TRAINING AND ADAPTING MLP FEATURES FOR ARABIC SPEECH RECOGNITION

J. Park, F. Diehl, M.J.F. Gales, M. Tomalin & P.C. Woodland

Engineering Department, Cambridge University, Trumpington St., Cambridge, CB2 1PZ U.K.
Email: {jhp33, fd257, mjfg, mt126, pcw}@eng.cam.ac.uk

ABSTRACT

Features derived from Multi-Layer Perceptrons (MLPs) are becoming increasingly popular for speech recognition. This paper describes various schemes for applying these features to state-of-the-art Arabic speech recognition: the use of MLP-features for short-vowel modelling in graphemic systems; rapid discriminative model training by standard PLP feature lattice re-use; and MLP feature adaptation using Linear Input Networks (LIN). The use of rapid training using MLP features and their use for short-vowel modelling and LIN adaptation gave reductions in word error rate. However significant improvements over explicit short-vowel modelling with standard multi-pass adaptation were not obtained, although they were useful in combination.

Index Terms— Arabic Speech Recognition, Multi-Layer Perceptron, Acoustic Modelling, Speaker Adaptation

1. INTRODUCTION

MLP-derived features are often added to features derived from standard processing schemes such as PLP to obtain improved Speech-To-Text (STT) performance [1, 2, 3]. MLP features provide additional options for acoustic model training and adaptation. This paper investigates some of the schemes that can be used with MLP features when recognising Arabic.

Modern standard Arabic (MSA) is usually written without the diacritics which specify such things as vowelisation and nunation. This causes non-trivial problems during the development of the acoustic models for Arabic STT systems. To deal with this, two kinds of system are used: graphemic and phonetic. Graphemic systems use a dictionary that is generated by one-to-one letter-to-sound rules. For instance, the word *ktAb* ('book'), is represented as 'k t A b' in a graphemic dictionary. By contrast, phonetic systems use tools such as the Buckwalter Morphological Analyser (version 2.0), referred to as Buckwalter in this paper, to insert hypothetical vowels.[1] Consequently, in a phonetic dictionary, *ktAb* can be written in various ways including 'k i t A b'. However, generating the dictionary word-forms in this way restricts the vocabulary coverage of the phonetic system as Buckwalter is not able to provide an analysis for every Arabic word.

Rather than explicitly representing the short vowels using phonetic models, short-vowel targets can be used for MLP features. These can then be used with a graphemic system to incorporate implicit short vowel modelling, with the advantage that only training data phonetic pronunciations are required while only graphemic pronunciations are required for the test vocabulary.

This work was in part supported by DARPA under 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. We would also like to thank Petr Fousek for making available code for computing wLP-TRAP features.

[1] Available at http://www.qamus.org/index.html.

A major problem with altering features for state-of-the-art STT systems is that the lattices used for many discriminative training schemes must be rebuilt. Though only a cost during training, this can become very expensive when using large amounts of training data, especially if multiple acoustic model types (for example graphemic and phonetic) are used. Therefore a rapid approach that allows the lattice to be re-used is described.

The final aspect investigated is the use of linear input networks (LINs) to adapt the MLP features. LINs have typically been investigated in the context of hybrid systems as the robust adaptation of MLP parameters is not possible [4]. When MLP features are used for training HMMs it is possible to use standard HMM adaptation schemes such as MLLR [5] and CMLLR [6]. However these standard approaches are linear, or base-class specific linear transforms. By contrast the effect of LINs is non-linear on the MLP features. This paper examines the use and combination of LIN adaptation with standard HMM adaptation schemes.

Both rapid training and LIN adaptation are generally applicable to any system, whereas using phonetic targets is Arabic specific. All the schemes are evaluated in a common framework using an Arabic Broadcast News/Conversation transcription task.

2. PHONETIC MLP FEATURES

MLP features are normally derived from a spectral representation of the speech signal. This representation may be the same as one for the standard features to which they are commonly appended, or it may be derived from a completely different form. It is known [7] that using different features as the basis of the MLP-features yields greater reductions in word error rate (WER). In this work the standard features were the PLP features used in many large vocabulary systems [10]. In contrast, the MLP features were derived from wLP-TRAP feature vectors extracted at the same frame-rate [3, 8]. Note that when using these wLP-TRAP features, there are significantly more features (475) than in case of standard PLP-features (39). In this work the 'bottle-neck' approach described in [9] was used. The use of features generated at an intermediate bottle-neck MLP layer has several advantages over using posteriors at the MLP output. In particular, the use of such features removes the need for additional dimensionality reduction. In contrast to phone-posteriors, bottle-neck features provide a properly scaled feature domain (scaled from $-\infty$ to $+\infty$), and they have been shown to be more discriminative [9]. These bottle-neck features (26) are appended to the standard PLP features (39) to yield a 65-dimensional feature vector, referred to as *PLP+MLP features*.

