PORTING: SWITCHBOARD TO THE VOICEMAIL TASK

M.J.F. Gales, Y. Dong, D. Povey and P.C. Woodland

Cambridge University Engineering Department
Trumpington Street, Cambridge, CB2 1PZ, UK
{mjfg,yd211,dp10006,pcw}@eng.cam.ac.uk

ABSTRACT

This paper examines techniques that allow a well-trained source system built on one task to be rapidly adapted, or *ported*, to another target task. The two tasks considered in this paper are Hub5, or Switchboard, as the source system and VoiceMail as the target task. The two tasks are acoustically similar, both being telephone-bandwidth speech tasks, but differ in speaking style. SwitchBoard is conversational speech, VoiceMail is a set of voicemail messages. Various porting schemes for acoustic models are examined including discriminative MAP and heteroscedastic LDA. Using around 28 hours of data the error rate on the VoiceMail was reduced by 42% relative compared to the baseline Switchboard performance.

1. INTRODUCTION

As speech recognition performance improves it is being applied to a wider range of applications. It is well known that for best possible performance on any specific task it is necessary to train a system on large amounts of data collected for that particular task. For many applications this is often impractical or too expensive. This has led to the need for techniques that allow either rapidly adapting an existing system to a particular task using little data, sometimes known as *porting*, or building systems that work on a wide range of tasks, *generic* systems.

This paper investigates schemes for rapidly porting a *source* system to a new *target* task. In previous work [1] an initial porting scheme was described. The source acoustic models were adapted using MAP [2] adaptation, in this paper referred to as ML-MAP, and the source language model adapted by interpolating the source language model with a target-task specific language model. Two forms of source acoustic model were investigated, maximum likelihood (ML) trained models and maximum mutual information (MMI) [3] trained models. This set-up will be the baseline porting scheme for this paper. A number of techniques for improving the recognition performance on the target task are investigated.

- **Generic systems**: all available training data is pooled together and treated as single block of data. A system is then trained, either using ML or MMI, estimation on this data. In contrast to the porting schemes, generic systems should yield good performance on both the source and target tasks.
- **Herteroscedastic linear discriminant analysis**: in recent years the use of linear transformations and projections have become popular in large vocabulary speech recognition systems [4, 5]. Using HLDA, a linear projection scheme, it is possible to "tune" a frontend for a specific task and model set. The frontend can thus be "ported" to the target task.
- **Discriminative MAP adaptation**: a new technique related to discriminative training has been proposed for adaptation, MMI-MAP [6]. As the amount of target task data increases this will tend to the discriminatively trained system performance, rather than the ML trained system performance.

This work was funded by the European Commission under the Language project Le-5 Coretex. Extensive use was made of equipment donated by IBM under an SUR award.

The tasks selected to examine the porting problem are Hub5, or SwitchBoard, and VoiceMail [7]. The SwitchBoard system system used as the source system to be adapted to the VoiceMail task. The next section describes the data setups used in the experiments. The results for the various porting schemes are given in section 3.

2. DATA SETS & EXPERIMENTAL SETUP

The source system to be ported to the VoiceMail was a Hub5, or SwitchBoard, system. SwitchBoard is a telephone bandwidth spontaneous speech recognition task. The acoustic training data is obtained from two corpora: Switchboard-1 (Swb1) and Call Home English (CHE). The training corpus consists of a 265 hour training set, 4482 sides from Swb1 and 235 sides from CHE. This is the "h5train00" training set described in [8] and will be referred to as the `swbd` training data. The speech waveforms were coded using perceptual linear prediction cepstral coefficients derived from a Mel-scale filterbank (MF-PLP) covering the frequency range from 125Hz to 3.8kHz. A total of 13 coefficients, including c_0, and their first and second order derivatives were used. Cepstral mean subtraction and variance normalisation were performed for each conversation side. Vocal tract length normalisation (VTLN) was applied in both training and test. The pronunciation dictionaries used in training and testing were originally based on the 1993 LIMSI WSJ lexicon, but have been considerably extended and modified. A gender-independent cross-word-triphone Gaussian-mixture tied-state HMM system was built. The baseline trigram language model, `Swbd-LM`, was built from two sources of data, Broadcast news data (204MW) and SwitchBoard transcriptions (3MW). The interpolation weight between the two model was optimised for the Hub5 task. The Hub5 test set used to evaluate genericity of the systems was a three hour subset of the 2001 development data, `dev01sub`.

