UNSUPERVISED CROSS-LINGUAL SPEAKER ADAPTATION FOR HMM-BASED SPEECH SYNTHESIS

Keiichiro Oura, Keiichi Tokuda

Dpartment of Computer Science and Engineering
Nagoya Institute of Technology, Japan
uratec@sp.nitech.ac.jp

Junichi Yamagishi, Simon King, Mirjam Wester

The Centre for Speech Technology Research
University of Edinburgh, UK
jyamagis@inf.ed.ac.uk

ABSTRACT

In the EMIME project, we are developing a mobile device that performs personalized speech-to-speech translation such that a user's spoken input in one language is used to produce spoken output in another language, while continuing to sound like the user's voice. We integrate two techniques, unsupervised adaptation for HMM-based TTS using a word-based large-vocabulary continuous speech recognizer and cross-lingual speaker adaptation for HMM-based TTS, into a single architecture. Thus, an unsupervised cross-lingual speaker adaptation system can be developed. Listening tests show very promising results, demonstrating that adapted voices sound similar to the target speaker and that differences between supervised and unsupervised cross-lingual speaker adaptation are small.

Index Terms— HMM-based speech synthesis, unsupervised cross-lingual speaker adaptation

1. INTRODUCTION

The goal of Speech-to-Speech Translation (S2ST) research is to "enable real-time, interpersonal communication via natural spoken language for people who do not share a common language" [1] and many large-scale projects (Verbmobil, Babylon, TC/LC-STAR, EU-Trans, ATR, etc.) have focused on this topic. In our EU FP7 project EMIME [2], we are developing a mobile device that performs personalized S2ST, such that a user's spoken input in one language is used to produce spoken output in another language, while continuing to sound like the user's voice.

Contrary to previous 'pipeline' S2ST systems that combined isolated automatic speech recognition (ASR), machine translation (MT), and text-to-speech (TTS) systems, or systems that coupled ASR with MT [3, 4], EMIME places the main emphasis on coupling ASR with TTS, specifically to enable cross-lingual speaker adaptation for HMM-based ASR and TTS [5, 6]. The principal modeling framework of speaker-adaptive HMM-based speech synthesis [6] is conceptually similar to conventional ASR systems (although without discriminative training) and it is therefore possible to share Gaussians, decision trees or linear transforms between the two [7].

In the EMIME project, we have conducted extensive experiments exploring the possibilities for combining ASR and TTS models. We have also developed unsupervised adaptation techniques for HMM-based TTS using either a phoneme recognizer [8] or a word-based large-vocabulary continuous speech recognizer (LVCSR) [9], and cross-lingual adaptation techniques for HMM-based TTS [10].

In this paper, we integrate these developments into a single architecture which achieves unsupervised cross-lingual speaker adaptation for HMM-based speech synthesis. We demonstrate an initial S2ST system built for four languages – American English, Mandarin, Japanese, and Finnish. Although all language pairs and directions are possible in our framework, only the English-to-Japanese adaptation was evaluated in the perceptual experiments presented here; these experiments focus on measuring the similarity between the output Japanese synthetic speech to the speech of the original English speaker. The following sections give an overview of the system built, the unsupervised cross-lingual speaker adaptation method and the TTS evaluation results.

2. OVERVIEW OF THE S2ST SYSTEM USING HMM-BASED ASR AND TTS

All acoustic models, for both ASR and TTS, are trained on large conventional speech databases, comprising speech from hundreds of speakers, which were originally intended for ASR: WSJ0/1 (for English), Speecon Mandarin, JNAS (Japanese), and Speecon Finnish databases. Details of the front-end text processing used to derive phonetic-prosodic labels from the word transcriptions can be found in [11].

