department, Trumpington Street, Cambridge, CB2 1PZ, UK
Email: {mt126,pcw}@eng.cam.ac.uk

ABSTRACT

This paper compares the performance of two types of Prosodic Feature Models (PFMs) in a sentence boundary detection task. Specifically, systems are compared that use discriminatively trained Gaussian Mixture Models (MMI-GMMs) and CART-Style Decision Trees (CDT-PFMs), along with task-specific language models, in a lattice-based decoding framework in order automatically to insert Slash Unit (SU) boundaries into Automatic Speech Recognition (ASR) transcriptions of input audio files. It is shown that a system which uses MMI-GMMs performs as well as a system that uses conventional CDT-PFMs. In addition, it is shown that, when the CDT-PFM and MMI-GMM systems are combined by taking weighted averages of their respective probability streams, Error rate improvements of up to 0.8% abs over the CDT-PFM baseline can be obtained for four different test sets.

1. INTRODUCTION

In recent years it has become desirable to produce ASR output that contains information concerning sentence boundaries [9] [12] [13]. The DARPA Effective, Affordable, Reusable Speech-to-Text (EARS) project defined sentence boundary detection in terms of Slash Units (SUs). SUs are sentence-like units, not traditional 'grammatical' sentences, and therefore they can be identified even in informal Conversational Telephone Speech (CTS). In recent studies, it has been demonstrated that information about SU boundaries in ASR transcriptions can improve text readability [2]. The EARS SU Boundary Detection (SUBD) task requires each SU endpoint to be detected in an input signal, and therefore an SUBD system must output a start time and duration for each SU. In addition, the SUs must be subclassified into one of the following subtypes: *statement, question, incomplete*, and *backchannel* [3].

This paper compares the performance of MMI-GMMs and CDT-PFMs. Accordingly, a CDT-PFM SUBD system is described; MMI-GMMs are introduced as a contrasting modelling strategy, and the performance of the CDT-PFM and MMI-GMM systems is compared. Results are also given for systems that combine the CDT-PFM and MMI-GMM modelling strategies. It is shown that simple system combination approaches can produce SU Error rates that are up to 0.8% lower than those produced by either the CDT-PFM and MMI-GMM systems in isolation.

This work was partly supported by DARPA grant MDA972-02-1-0013 and partly by the GALE programme via a subcontract to BBN Technologies. The paper does not necessarily reflect the position or the policy of the US Government and no official endorsement should be inferred.

2. SUBD TRAINING AND TEST DATA

The SUBD systems were built using 40 hours of training data referred to as the ctsrt04 data which was prepared for the EARS project and annotated in accordance with V6.2 of the EARS Meta-Data Extraction (MDE) annotation specification [3]. System performance was explored using four test sets. The dev03f and eval03s data sets (c.1.5 hours each) constituted the development and evaluation data sets for the EARS Rich Transcription 2003 Fall Evaluation (RT-03f) [5], while the ctsdev04 and eval04f data sets (c.3 hours each) were prepared as the development and evaluation sets for Rich Transcription 2004 Evaluation (RT-04) [6]. The test sets were all annotated using V6.2 of the EARS MDE annotation specification [3].

Each token in the ctsrt04 data can be classified as marking either an SU boundary of a specific subtype or else a non-SU boundary depending upon whether it constitutes the final element in an SU or not. Importantly, the ctsrt04 data is dominated by non-SU tokens, as indicated in Table 1.

SU Subtype	# Tokens	% of ctsrt04 Data
non-SU	404,464	86.8%
statement	36,045	7.7%
backchannel	18,274	3.9%
incomplete	4,378	0.9%
question	3,065	0.7%

Table 1. SU subtype tokens in the ctsrt04 data.

As a result, it is common practice to downsample the training data when PFMs are constructed so that the number of non-SU tokens is equal to the total number of SU tokens [12] [13]. In this paper, the downsampled subset of the total training data will be referred to as the ctsrt04_ds data. SU subtype information for the ctsrt04_ds data is given in Table 2.

SU Subtype	# Tokens	% of ctsrt04_ds Data
non-SU	61,762	50.0%
statement	36,045	29.2%
backchannel	18,274	14.8%
incomplete	4,378	3.5%
question	3,065	2.5%

Table 2. SU subtype tokens in the ctsrt04_ds data.