The target task was VoiceMail [7] (VM). The data for this task consists of telephone messages addressed to IBM employees. The training data was released in two stages. The first set of training data, `vmtrain1`, consists of 1801 messages of total length 14.6

Subset	Amount	Comment
`1hr`	1.0hr	randomly selected from `vmtrain1`
`4hr`	4.0hrs	randomly selected from `vmtrain1`
`15hr`	14.6 hrs	`vmtrain1`
`20hr`	21.0 hrs	`vmtrain1` + 6.4hrs randomly selected from `vmtrain2`
`30hr`	28.1 hrs	`vmtrain1` + `vmtrain2`

Table 1. Voicemail training data partitions for experiments

hours[1]. In addition a test set, `vmdevtest`, of 42 messages, a total of 11 minutes of data, was released. The second phase of data was originally split into 14.5 hours of training data and 23 minutes of test data, `vmtest2`. Due to the small amount of test data the second set of training data was further split into 13.5 hours of training data, `vmtrain2`, and 60 minutes of test data, `vmtest3`. The training data was then partitioned into subsets as described in table 1. All results quoted are averaged over the three tests, a total of 243 messages, 94 minutes of data. This complete test set will be referred to as `vmtest`.

An important issue in any porting experiment is how the recognition vocabulary is specified. Three vocabularies were used for the experiments described in this paper. For systems using the standard Switchboard language model, `Swbd-LM`, a 27 thousand word vocabulary was used. For language models built on, or interpolated with, the `1hr`, `4hr` and `15hr` VoiceMail subsets, the 27 thousand word Switchboard wordlist had an additional 2,000 words from the VoiceMail `15hr` training data added. For the language models using the `21hr` and `30hr` subsets an additional 3,500 words from the `30hr` subset were added to the original Switchboard wordlist. Where interpolated language models are used for the interpolation weights were optimised on a subset of `vmtest`[2].

3. PORTING SWITCHBOARD TO VM EXPERIMENTS

This section describes the experimental results of porting a source SwitchBoard system to the VoiceMail task. The baseline porting results are an extension to previously published porting results [1] making use of the additional VoiceMail training data, `vmtrain2` and the larger VoiceMail test set `vmtest`.

3.1. Baseline Porting Performance

For this paper the baseline porting approach is to use ML-MAP adaptation [2] to modify the acoustic models and interpolation `Swbd-LM` with a target-task language model for the language model adaptation.

Figure 1 shows the recognition performance of the baseline porting system against amount of VoiceMail data, source acoustic model, and whether the language model was adapted. The baseline performance of the source SwitchBoard, `swbd`, acoustic models on the VoiceMail task with source language model, `Swbd-LM`, was quite poor, 50.8% word error rate for the ML system. As previously noted in [1], using an MMI trained Switchboard system yields improvement even when there is a mismatch between

[1]In previous work on this data [1] `vmtrain1` was incorrectly described as being 20 hours of data.

[2]In experiments this was found to produce a negligible bias.

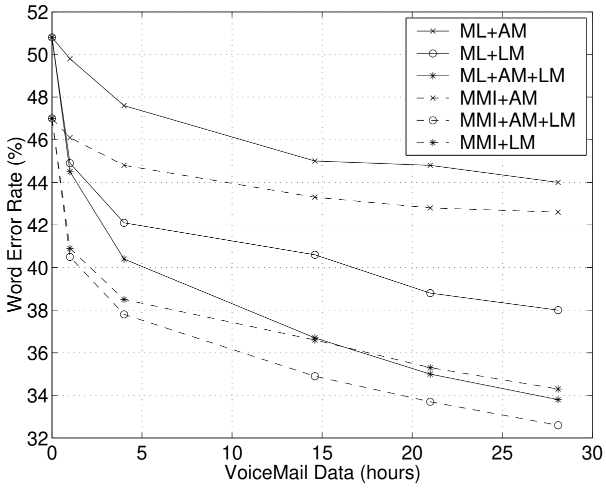

Fig. 1. Performance with amount of adaptation data on `vmtest` for ML and MMI `swbd` trained acoustic models, using ML-MAP adaptation (AM) and/or language model interpolation (LM)

