BUT system description for NIST LRE 2007

Pavel Matějka, Lukáš Burget, Ondřej Glembek, Petr Schwarz, Valiantsina Hubeika
Michal Fapšo, Tomáš Mikolov, and Oldřich Plchot

Speech@FIT, Faculty of Information Technology,
Brno University of Technology, Czech Republic
matejkap|burget|glembek|schwarzp|xhubei00|ifapso|imikolov|iplchot@fit.vutbr.cz

1. Introduction

This paper describes Brno University of Technology (BUT) system for LRE 2007 evaluations. The primary system is a fusion of 4 acoustic systems and 9 phonotactic ones. The work builds on our previous LRE 2005 system [1] but also brings several new sub-systems such as binary decision trees, discriminatively trained language models in phonotactic systems, and eigen-channel adaptation in model and feature domain in acoustic systems.

We made one primary and 4 contrastive submissions only for the 'closed set' condition of the 14-class 'General LR' test. All scores can be interpreted as log likelihood ratios.

2. Datasets

2.1. Training data

The following training data (distributed by LDC and ELRA) were used to train our systems (see Table 1 for more details):

CF	CallFriend
CH	CallHome
F	Fisher English Part 1.and 2.
F	Fisher Levantine Arabic
F	HKUST Mandarin
SRE	Mixer (data from NIST SRE 2004,2005,2006)
LDC07	development data for NIST LRE 2007
OGI	OGI-multilingual
OGI22	OGI 22 languages
FAE	Foreigen Accented English
SpDat	SpeechDat-East [1].
SB	SwitchBoard

2.2. Development and Test data

The development and test data were defined by MIT Lincoln Labs (see the Acknowledgments section). They provided us files with nominal duration 3,10 and 30 seconds. The development and test set were based on segments from previous evaluations plus additional segments extracted from longer files from training databases (which were not contained in the training set).

3. Primary system

The primary system was a fusion of 13 sub-systems: 4 acoustic ones and 9 phonotactic. All sub-system descriptions are completed with the "code" of the system for easy identification.

3.1. Pre-processing

The voice activity detection (VAD) is performed by our Hungarian phoneme recognizer, with all the phoneme classes linked to 'speech' class. This VAD was successfully used by BUT for many other tasks.

3.2. Acoustic systems

All acoustic systems used the popular shifted-delta-cepstra (SDC) [2] feature extraction. The feature extraction was the same as in our LRE 2005 system [1]: 7 MFCC coefficients (including coefficient C0) concatenated with SDC 7-1-3-7, which totals in 56 coefficients per frame.

Vocal-tract length normalization (VTLN) performs simple speaker adaptation. VTLN warping factors are estimated using single GMM (512 Gaussians), ML-trained on the whole CallFriend database (using all the languages). The model was trained in standard speaker adaptive training (SAT) fashion in four iterations of alternately re-estimating the model parameters and the warping factors for the training data.

3.2.1. GMM system with 2048 Gaussians per language and eigen-channel adaptation `GMM2048-eigchan`

This sub-system is a variation on BUT GMM system for speaker recognition [5]. Target language models are MAP adapted from a UBM. The eigen-channel adaptation is performed on the models with the eigen-channels derived in the following way:

1. UBM is trained using the original features.
2. for each utterance, a new GMM is obtained by MAP adaptation.
3. a super-vector of means normalized by corresponding standard deviations is obtained from each GMM.
4. maximum of 100 super-vectors per each database and language were selected. That means 100 super-vectors from Arabic CallFriend, 100 from Arabic Fisher, 100 from Arabic Mixer .... (Thai, Bengali, OGI, OGI22, LDC07 and databases with less than 5 utterances were omitted).
5. mean is subtracted from super-vectors over each database (not over language as one would expect)
6. eigen-channels (i.e. directions in which language models are adapted for each test utterance) are given by eigen vectors of covariance matrix estimated from super-vectors (see [5] for details).