1-4244-0469-X/06/$20.00

3. SUBD SYSTEM OVERVIEW

The SUBD systems discussed in this paper all use as input the CUED RT-04 CTS 20xRT ASR output, which provides hypothesised token sequences and timing information for the input audio files [12] [1]. The main components of SUBD system are the following:

- task-specific SU Language Models (SULMs)
- task-specific PFMs
- a lattice-based 1-Best Viterbi decoding framework

Fourgram (fg) word-based SULMs, and class-based trigram SULMs with 40 classes (cl40-tg), were constructed. The ctsrt04 data was converted into standard language model training texts, and unique tokens for the SU subtypes were inserted after those lexemes that preceded the SU boundaries. SU tokens were only inserted in the boundary locations, and no special tokens were inserted after lexemes that did not constitute an SU boundary [11]. The word-based SULMs were constructed using Kneser-Ney discounting as implemented in the SRI LM Toolkit [10], while the class-based SULMs were built using the HTK LM Tools [14]. The class-based N-gram SULMs were estimated, and the class-based models were trained using 4 iterations of Cluster [4] [12]. In all the experiments described in this paper an interpolated fg and cl40-tg was used, and these SULMs were interpolated with weights of 0.6 and 0.4 respectively. The free-standing PFMs were constructed using the ctsrt04 data as discussed in detail in section 4.

The SULMs were combined with the PFMs in a lattice-based 1-Best Viterbi decoding framework with empirically determined grammar scale factors (gsf). The probabilities obtained from the PFMs for each token in the test sets were divided by their priors, and the resulting likelihoods were placed on the arcs of the initial lattices which were then expanded using the SULMs and standard HTK lattice tools [14]. The 1-Best decoder output produced token sequences for each file in the test sets, and these contained the ASR lexeme token sequence and SU boundary tokens that had been inserted automatically during the decoding process. The output files were scored using the NIST produced tool md-eval-v19a.pl [7].

The primary scoring metric for the SUBD task simply sums the number of SU boundary insertion, deletion, and substitution errors in a given system output file, when compared to the corresponding reference file, and divides this sum by the number of SUs boundaries in the reference file in order to produce an SU Error (SUErr) score. In this way, the SUErr metric is similar to the familiar WER metric. Full information about the SUBD task and the scoring metric used for RT-04 can be found in [6].

4. DECISION TREE PROSODIC FEATURE MODELS

The CDT-PFMs that provided the baseline for the experiments discussed in this paper were created as follows. Forced alignments were obtained for the ctsrt04 data, and the segmented files were aligned using non-VTLN HLDA MPE triphone models [12]. The alignments provided word sequences and word-level start and end times. Using this timing information, a set of prosodic features was extracted for each lexeme token in the ctsrt04 data. The 10 prosodic features used are given in Table 3.

The prosodic features were extracted either from the waveform data or from corresponding plp encoded data files using ESPS tools (e.g., get_f0) and CUED-internal tools as described in [12]. The features were extracted from 0.2 sec window at the end of each word, and the feature vectors obtained were used in order to construct the CDT-PFMs.

Prosodic Feature	Description
pause	abs pause length after word
durabs	abs duration of word
avg_f0_L	mean of good F0 values in left window
avg_f0_R	mean of good F0 values in right window
avg_f0_ratio	avg_f0_L / avg_f0_R
cnt_f0_L	number of good F0s in left window
cnt_f0_R	number of good F0s in right window
eng_L	RMS energy in left window
eng_R	RMS energy in right window
eng_ratio	eng_L / eng_R

Table 3. Prosodic Features: 'good' F0s values are those that fall between 50Hz and 400Hz. The five f0 features and the three eng features will be referred to as the f0x5 and engx3 features respectively.

5. RESULTS FOR CDT-PFM SYSTEMS

CDT-PFMs, built using the ctsrt04_ds data, were combined during the decoding stage with an interpolated fg and cl40-tg SULM with gsf=1.0. SUErr results for various combinations of prosodic features are given in Table 4.

SYSTEM	dev03f	eval03s	dev04f	eval04f
SULM	63.4	61.7	60.6	62.8
+ pause	58.0	57.2	56.1	57.5
+ durabs	**56.9**	**56.1**	**54.5**	56.4
+ f0x5	57.2	56.3	54.9	**56.3**
+ engx3	57.0	56.1	54.6	56.5

Table 4. CDT-PFM SUErr results for the dev03f, eval03s, dev04f, and eval04f test sets using the ctsrt04_ds data.