For each language, state-tied context-dependent speaker-independent HMMs (or multi-space distribution hidden semi-Markov models – MSD-HSMMs) are trained using speaker-adaptive training (SAT) [12]. For the state tying, minimum description length (MDL) automatic decision tree clustering is used [5]. The acoustic features for ASR are either the same as those for TTS or more typical ASR features such as

MFCCs or PLPs. TTS acoustic features comprise the spectral and excitation features required for the STRAIGHT mel-cepstral vocoder with mixed excitation [6]. For unsupervised cross-lingual speaker adaptation and decoding, a multi-pass framework is used: in the first pass, initial transcriptions are obtained from speaker independent (SI) HMMs, and then CSMAPLR adaptation [13] is applied to SAT-HMMs (ASR) using these obtained transcriptions. In the second pass, using these adapted models, the transcriptions are refined. In the final pass, CSMAPLR transforms are estimated for SAT-HSMMs (TTS) with the refined transcriptions. These transforms can then be applied to the SAT-HSMMs for the output language, by employing a state-level mapping that has been constructed based on the Kullback-Leibler divergence (KLD) between pairs of states from the input and output TTS HMMs [10]. The ASR language models used for English, Mandarin and Japanese each contain about 20k bi-grams; the language model for Finnish is a word 10-gram plus a morph bi-gram [14]. For MT we simply used Google's AJAX language API[1]. In future work, this will be replaced by our own MT system based on one being developed for the AGILE project[2]. In the TTS module, acoustic features are generated from the adapted HSMMs in the output language [6] and an MLSA filter is used to generate the speech waveform.

3. UNSUPERVISED CROSS-LINGUAL ADAPTATION BASED ON A STATE-LEVEL MAPPING LEARNED USING MINIMUM KLD

A cross-lingual adaptation method based on a state-level mapping, learned using the KLD between pairs of states, was proposed by Wu *et al.* [10] and is summarized here. We call this approach "state-level transform mapping".

3.1. Learning the mapping between states

For each state $^{\forall}j \in [1, J]$ in the output language HMM $\lambda_{\mathrm{output}}$, we search for the state $\widehat{i}$ in the input language HMM λ_{input} with the minimum symmetrized KLD to state j in $\lambda_{\mathrm{output}}$:

$$\widehat{i} = \operatorname*{argmin}_{1 \le i \le I} D_{\mathrm{KL}}(j, i), \tag{1}$$

where $\lambda_{\mathrm{output}}$ has J states and $D_{\mathrm{KL}}(j, i)$ represents the KLD between state i in λ_{input} and state j in $\lambda_{\mathrm{output}}$ (Fig. 1). $D_{\mathrm{KL}}(j, i)$ is calculated as [15]:

$$D_{\mathrm{KL}}(j, i) \approx D_{\mathrm{KL}}(j \,||\, i) + D_{\mathrm{KL}}(i \,||\, j), \tag{2}$$

$$\begin{aligned} D_{\mathrm{KL}}(i \,||\, j) = \frac{1}{2} \ln\left(\frac{|\boldsymbol{\Sigma}_j|}{|\boldsymbol{\Sigma}_i|}\right) - \frac{D}{2} + \frac{1}{2}\mathrm{tr}\left(\boldsymbol{\Sigma}_j^{-1}\boldsymbol{\Sigma}_i\right) \\ + \frac{1}{2}(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)^{\top}\boldsymbol{\Sigma}_j^{-1}(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i), \end{aligned} \tag{3}$$

[1] http://code.google.com/intl/ja/apis/ajaxlanguage/

[2] http://svr-www.eng.cam.ac.uk/research/projects/AGILE/

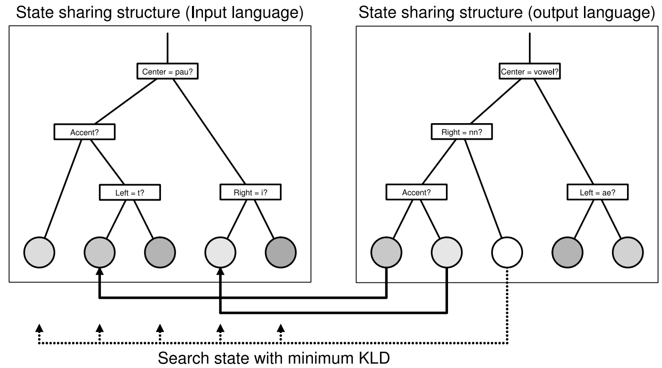

Fig. 1. The state-mapping is learned by searching for pairs of states that have minimum KLD between input and output language HMMs. Linear transforms estimated with respect to the input language HMMs are applied to the output language HMMs, using the mapping to determine which transform to apply to which state in the output language HMMs.

where $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ represent the mean vector and covariance matrix of the Gaussian pdf associated with state i.