training and test tasks. On the `vmtest` data a 7% relative reduction in error rate was achieved using the MMI source models. Using ML-MAP adaptation the performance of both the ML and MMI `swbd` systems were improved. Using the `30hr` subset the error rate of the adapted ML system using `Swbd-LM` was 44.0% and for the MMI system 42.6%. As expected the difference between the two system decreases as the the amount of porting data increases. As previously observed [1] for this problem the gains obtained by adapting the language model are greater than those obtained from adapting the acoustic models using ML-MAP. The best performance was obtained using the `30hr` data set, adapting both the acoustic and language models using a source MMI trained SwitchBoard system. This gave an error rate of 32.6%, a 36% relative reduction in error rate compared to the baseline SwitchBoard performance.

Training Subset	Acoustic Model	Language Model	System ML	System MMI
15hr	adapt	adapt	36.7	34.9
	pure	adapt	39.9	37.7
	adapt	pure	38.7	37.1
	pure	pure	41.9	40.3
30hr	adapt	adapt	33.8	32.6
	pure	adapt	35.6	33.5
	adapt	pure	35.2	34.0
	pure	pure	37.3	35.5

Table 2. Performance on `vmtest` of adapted Switchboard acoustic and language models (adapt) with "pure" VoiceMail acoustic and language models (pure)

Rather than porting an existing system to a new task, it is possible to simply build a system on the available task-specific data. Table 2 shows the performance of these "pure" systems using the `15hr` and `30hr` training sets on `vmtest`. For both ML and MMI training it was better to port existing systems rather than train pure systems. For example the performance of the pure MMI trained `15hr` system with the adapted language model is about the same as the `4hr` fully adapted MMI SwitchBoard system. The pure `30hr` VoiceMail MMI trained system performance, 35.5%, is worse than that of the `15hr` fully adapted system, 34.9%. Using the baseline porting set-up with upto 28.1 hours of task-specific training data, on this task it is better to adapt a good system to the task, rather than build a pure task specific system.

3.2. Generic SwitchBoard and VoiceMail System

A standard approach to obtaining good performance on a range of tasks is to combine the training data from the multiple tasks and to train a generic system on all the data.

	Test	swbd+VM-30hr-×				
		—	1	2	5	10
ML	vmtest	50.8	47.5	46.5	45.3	44.5
	dev01sub	38.1	37.8	37.8	38.1	38.4
MMI	vmtest	47.0	41.8	41.6	41.6	41.8
	dev01sub	34.8	34.9	35.1	36.0	36.4

Table 3. Performance on `vmtest` and `dev01sub` using ML and MMI trained generic systems with variations in weighting of the VoiceMail `30hr` data using `Swbd-LM`

Table 3 shows the performance on the `vmtest` and `dev01sub` test sets of generic SwitchBoard/VoiceMail systems (`swbd+VM-30hr`). As the amount of training data for VoiceMail is significantly smaller (28.1 hours) than the `swbd` training data (265 hours), systems were built at various weightings of the VoiceMail data. The source language model was used, `Swbd-LM`. Using ML training the performance on `vmtest` improved as the weighting of the Voicemail data increased. Using a weighting of 10 the error rate of this generic system, 44.5%, is similar to the adapted system performance of 44.0%. The performance on the SwitchBoard `dev01sub` data improved slightly when the Voicemail data was added at relatively low weightings (1 or 2). This may be a result of more robustly estimating model parameters. At a weighting of 10 there was only a slight degradation in performance, 38.4%, compared to the pure Switchboard system, 38.1%. The performance of the adapted system on the `dev01sub` data was 39.5%. This is 1.4% absolute worse than the source system and 1.1% worse than the generic system

Table 3 also shows the performance when using MMI training to build a generic acoustic model. For the `vmtest` data there is very little performance difference as the weighting of the VoiceMail data is varied from 1 to 10. The performance at a weighting of 1 for the VoiceMail data, 41.8%, is better than the MMI MAP-adapted system, 42.6%. This is due to the baseline porting scheme using ML-MAP which will tend to the ML system performance. In contrast the generic MMI system solely uses discriminative techniques. For the `dev01sub` test set the performance of the MMI trained system decreased slightly as the weighting of the VoiceMail data increased. The performance at a weighting of 1, 34.9%, was little changed from the baseline system, 34.8%. The ML-MAP-adapted MMI-trained system had an error rate of 37.5% on the `dev01sub` data, significantly worse than that of the MMI trained generic system.