Table 1: *Training data in hours for each language and source.*

	sum	CF	CH	F	SRE	LDC07	OGI	OGI22	Other
Arabic	212	19.5	10.4	175	5.93	1.45		0.33	
Bengali	4.27				2.86	1.42			
Chinese	93.2	41.7	1.64	17.2	44.9	4.2	0.87	0.85	
English	264	39.8	4.68	162	34.9		6.77	0.52	15.6 (FAE)
Hindustani	23.5	19.6			0.64	1.32	1.53	0.42	
Spanish	54.3	43.8	6.71		2.63		1.18	0.38	
Farsi	22.7	21.2			0.03		1.00	0.42	
German	28.2	21.6	5.10				1.12	0.38	
Japanese	23.9	19.1	3.47				0.87	0.35	
Korean	19.7	18.4			0.09		0.72	0.5	
Russian	15.1				3.38	1.33		0.43	10.0 (SpDat)
Tamil	19.6	18.4					0.96	0.26	
Thai	1.45				0.15	1.23			
Vietnamese	21.6	20.6					0.79	0.27	
Other	62.5	20.7					1.10	3.29	37.4 (SpDat)

To speed up evaluation of individual language models, only 10 best scoring Gaussians from UBM were considered for eigen-channel adaptation and likelihood computation.

3.2.2. GMM-MMI: `GMM256-MMI`

This subsystem uses GMM models with 256 Gaussians per language, where mean and variance parameters are re-estimated using Maximum Mutual Information criterion - the same as for LRE2005 [1].

3.2.3. GMM-MMI with channel compensated features: `GMM256-MMI-chcf`

Similar set of GMM models with 256 Gaussians per language are trained with Maximum Mutual Information criterion. However, the features are first compensated using eigen-channel adaptation in feature domain [3, 4]. Starting from target language models with means MAP adapted from UBM using the compensated features, only mean parameters are further re-estimated using MMI criterion.

3.2.4. SVM on GMM super-vectors: `GMM512-SVM`

Here the GMMs with 512 mixture components were MAP adapted from a UBM. These were used as features for SVM classifier, similarly as in our speaker recognition system [6], and MIT language recognition work [14].

3.3. Phonotactic systems

The phonotactic systems were based on 3 phoneme recognizers: two left-context/right-context hybrids and one based on GMM/HMM context dependent models.

3.3.1. Hybrid phoneme recognizers

The phoneme recognizer is based on hybrid ANN/HMM approach, where artificial neural networks (ANN) are used to estimate posterior probabilities of phonemes from Mel filter bank log energies using the context of 310ms around the current frame. Hybrid recognizers were trained for Hungarian and Russian on the SpeechDat-E databases. For more details see [7, 8].

3.3.2. GMM/HMM phoneme recognizers

The third phoneme recognizer was based on GMM/HMM context dependent state clustered triphone models, which are trained in similar way as the models used in AMI/AMIDA LVCSR [9]. The models were trained using 2000 hours of English telephone conversational speech data from Fisher, Switchboard and CallHome databases. The features are 13 PLP coefficients augmented with their first, second and third derivatives projected into 39 dimensional space using HLDA transformation. The models are trained discriminatively using MPE criterion [18]. VTLN and MLLR adaptation is used for both training and recognition in SAT fashion. The triphones were used for phoneme recognition with a bi-gram phonotactic model trained on English-only data. Only 3 hours of data per language were decoded and used to train phonotactic models based on this phoneme recognizer.

All the recognizers were able to produce phoneme strings as well as phoneme lattices. In case of lattices, posterior-weighted counts ("soft-counts") were used in the following processing [10].

The modeling in the individual phonotactic subsystems is outlined in Table 2 and in the following paragraphs.