Having determined the form of the features to be used at the network input and those to be extracted from the network, the targets for the MLP training must be specified. There are two forms of acoustic models often used for Arabic STT: graphemic and phonetic. Associated with these there will be different forms of targets that may be

978-1-4244-2354-5/09/$25.00

used for the MLP features[2]. In this work, phone-level targets were used. This allows the MLP features to have some short-vowel information for use with graphemic systems without having to model the short vowels explicitly. The advantage of this approach is that it is only necessary to have phonetic pronunciations for all the words in the acoustic training data, only graphemic pronunciations are required for the test vocabulary.

The standard approach to obtain the phonetic pronunciations is to use Buckwalter. Though this covers approximately 75% of the words in the 350K vocabulary used in this work, it is normally necessary to derive pronunciations for the remaining words. Automatic approaches for doing this are available [10], but such pronunciations are not as reliable as those directly extracted from Buckwalter. However if phonetic pronunciations are only required for words that are seen in acoustic training data, the reliability of the pronunciations can be greatly increased. As there is acoustic data available, it is possible to force-align the acoustic data using a set of possible pronunciations to get the targets. The 'unusual' hypothesised pronunciations that are sometimes automatically derived should never be used. To a lesser extent this is also true of the Buckwalter pronunciations, as Buckwalter is known to over-generate the number of possible pronunciations.

3. RAPID MLP SYSTEM BUILD

State-of-the-art speech recognition systems typically make use of discriminative training schemes. A standard approach for efficient discriminative training is to use lattices as a compact representation of all possible competing paths. This set of possible lattice paths will depend on the exact form of acoustic model being used. Changing either the acoustic features, the decision tree clustering, or the feature linear transforms can significantly change the set of reasonable alternatives encoded as lattice paths which can reduce the effects of discriminative training. For example, in preliminary experiments using lattices derived from non-HLDA features for training HLDA models, the resulting models performed significantly worse than those using HLDA derived lattices.

Altering the acoustic model front-end to incorporate the MLP features should therefore require obtaining new lattices for discriminative training. For large systems trained on hundreds of hours of data, this can be a significant cost. For languages such as Arabic where multiple phonetic and graphemic acoustic models must be built and where these systems cannot share decision trees or linear transforms, it can be extremely expensive to build new sets of lattices for each of the systems. To address this problem, a rapid build approach is proposed, where the system using the MLP-features is constrained to share as much of the standard configuration as possible. This should reduce the impact of shared lattices and dramatically reduce the computational load. The acoustic systems built using the PLP+MLP features were therefore constrained to share the same decision tree and linear feature-transform as the PLP systems. To further make the lattices more appropriate for use, *single-pass retraining* (SPR) from the standard PLP system to PLP+MLP system was used as the starting point. The overall structure for generating the PLP+MLP features is shown in figure 1. The rapid training procedure is:

1. train a standard PLP system including decision tree clustering, HLDA transform, lattice generation, and discriminative training;
2. SPR from the PLP system (39-dimension) to PLP+MLP system (65 dimension)

[2]For this work only phone-level targets are considered. Though a state-level version of this may be used, it was not found to yield WER reductions.

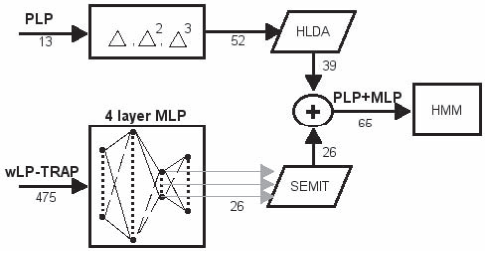

Fig. 1. System Architecture for combined PLP+MLP features.

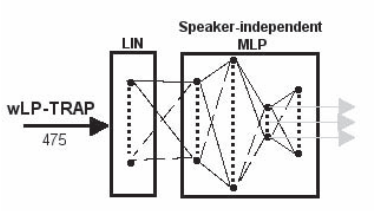

Fig. 2. MLP-feature adaptation by a Linear Input Network.

3. estimate a semi-tied transform for the MLP features and concatenate (to yield a block diagonal matrix) the semi-tied MLP transform with the PLP HLDA transform;
4. multiple Baum-Welch iterations to complete ML training;
5. discriminative training using the PLP lattices.

This process is repeated for each type of acoustic model – in this case both graphemic and phonetic models.