Comparing the two forms of training for building generic systems it is interesting to note that for ML training it was necessary to significantly weight the data, to the point of approximate equal amounts of VoiceMail and SwitchBoard data, whereas for MMI training no weighting was required. Overall building a generic MMI trained system gave slight performance gains, even on the target-task data, than the baseline porting of an MMI source model.

3.3. Heteroscedastic LDA

The systems described so far have been based on MF-PLP frontend with first and second order derivatives. In recent years the use of linear transformations and projections have become popular for speech recognition. This section examines the performance of HLDA, a standard projection scheme, for use with porting and generic system building. For this task the projection was from a 52 dimensional frontend consisting of static, first, second and third order derivatives to a 39-dimensional feature-space. All linear projections were estimated using maximum likelihood training.

HLDA Training	Acoustic Training	VM ×	WER (%)	
			vmtest	dev01sub
swbd	ML	—	49.2	37.0
	MMI	—	45.9	34.0
	ML	10	42.1	37.2
	MMI	1	39.5	34.1
swbd+ VM-30hr	ML	10	42.4	37.5
	MMI	1	39.8	34.5
VM-30hr	ML	10	41.7	37.6
	MMI	1	40.2	36.0

Table 4. Performance on `vmtest` and `dev01sub` using ML and MMI trained generic systems (`swbd+VM-30hr`) and variations in HLDA transform estimation with the `Swbd-LM`

Table 4 shows the performance using a `swbd` trained HLDA transform. Using the source SwitchBoard ML system, the error rate was reduced by about 3% relative, and about 2% for the MMI system on the `vmtest` data compared to the standard frontend. Similar gains were obtained on the `dev01sub` data. Even when using an HLDA transform trained on `swbd` it is still beneficial when recognising VoiceMail data. Table 4 also gives the performance of ML and MMI trained generic systems. In both cases the error rate is reduced by using the Switchboard trained HLDA transform.

Rather than using a generic system the ML-MAP-adapted system may be used with HLDA. Using the `30hr` ML-MAP-adapted MMI system and the `Swbd-LM` language model the error rate on `vmtest` was 40.2% compared to 42.6% for the standard frontend. The error rate on `dev01sub` was 36.6%. For both test sets the performance using HLDA was better than the standard frontend, but the performance was again slightly worse than that of the generic system for `vmtest` and significantly worse on `dev01sub`.

The HLDA projection may also be tuned for the particular task in question. Table 4 shows the effect of training an HLDA transform on both the SwitchBoard and VoiceMail (`30hr`) training data, and on just the VoiceMail (`30hr`) data. There is no consistent significant variation in performance. The original SwitchBoard transform is comparable, or better, than the target task tuned transforms.

3.4. Discriminative-MAP

The baseline porting scheme described in this paper uses ML-MAP adaptation of the source acoustic models. When ML-MAP adaptation is applied with sufficient adaptation data the performance tends to the ML system performance. Thus any gains from improved source models are reduced, or disappear. Recently a modified version of MAP, based on discriminative training, has been proposed [6]. When there is large amounts of adaptation data available, this should tend towards the performance of an MMI trained system.