3.3.3. 4-gram language model based on strings: `HU_strLM`, `EN_strLM`

These systems use 4-gram model estimated on phoneme strings from the Hungarian LCRC and English GMM/HMM phoneme recognizers. In the case of Hungarian phoneme recognizer, the LM for each language was derived by interpolating several LMs. At first, we estimated individual LMs using each data source and each language (separate LM for Arabic Fisher, Arabic CF,....). Then we interpolated these separate LMs with LMs estimated on all data from the target language. In the case of English phoneme recognizer, the final target language LMs were interpolated with single LM trained on all languages together. This was helpful because of the limited amount of data to train LMs (at most 3 hours per language). The interpolation weights were tuned to give minimal perplexity on development set. Witten-Bell smoothing was used. Pruning using minimal count was applied. The thresholds 2,3 and 8 for bi-, tri- and four-grams respectively were found to perform well on the development data.

Table 2: *Phonotactic systems.*

type	recognizer	HU [LCRC hybrid]	RU [LCRC hybrid]	EN [GMM/HMM]
string	4-grams	HU_strLM		EN_strLM
lattice counts	2-grams LM	HU_LM-MMI		
	3-grams LM	HU_LM	RU_LM	
	decision trees	HU_TREE_A3E7M5S3G3_LFA	RU_Tree	EN_Tree
	SVM	HU_SVM-3gram-counts		

3.3.4. 3gram language model based on lattice counts: `HU_LM, RU_LM`

The phonotactic models were based on soft-counts. They were adapted from "UBM" trained on all data in the same way as described in [12] for decision tree based phonotactic models.

3.3.5. Binary decision trees: `HU_TREE_A3E7M5S3G3_LFA, RU_Tree and EN_Tree`

In all our systems, binary decision tree language modeling was based on creating a single language independent tree (referenced as "UBM") and adapting its distributions to individual language training data, as described in Navratil's work [11, 12]. While the sub-systems built on Russian LCRC and English GMM/HMM phoneme recognizers use this basic approach only, the Hungarian output was processed in a more complex way:

Instead of merging all resources (databases) of one language together for a UBM adaptation, those resources with large amount of data were "hand-clustered", and a single LM was created for each of these clusters (e.g. 7 LMs for English, see the abbreviation `A3E7M5S3G3`). Such hand-clustering reflected some specifics such as foreign-accented English, different dialects, etc. A linear backend is used to post-process these individual outputs to come up with one score per language.

Latent factor analysis (LFA) was used to compensate for inter-session variation in decision tree modeling:

Let us denote the concatenation of leaf log distributions as a column super-vector $\mathbf{d}$ (around 68k elements in our case). The standard way to evaluate the tree score for utterance s is to compute the inner product of $\mathbf{d}$ and clustered input data $\mathbf{n}_s$, where n_s^i are counts:

$$score_s = \mathbf{n}_s^{\mathrm{T}} \mathbf{d}$$

In LFA, we define a transform matrix $\mathbf{V}$ with the same number of rows as is the dimensionality of $\mathbf{d}$ and the number of columns corresponding to the desired number of vectors of factors. The LFA objective function and output score are computed as:

$$score_s = \sum_i n_s^i \log \frac{e^{l_s^i}}{\sum_{j \in cluster(i)} e^{l_s^j}}, \qquad (1)$$

where $l_s^i = d^i + \mathbf{v}^i \mathbf{x}_s$. Here, $cluster(i)$ corresponds to the leaf to which i belongs. $\mathbf{v}^i$ is i-th row of matrix $\mathbf{V}$, $\mathbf{x}_s$ is a column vector of weights estimated for each utterance by maximizing Eq. 1. $\mathbf{V}$ is estimated by maximization of Eq. 1 summed over a selected sub-set of training utterances balanced among languages. The critical issue is the initialization of $\mathbf{V}$, where simple principal component analysis framework was used. In the evaluation system, the number of factors (size of $\mathbf{x}$) is 4.

3.3.6. trigram lattice counts as super-vectors to SVM: `HU_SVM-3gramcounts`

In this subsystem, the trigram-counts from Hungarian phoneme recognizer were used as features for subsequent classification by SVMs, similarly as in MIT's work [13].