Table 4 indicates that the CDT-PFM system that uses only the pause and durabs features performs consistently well across all the test sets.[1] Results for CDT-PFMs that model pause and durabs features using the ctsrt04_ds and ctsrt04 data respectively are given in Table 5.

SYSTEM	dev03f	eval03s	dev04f	eval04f
SULM	63.4	61.7	60.6	62.8
CDT-PFM (ctsrt04)	58.4	56.9	56.8	58.6
CDT-PFM (ctsrt04_ds)	**56.9**	**56.1**	**54.5**	**56.4**

Table 5. CDT-PFM SUErr results for the dev03f, eval03s, dev04f, and eval04f test sets using the ctsrt04_ds and ctsrt04 training data.

Clearly, the performance of the CDT-PFMs system degrades across all test sets when all the ctsrt04 training data is used. This

[1] Various other feature combinations were explored, but the best results across the four test sets were obtained using only the pause and durabs features.

is due to the fact that the non-SU tokens dominate the training data and therefore increase the number of DEL errors. To compensate for this, researchers use various techniques, such as ensembles of 'bagged' features, in order to side-step this fundamental modelling problem [9] [13].

6. DISCRIMINATIVELY TRAINED GMM PROSODIC FEATURE MODELS

As indicated in Section 5, CDT-PFMs have inherent weaknesses as models for the SUBD task. Consequently, it is desirable to explore alternative modelling strategies, and MMI-GMMs are introduced here for this purpose. Although discriminative training techniques have been used with considerable success in large vocabulary ASR systems in recent years [15], they have not standardly been employed in sentence boundary detection systems.

Since the pause and durabs features gave the best performance when the CDT-PFM system was explored, the same features were used when the MMI-GMMs were built, though natural logs of these features were used in order to ensure that the distributions were roughly 'Gaussian'. The ctsrt04 prosodic features for each SU subtype were grouped together, and GMMs were built for each subtype using 4 iterations of Maximum Likelihood (ML) training, creating a set of ML-GMMs.

The ML-GMMs were then further trained using 8 iterations of Maximum Mutual Information (MMI) training, with the LM scale factor in the lattices set to 1. If $\mathcal{O}_1, \mathcal{O}_2, ..., \mathcal{O}_R$ is a training observation sequence for an SU subtype, and if the corresponding transcriptions are $\{s_r\}$, then the MMI objective function for the GMM parameter set λ is as follows:

$$\mathcal{F}_{\mathrm{MMI}}(\lambda) = \sum_{r=1}^{R} \log \frac{p_\lambda\left(\mathcal{O}_r|\mathcal{M}_{s_r}\right) P(s_r)}{\sum_s p_\lambda\left(\mathcal{O}_r|\mathcal{M}_s\right) P(s)} \qquad (1)$$

where $\mathcal{M}_s$ is the composite model that corresponds to the SU token sequence s and $P(s)$ is the (prior) unigram probability of s. The summation of the denominator is taken over all possible SU token sequences. Therefore, as usual, the MMI function maximises the posterior probability of the correct SU token sequence. The MMI training framework was implemented using lattices. The numerator lattices simply consisted of sequences of the relevant SU subtype token (i.e., statement, question, incomplete, backchannel, or non-SU), while the denominator lattices contained *all* possible SU subtypes sequences.[2] In both the numerator and denominator lattices, the SU subtype priors were placed on the arcs. The probabilities produced by the ML-GMMs and MMI-GMMs were used in the decoding framework described in Section 3. The best gsf was determined empirically, and therefore gsf=1.5 for all the GMM experiments.

7. RESULTS FOR ML-GMMS AND MMI-GMM SYSTEMS

ML-GMM and MMI-GMM systems were constructed using both the ctsrt04_ds and ctsrt04 training data, and different numbers of mixtures were explored. Results for the ctsrt04_ds systems are given in Table 6

Table 6 indicates that no single MMI-GMM system gives consistent improvements over the CDT-PFM baseline across all test sets, though gains of up to 0.9% abs can be obtained using MMI-GMMs. Results for the ctsrt04 and ctsrt04_ds MMI-GMM systems are given in Table 7

[2]This implementation is similar to the ASR Frame Discrimination training scheme [8].