Adaptation	`1hr`	`4hr`	`15hr`	`20hr`	`30hr`
ML-MAP	46.1	44.8	43.3	42.8	42.6
MMI-MAP	46.1	44.0	41.4	41.1	40.5

Table 5. Performance on `vmtest` comparing standard MAP (ML-MAP) with discriminative MAP (MMI-MAP) from an MMI system using `Swbd-LM`

Table 5 compares the performance of ML-MAP with discriminative MMI-MAP using a MMI-trained SwitchBoard source model. For very limited porting data the performance of both adaptation schemes is the same, 46.1%. As the amount of porting data increases the MMI-MAP schemes makes better use of the available data than the ML-MAP scheme. For the `15hr` VM subset the MMI-MAP scheme is 4% relative better than the ML-MAP scheme and for `30hr` subset 5% better. The performance of the MMI-MAP adapted scheme, 40.5%, is better than the generic MMI trained system, 41.8%. However, the performance of this MMI-MAP `30hr` system on the `dev01sub` test data was 36.6%. Not surprisingly this is significantly worse than the generic system performance of 34.9% as the MMI-MAP adaptation has tuned the system to the VoiceMail task.

3.5. Combined System for Porting

System	Style	Std	HLDA
ML (VM$\times$10)	generic	34.1	32.6
MMI	generic	31.7	30.3
MMI	ML-MAP	32.6	31.4
MMI	MMI-MAP	30.8	29.7

Table 6. Performance on `vmtest` using the VM-`30hr` training subset to produce a generic system (generic) or for adaptation, the SwitchBoard trained HLDA transform, and interpolated language model `Swbd-LM`+VM-`30hr`

The porting schemes may be combined to obtain the best results on the `vmtest` data. Table 6 shows the performance for various combination of porting schemes. The best performance was obtained using an MMI-trained source model, MMI-MAP, HLDA with the interpolated Switchboard language model with the VoiceMail language model. This gave an overall 42% relative reduction in word error rate over the baseline SwitchBoard system and 9% relative reduction over the baseline porting scheme for the `30hr` data. The best generic system performance on the `vmtest` data was using MMI training with HLDA. This was 0.6% absolute worse than the MMI-MAP system, but the performance of this generic system on the `dev01sub` data was significantly better.

4. CONCLUSION

This paper has examined the problem of porting a SwitchBoard system to the VoiceMail task. The baseline porting scheme considered was ML-MAP adaptation for adapting the acoustic models and language model interpolation with a task-specific language model for the language model adaptation. Using 28.1 hours of target-task data the error rate was reduced by 36% relative compared to the source SwitchBoard model performance. The baseline porting performance was significantly better for both acoustic and language models than building pure task specific models. Using MMI training to build a generic system on both the SwitchBoard and VoiceMail data gave a slightly lower error on the VoiceMail task than the baseline porting scheme, with only minimal degradation in the performance on the SwitchBoard task. Using a SwitchBoard trained HLDA transform further decreased the error rate. Discriminative MAP for porting was found to outperform both standard MAP adapted and generic systems. Overall using the best new porting scheme the VoiceMail error rate was reduced by 42% relative compared to the source SwitchBoard system performance and 9% relative compared to the baseline porting scheme. Future work will concentrate on improving the porting scheme for the language model.

5. REFERENCES

[1] R Cordoba, P C Woodland, and M J F Gales, "Improved cross-task recognition using MMIE training," in *Proceedings ICASSP*, 2002.

[2] J L Gauvain and C-H Lee, "Maximum a-posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.

[3] P C Woodland and D Povey, "Large scale discriminative training of hidden Markov models for speech recognition," *Computer Speech and Language*, vol. 16, pp. 25–48, 2002.

[4] N Kumar, *Investigation of Silicon-Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition*, Ph.D. thesis, John Hopkins University, 1997.

[5] M J F Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Transactions Speech and Audio Processing*, vol. 7, pp. 272–281, 1999.

[6] D Povey, P C Woodland, and M J F Gales, "Discriminative MAP for acoustic model adaptation," in *Proceedings ICASSP*, 2003, Paper submitted.

[7] M Padmanabhan, B Ramabhadran, E Eide, G Ramaswamy, L R Bahl, P S Gopalakrishnan, and S Roukos, "Transcription of new speaking styles - Voicemail," in *Proceedings DARPA Hub4 Speech Recognition Workshop*, 1997.

[8] T Hain, P C Woodland, G Evermann, and D Povey, "The CU-HTK March 2000 HUB5E transcription system," in *Proceedings of the Speech Transcription Workshop*, 2000.