3.3.7. MMI trained bigram language model: `HU_LM-MMI`

Finally, this sub-system included a discriminatively trained language model based on lattice counts. This work is inspired by the discriminative LM training described in [17]. In our case, bi-gram probabilities are iteratively re-estimated using gradient descent algorithm to optimize the MMI objective function.

3.4. Normalization and Calibration

All systems were first processed by linear backend and then fused (or calibrated) using multi-class linear logistic regression [6]. Both linear backend and fusion parameters were trained using our development data. The excellent FoCal Multi-class toolkit by Niko Brummer[2] was used for the preprocessing and fusion.

3.5. Processing speed

Real time factor was approximately 7×RT.

4. Contrast system 1

4.1. System description

The first contrast system is fusion of all subsystems from the primary system and three confidence measures derived from English LVCSR system. The confidence measures are based on:

- lattice width
- number of active words
- frame word entropy.

Details about the used confidence measures can be found in [15, 16]. The LVCSR is derived from the system developed for AMI/AMIDA project [9] (50k word dictionary, tri-gram language model, acoustic modes are the same as for English phoneme recognizer described in section 3.3.2).

4.2. Processing speed

Real time factor was approximately 10×RT.

[2] http://niko.brummer.googlepages.com/focalmulticlass

5. Contrast system 2

5.1. System description

This system included only 6 the most successful sub-systems. The systems were selected by greedy search based on the results obtained on the development data. The following sub-systems were selected:

1. acoustic GMM with 2048 Gaussians per language and eigen-channel adaptation `GMM2048-eigchan`
2. phonotactic binary decision tree processing lattices from Hungarian phoneme recognizer, with Latent Factor analysis and several trees for the most represented languages `HU_TREE_A3E7M5S3G3_LFA`
3. acoustic SVM system processing means from 512 GMMs `GMM512-SVM`
4. phonotactic 3-gram language model based on lattice counts from Russian phoneme recognizer adapted from UBM `RU_LM`
5. phonotactic 4-gram language model based on strings from English phoneme recognizer `EN_strLM`
6. acoustic GMM models trained using MMI with eigen-channel compensated features `GMM256-MMI-chcf`

5.2. Processing speed

Real time factor was approximately 6×RT.

6. Contrast system 3

6.1. System description

This was only one acoustic system already described in the primary system: GMM with 2048 Gaussians and eigen channel compensation: `GMM2048-eigchan`

6.2. Processing speed

Real time factor was 0.5×RT (0.3 is consumed by the segmentation, since it is done by the phoneme recognizer).

7. Contrast system 4

7.1. System description

Only one phonotactic system mentioned in primary system as Binary tree with LFA on phoneme posterior weighed 3gram counts from Hungarian phoneme recognizer: `HU_TREE_A3E7M5S3G3_LFA`.

7.2. Processing speed

Real time factor was 0.35×RT.

8. Acknowledgments

We would like to thank to MIT Lincoln Labs for preparing the test and dev sets – we were really glad to be able to run our systems on them. Special thanks to Niko Brummer for his excellent FoCal toolkit. Thanks MIT, Niko, and David van Leeuwen for valuable discussions and support to our group. Thanks also to AMIDA LVCSR team coordinated by Thomas Hain.

This work was partly supported by European project AMIDA (IST-033812), by Grant Agency of Czech Republic under project No. 102/05/0278 and by Czech Ministry of Education under projects No. MSM0021630528 and LC06008. Lukáš Burget was supported by Grant Agency of Czech Republic under project No. GP102/06/383. The hardware used in this work was partially provided by CESNET under projects Nos. 162/2005 and 201/2006.

9. References

[1] P. Matejka, L. Burget, P. Schwarz, and J. Cernocky: Brno University of Technology System for NIST 2005 Language Recognition Evaluation, in Proc. Odyssey 2006, San Juan, Puerto Rico, USA, June 2006.