SYSTEM	dev03f	eval03s	dev04f	eval04f
SULM	63.4	61.7	60.6	62.8
CDT-PFM	**56.9**	**56.1**	**54.5**	**56.4**
ML-GMM (2m)	57.7	56.5	55.9	57.1
ML-GMM (4m)	57.8	56.3	55.8	57.1
ML-GMM (8m)	57.4	56.0	55.7	57.5
ML-GMM (16m)	57.7	56.1	55.3	57.3
MMI-GMM (2m)	57.2	**55.2**	55.2	**56.5**
MMI-GMM (4m)	**57.0**	55.6	55.4	56.8
MMI-GMM (8m)	57.6	56.2	**55.1**	56.9
MMI-GMM (16m)	57.5	56.4	55.2	56.8

Table 6. SUErr results for the ctsrt04_ds CDT-PFM, ML-GMM, and MMI-GMM systems for the dev03f, eval03s, dev04f, and eval04f test sets. Nm = N mixture components. The GMM systems used gsf=1.5, while the CDT-PFM system used gsf=1.0.

SYSTEM	dev03f	eval03s	dev04f	eval04f
SULM	63.4	61.7	60.6	62.8
MMI-GMM (ctsrt04)	57.3	55.9	**55.0**	56.8
MMI-GMM (ctsrt04_ds)	**57.0**	**55.6**	55.4	**56.8**

Table 7. SUErr results for the ctsrt04_ds and ctsrt04 MMI-GMM systems for the dev03f, eval03s, dev04f, and eval04f test sets. The MMI-GMM systems all use 4 mixture components and gsf=1.5.

When the results in Table 7 are juxtaposed with those in Table 5, it is clear that, unlike the CDT-PFMs, the MMI-GMMs do not require the ctsrt04 data to be downsampled. To consider the difference between these systems in more detail, DEL, INS, and SUB errors for the eval04f test set are given in Table 8.

SYSTEM	DEL	INS	SUB	ERR
CDT-PFM (ctsrt04)	36.1	10.8	11.7	58.6
CDT-PFM (ctsrt04_ds)	30.5	14.1	11.8	56.4
MMI-GMM (ctsrt04)	27.4	17.8	11.6	56.8
MMI-GMM (ctsrt04_ds)	26.8	18.3	11.7	56.8

Table 8. SUErr results for the ctsrt04_ds and ctsrt04 CDT-PFM and MMI-GMM systems for the eval04f test set. The MMI-GMM systems all use 4 mixtures and gsf=1.5, while the CDT-PFM systems have gsf=1.0.

Table 8 indicates that the DEL and INS errors for the ctsrt04 and ctsrt04_ds MMI-GMMs differ by 0.6% abs and 0.5% abs respectively, while the DEL and INS errors for the ctsrt04 and ctsrt04_ds CDT-PFMs differ by 5.6% abs and 3.3% abs respectively. The SUErr patterns indicate that the CDT-PFMs undergenerate when they are trained using all the ctsrt04 data.

8. SYSTEM COMBINATION

Since the CDT-PFMs and the MMI-GMMs model the training data in very different ways, the probability streams obtained from both modelling strategies can be combined. The easiest way to accomplish this is simply to produce a weighted average of each CDT-PFM and MMI-GMM probability pair for each SU subtype associated with each token in the test data sets. The C1 system combines the probabilities obtained from the ctsrt04_ds CDT-PFM and the ctsrt04 MMI-GMM systems, while the C2 system combines the probabilities obtained from the ctsrt04_ds CDT-PFM and the ctsrt04_ds MMI-GMM systems. Various weighting strategies were explored, and it was found empirically that the optimal weightings were 0.5 and 0.5 respectively for the MMI-GMM and CDT-PFM probabilities in the C1 system, and 0.4 and 0.6 respectively for the MMI-GMM and CDT-PFM probabilities in the C2 system. The system combination results using these weightings are given in Table 9.

SYSTEM	dev03f	eval03s	dev04f	eval04f
CDT-PFM (ctsrt04_ds)	56.9	56.1	54.5	56.4
C1	**56.8**	**55.3**	54.4	55.9
C2	56.8	55.4	**54.3**	**55.6**

Table 9. SUErr results for the baseline ctsrt04_ds CDT-PFM system and the C1 and C2 systems for the dev03f, eval03s, dev04f, and eval04f test sets. The C1 and C2 systems use weighted MMI-GMM probability streams that were produced by the 4 mixture MMI-GMMs. The CDT-PFM system used gsf=1.0, while the C1 and C2 systems used gsf=1.2.