3.2. Estimating the transforms for the input language HMM

Next, we estimate a set of *state-dependent* linear transforms $\widehat{\Lambda}$ for the input language HMM λ_{input} in the usual way:

$$\begin{aligned} \widehat{\Lambda} &= \left(\widehat{\boldsymbol{W}}_1, \cdots, \widehat{\boldsymbol{W}}_I\right) \\ &= \operatorname*{argmax}_{\Lambda} P(\boldsymbol{O}|\lambda_{\mathrm{input}}, \Lambda)P(\Lambda), \end{aligned} \tag{4}$$

where $\boldsymbol{W}_i$ represents a linear transform for state i, I is the number of states in λ_{input}, and $\boldsymbol{O}$ represents the adaptation data. $P(\Lambda)$ represents the prior distribution of the linear transforms, which is a uniform distribution for MLLR and CMLLR and a matrix variate normal distribution for SMAPLR and CSMAPLR [13]. Note that the linear transforms will usually be tied (shared) between groups of states known as regression classes, to avoid over-fitting and to enable adaptation of all states, including those with no adaptation data.

3.3. Applying the transforms to the output language HMM

Finally, these transforms are mapped to the output language HMM. The Gaussian pdf in state j of $\lambda_{\mathrm{output}}$ is transformed using the linear transform for state $\widehat{i}$, which is transform $\widehat{\boldsymbol{W}}_{\widehat{i}}$. By transforming all Gaussian pdfs in $\lambda_{\mathrm{output}}$ in this way, cross-lingual speaker adaptation is achieved.

3.4. Unsupervised cross-lingual adaptation

We can extend this method to unsupervised adaptation simply by automatically transcribing the input data using ASR-HMMs. For supervised adaptation, λ_{input} and $\lambda_{\mathrm{output}}$ are both TTS-HMMs (for the input and output languages, respectively). For unsupervised adaptation of HMM-based speech synthesis, λ_{input} may be either a TTS-HMM, or an ASR-HMM that utilizes the same acoustic features as TTS. No other constraints need to be placed on the ASR-HMM. In particular, it does not need to use prosodic-context-dependent-quinphones (which would be necessary for TTS models).

4. EXPERIMENTS

4.1. Experimental conditions

We performed experiments on unsupervised English-to-Japanese speaker adaptation for HMM-based speech synthesis. An English speaker-independent model for ASR and average voice model for TTS were trained on the pre-defined training set "SI-84" comprising 7.2k sentences uttered by 84 speakers included in the "short term" subset of the WSJ0 database (15 hours of speech). A Japanese average voice model for TTS was trained on 10k sentences uttered by 86 speakers from the JNAS database (19 hours of speech). One male and one female American English speaker, not included in the training set, were chosen from the "long term" subset of the WSJ0 database as target speakers. The adaptation data comprised 5, 50, or 2000 sentences selected arbitrarily from the 2.3k sentences available for each of the target speakers.

Speech signals were sampled at a rate of 16 kHz and windowed by a 25ms Hamming window with a 10 ms shift for ASR and by an F_0-adaptive Gaussian window with a 5 ms shift for TTS. ASR feature vectors consisted of 39-dimensions: 13 PLP features and their dynamic and acceleration coefficients. TTS feature vectors comprised 138-dimensions: 39-dimension STRAIGHT mel-cepstral coefficients (plus the zeroth coefficient), $\log F_0$, 5 band-filtered aperiodicity measures, and their dynamic and acceleration coefficients. We used 3-state left-to-right triphone HMMs for ASR and 5-state left-to-right context-dependent multi-stream MSD-HSMMs for TTS. Each state had 16 Gaussian mixture components for ASR and a single Gaussian for TTS. For speaker adaptation, the linear transforms $\boldsymbol{W}_i$ had a tri-block diagonal structure, corresponding to the static, dynamic, and acceleration coefficients. Since automatically transcribed labels for unsupervised adaptation contain errors, we adjusted a hyperparameter (τ_b in [13]) of CSMAPLR to higher-than-usual value of 10000 in order to place more importance on the prior (which is a global transform that is less sensitive to transcription errors).