[2] P.A. Torres-Carrasquillo, E. Singer, M.A. Kohler, R.J. Greene, D.A. Reynolds, and J.R. Deller Jr., "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *Proc. International Conferences on Spoken Language Processing (ICSLP)*, Sept. 2002, pp. 89–92.

[3] V. Hubeika, L. Burget, P. Matejka and J. Cernocky: Channel Compensation for Speaker Recognition, poster at MLMI 2007, Brno, June 2007.

[4] F. Castaldo, E. Dalmasso, P. Laface, D. Colibro and C. Vair: Language identification using acoustic models and speaker compensated cepstral-time matrices, Proc. ICASSP 2007.

[5] L. Burget, P. Matejka, P. Schwarz, O. Glembek, and J. Cernocky: Analysis of feature extraction and channel compensation in GMM speaker recognition system, In: IEEE Transactions on Audio, Speech, and Language Processing, Vol. 15, No. 7, 2007, pp. 1979-1986, ISSN 1558-7916.

[6] N. Brümmer, L. Burget, J. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D. van Leeuwen, P. Matejka, P. Schwarz, A. Strasheim: Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006, In: IEEE Transactions on Audio, Speech, and Language Processing, Vol. 15, No. 7, 2007, pp. 2072-2084, ISSN 1558-7916.

[7] P. Schwarz, P. Matějka, and J. Černocký, "Hierarchical structures of neural networks for phoneme recognition," in *Proc. ICASSP*, Toulouse, France, May 2006, pp. 325-328.

[8] P. Schwarz, P. Matějka, and J. Černocký, "Towards lower error rates in phoneme recognition," in *Proc. International Conference on Text, Speech and Dialogue*, Brno, Czech Republic, Sept. 2004, pp. 465–472.

[9] T. Thomas, V. Wan, L. Burget, M. Karafiát, J. Dines, J. Vepa, G. Garau and M. Lincoln: The AMI System for the Transcription of Speech in Meetings, In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2007), Hononulu, 2007, pp. 357-360.

[10] J.L. Gauvain, A. Messaoudi and H. Schwenk, "Language Recognition using Phone Lattices," in *Proc. International Conferences on Spoken Language Processing (ICSLP)*, Sept. 2004, pp.1283–1286.

[11] J. Navratil: Spoken language recognition-a step toward multilinguality in speech processing, in IEEE Trans. on Speech and Audio Processing, Vol. 9, No. 6, pp. 678-685 ISSN: 1063-6676, September 2001.

[12] J. Navratil: "Recent advances in phonotactic language recognition using binary-decision trees," in *Proc. International*

Conferences on Spoken Language Processing (ICSLP), Pittsburgh, PA, October 2006

[13] W.M. Campbell, F. Richardson, and D.A. Reynolds: Language Recognition with Word Lattices and Support Vector Machines, in Proc. ICASSP 2007.

[14] W.M. Campbell, D.E. Sturim, D.A. Reynolds, and A. Solomonoff: SVM Based Speaker Verification using a GMM Supervector Kernel and NAP Variability Compensation, in Proc. ICASSP 2006.

[15] L. Burget, P. Schwarz, P. Matějka, M. Hannemann, A. Rastrow, C. White, S. Khudanpur, H. Hermansky, and J. Černocký: Combination of strongly and weakly constrained recognizers for reliable detection of OOVs, submitted to ICASSP 2008.

[16] F. Wessel, R. Schlüter, K. Macherey and H. Ney: "Confidence measures for large vocabulary continuous speech recognition", *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 3, pp. 288–298, 2001.

[17] Kuo, H.-K.J. Fosle-Lussier, E. Hui Jiang Chin-Hui Lee: Discriminative training of language models for speech recognition, in Proc. ICASSP 2002.

[18] D. Povey, "Discriminative Training for Large Vocabulary Speech Recognition," Ph.D. thesis, Cambridge University, Jul. 2004.