Table 9 indicates that the SUErr rates for the C1 and C2 systems are lower than those obtained using any of the systems individually, with the SUErr gains over the CDT-PFM baseline ranging from 0.1% to 0.8% abs.

9. CONCLUSIONS

This paper has demonstrated that discriminatively trained GMMs can be used as PFMs in the CTS SUBD task. It has been shown that MMI-GMMs that are trained using either the ctsrt04 or ctsrt04_ds training data sets achieve SUErr rates across four test sets that are comparable to the SUErr rates produced by a system that uses a CDT-PFM trained on the ctsrt04_ds data. In addition, it has been shown that simple system combination strategy which uses a weighted average of the CDT-PFM and MMI-GMM probability streams can achieve SUErr rates that are up to 0.8% abs lower than those produced by the baseline ctsrt04_ds CDT-PFM system. Since MMI-GMMs have not been standardly used as PFMs in the past, there are many aspects of this modelling framework that have yet to be explored in this context. For instance, the various discriminative training techniques that are conventionally employed in ASR tasks, as well as different sorts of prosodic features, can all be exploited in the GMM modelling framework. Also, the system combination results presented here suggest that contrasting modelling approaches can be combined in order to improve the performance of state-of-the-art SUBD systems.

10. REFERENCES

[1] G. Evermann and H.Y. Chan and M.J.F. Gales and B. Jia and D. Mrva and K.C. Sim, P.C. Woodland, and K. Yu, 'Development of the 2004 CU-HTK English CTS systems using two thousands hours of data' Proc. Fall 2004 Rich Transcription Workshop (RT-04)

[2] D. Jones, F. Wolf, E. Gibson, E. Williams, E. Fedorenko, D. Reynolds, and M. Zissman, 'Measuring the readability of automatic speech-to-text transcripts', Proc. of Eurospeech, 2003

[3] Linguistic Data Consortium, 'Simple Metadata Annotation Specification V6.2', http://www.nist.gov/speech/tests/rt/rt2004/fall/docs/rt04f-eval-plan-v14.pdf May 13th, 2004

[4] G. Moore and S. J. Young, 'Class-based Language Model Adaptation Using Mixtures of Word-class Weights', Proc. ICSLP, 2000

[5] NIST, 'The Rich Transcription Fall 2003 (RT-03F) Evaluation Plan, Version 4', http://www.nist.gov/speech/tests/rt/rt2003/fall/docs/rt03-fall-eval-plan-v9.pdf 9th October, 2003

[6] NIST, 'Fall 2004 Rich Transcription (RT-04F) Evaluation Plan', http://www.nist.gov/speech/tests/rt/rt2004/fall/docs/rt04f-eval-plan-v14.pdf 30th August, 2004

[7] NIST, md-eval-v19a.pl, http://www.nist.gov/speech/tests/rt/rt2004/fall/index.htm

[8] D. Povey and P. C. Woodland, 'Frame Discrimination Training of HMMs for Large Vocabulary Speech Recognition', Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 1999

[9] E. Shriberg and A. Stolke, 'Prosody Modeling for Automatic Speech Recognition and Understanding', Mathematical Foundations of Speech and Language Processing, IMA Volumes in Mathematics and Its Applications, Vol 138, Springer-Verlag, New York 105-114, 2004

[10] A. Stolke, 'The SRI Language Modelling Toolkit' http://www.speech.sri.com/projects/srilm 2004

[11] M. Tomalin and P. C. Woodland, 'Advances in Structural Metadata at CUED', MDE Technical Meeting, Washington DC, 1st May, 2004

[12] M. Tomalin and P. C. Woodland, 'The RT04 Evaluation Structural Metadata Systems at CUED', Proc. Fall 2004 Rich Transcription Workshop (RT-04)

[13] Y. Liu, E. Shriberg, A. Stolcke, B. Peskin, J. Ang, D. Hillard, M. Ostendorf, M. Tomalin, P. C. Woodland, and M. Harper, 'Structural Metadata Research in the EARS Program', Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005

[14] S. J. Young and G. Evermann and M. J. F. Gales and T. Hain and X. L. Liu and J. J. Odell and D. Ollason and D. Povey and V. Valtchev and P. C. Woodland, 'The HTK Book', http://htk.eng.cam.ac.uk, 2003

[15] P. C. Woodland and D. Povey, 'Large Scale Discriminative Training of Hidden Markov Models for Speech Recognition', Computer Speech and Language, vol. 16, no. 1, pp. 25–47 2002