4.2. Listening tests

Synthetic stimuli were generated from 7 models: the average voice model and supervised or unsupervised adapted models each with 5, 50, or 2k sentences of adaptation data. 10 Japanese native listeners participated in the listening test. Each listener was presented with 12 pairs of synthetic Japanese speech samples in random order: the first sample in each pair was a reference original utterance from the database and the second was a synthetic speech utterance generated from one of the 7 models. For each pair, listeners were asked to give an opinion score for the second sample relative to the first (DMOS), expressing how similar the speaker identity was. Since there were no Japanese speech data available for the target English speakers, the reference utterances were English. The text for the 12 sentences in the listening test comprised 6 written Japanese news sentences randomly chosen from the Mainichi corpus and 6 spoken English news sentences from the English adaptation data that had been recognized using ASR then translated into Japanese text using MT.

Figure 2 shows the average DMOS and their 95% confidence intervals. First of all, we can see that the adapted voices are judged to sound more similar to target speaker than the average voice. Next, we can see that the differences between supervised and unsupervised adaptation are very small. This is a very pleasing result. However, the effect of the amount of adaptation data is also small, contrary to our expectations. This requires further investigation in future work.

Figure 3 shows the average scores using Japanese news texts from the corpus and English news texts recognized by ASR and translated by MT. It appears that the speaker similarity scores are affected by the text of the sentences. Interestingly the gap becomes larger as the number of adaptation sentences increases; this also deserves further investigation in future work.

5. CONCLUSIONS

In this paper, we described the integration of several techniques we have developed for model adaptation into a single architecture which achieves unsupervised cross-lingual speaker adaptation for HMM-based speech synthesis. The listening tests show very promising results: it has been demonstrated that the adapted voices sound more similar to the target speaker than the average voice and that differences between supervised and unsupervised cross-lingual speaker adaptation are small. It appears that the speaker similarity scores are affected by the text of the sentences, which needs further investigation.

Although all language pairs and directions are possible in our system, only English-to-Japanese adaptation has been evaluated in the perceptual experiments presented here. Evaluation of other language pairs and directions is ongo-

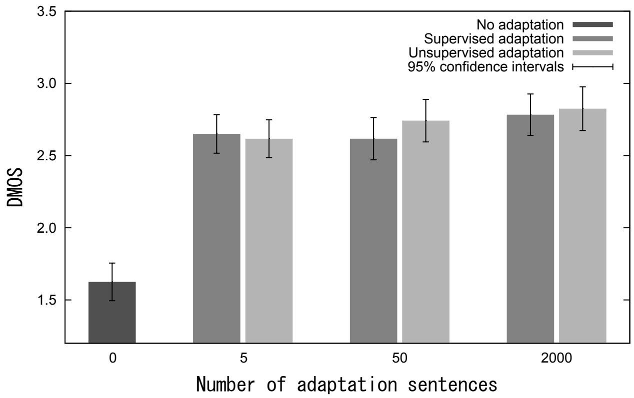

Fig. 2. Experimental results: comparison of supervised and unsupervised speaker adaptation. “0 sentences” means the unadapted average voice model for the output language.

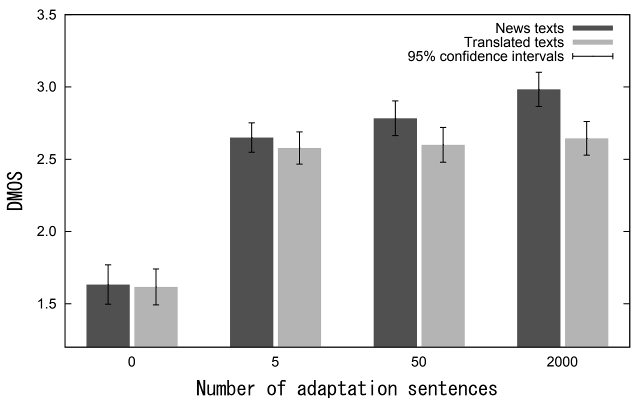

Fig. 3. Experimental results: comparison of Japanese news texts chosen from the corpus and English news texts which were recognized by ASR then translated into Japanese by MT. “0 sentences” means the unadapted average voice model for the output language.

ing. Other future work includes unsupervised cross-lingual speaker adaptation using linear transform estimated directly by ASR-HMMs, which must then use the same acoustic features as TTS-HSMM.

6. ACKNOWLEDGEMENTS

The authors thank Ms. Kaori Yutani and Ms. Xiang-Lin Peng of the Nagoya Institute of Technology for their help with the experiments reported in this paper.

The research leading to these results was partly funded from the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement 213845 (the EMIME project), and the Strategic Information and Communications R&D Promotion Programme (SCOPE), Ministry of Internal Affairs and Communication, Japan. SK holds and EPSRC Advanced Research Fellowship.

7. REFERENCES

[1] F. H. Liu, L. Gu, Y. Gao, and M. Picheny, “Use of statistical N-gram models in natural language generation for machine translation,” Proc. ICASSP 2003, pp. 636–639, 2003.

[2] Effective Multilingual Interaction in Mobile Environments (The FP7 EMIME Project) http://www.emime.org

[3] Y. Gao, “Coupling vs. Unifying: Modeling Techniques for Speech-to-Speech Translation” Proc. EUROSPEECH 2003, pp. 365–368, 2003.

[4] H. Ney, “Speech translation: coupling of recognition and translation,” Proc. ICASSP-99, pp. 517–520, 1999.

[5] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, “Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis,” Proc. Eurospeech, pp. 2347–2350, 1999.

[6] J. Yamagishi, T. Nose, H. Zen, L. Zhen-Hua, T. Toda, K. Tokuda, S. King, and S. Renals, “A robust speaker-adaptive HMM-based text-to-speech synthesis,” IEEE TSALP, 17(6) pp. 1208–1230, 2009.

[7] J. Dines, J. Yamagishi, and S. King, “Measuring the gap between HMM-based ASR and TTS,” Proc. Interspeech 2009, pp. 1391–1394, 2009.

[8] S. King, K. Tokuda, H. Zen, and J. Yamagishi, “Unsupervised adaptation for HMM-based speech synthesis,” Proc. Interspeech 2008, pp. 1869–1872, 2008.

[9] M. Gibson, “Two-pass decision tree construction for unsupervised adaptation of HMM-based synthesis models,” Proc. Interspeech 2009, pp. 1791–1794, 2009.

[10] Y. J. Wu and K. Tokuda, “State mapping based method for cross-lingual speaker adaptation in HMM-based speech synthesis,” Proc. Interspeech 2009, pp. 528–531, 2009.

[11] J. Yamagishi *et al.*, “Thousands of voices for HMM-based speech synthesis,” Proc. Interspeech 2009, pp. 420–423, 2009.

[12] M. J. F. Gales, “Maximum likelihood linear transformations for HMM-based speech recognition,” Computer Speech & Language, 12(2), pp. 75–98, 1998.

[13] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, “Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm,” IEEE Trans. Speech, Audio & Language Process., 17(1), pp. 66–83, 2009.

[14] T. Hirsimäki, J. Pylkkonen, and M. Kurimo, “Importance of high-order N-gram models in morph-based speech recognition,” IEEE Trans. Speech, Audio & Language Process., 17(4), pp. 724–732, 2009.

[15] Y. Qian, H. Lang, and F. K. Soong, “A cross-language state sharing and mapping approach to bilingual (Mandarin – English) TTS,” IEEE Trans. Speech, Audio & Language Process., 17(6) pp. 1231–1239, 2009.