4. NETWORK ADAPTATION

The standard approach for speaker adaptation when using MLP features is simply to use the same form of adaptation applied, for example, to PLP-based systems. MLLR and CMLLR are the most popular versions that are used. It is interesting to compare this form of speaker adaptation with the form used in hybrid systems where it is not possible to robustly adapt the 'acoustic model' parameters. For these hybrid systems LIN adaptation can be used [4]. The LIN used in this paper is illustrated in figure 2. Parameter estimation uses cross-entropy minimisation at the MLP output. Only the network connecting the LIN and the MLP are updated, and the MLP parameters are kept fixed. These two styles of adaptation are very different to one another and may both be applied to MLP features.

MLLR and CMLLR perform a linear transform of either the model-space or the feature-space. To allow more complex transforms multiple base-classes can be used. This yields transforms which are piecewise linear determined by the component being adapted. On the other-hand LIN adaptation acts on the input to a speaker-independent MLP (used to determine phone posteriors). When used with MLP-features the impact of the LIN will be non-linear on the features used for the HMM. However it is not possible to incorporate multiple base-class dependent LINs since, due to the non-linearity impact of the transform on the features, the associated Jacobians cannot be computed. Thus both types of transforms may be described as non-linear, but they achieve this in very different ways. Another aspect in which the two forms of adaptation differ is the training criterion used. For MLLR and CMLLR unsupervised adaptation, maximum likelihood is used to estimate the transform parameters. This directly makes use of the HMM acoustic model parameters. Conversely a frame-level

discriminative criterion is used for LIN parameter estimation, but no use is made of the HMM acoustic model parameters. Both schemes rely on the use of an hypothesis during transform estimation. For the MLLR and CMLLR this is usually in the form of a word-sequence from which statistics are extracted using the forward-backward algorithm. For the LIN adaptation, frame-level targets are required which are obtained by force aligning the hypothesis.

Given the differences between the approaches it is sensible to examine the relative performance of PLP+MLP based systems and whether they are complementary to one another.

5. EXPERIMENTS

5.1. System Description

Two baseline Arabic PLP-based acoustic models were built. The first, a graphemic system (G1) was based on the 36 graphemes. The second, a phonetic system (V1), was based on 39 phones, the graphemic ones plus the three short vowels. For further details of the two systems see [11]. Both models used a 39-dimensional PLP-based front-end which used 13 PLP cepstra, including the zeroth cepstral coefficient with first, second and third delta parameters appended followed by an HLDA projection from 52-dimensions down to 39. Cepstral mean normalisation was also applied. Both systems used the same acoustic training data (just over 1500 hours). Cross-word decision-tree state-clustered triphones were built with about 9K states and an average of 36 Gaussians per state. Minimum phone error (MPE) was used for discriminative HMM parameter estimation. Gender-independent (GI) and gender-dependent (GD) models were then constructed. For the phonetic training pronunciations Buckwalter was used. Any missing pronunciations from the acoustic model training data were obtained using the automatic pronunciation system described in [11].

The PLP+MLP-based systems were built using the process described in section 3. 475-dimensional wLP-TRAP features were used as the input to the MLP. The targets were obtained by force-aligning the training data to get phone boundaries using the PLP phonetic system. A total of 40 phone targets were used, including silence. 26 dimensional bottle-neck features were trained using a 1-of-K coding at the MLP output and cross-entropy minimisation. The size of the hidden layer prior to the bottle-neck layer was set to 3500 to constrain training time. Thus the total number of nodes in the input, hidden, bottle-neck and output layer were $475 \times 3500 \times 26 \times 40$. The ICSI toolkit was used for MLP training [1].

During system development two MLPs were trained. Though structurally identical, the MLP used for the graphemic system reflects an earlier development step and was trained on only 200 hours of data, whereas the MLP for the phonetic system was trained on 1350 hours of data. Table 1 compares both MLPs in terms of frame accuracy, showing an performance gain of 1.5% absolute by the use of the additional 1150 hours of training data. The system performance was evaluated on three test sets dev07 (2.58 hours) dev08 (3.04 hours) and a set not used for development eval07 (2.85 hours). All these test sets consist of both Broadcast News and Broadcast Conversation styles of data. The language model used for these experiments was trained using approximately 1G words. 24 language model components were trained, four components from the STT acoustic data, six were newswire texts, and the rest were webdata which was mainly collected at CUED. The language model interpolation weights were optimised on a range of development sets, which included the dev07 and dev08 test sets. Two forms of word-list were used. The first was based on 350K most frequent words determined using weighted combinations of all the acoustic training sources. The 260K word-list is the subset of the 350K word-list for which phonetic pronunciations could be obtained using Buckwalter. The 90K missing pronunciations were found using automatically derived rules described in [11]. The out-of-vocabulary (OOV) rates for the three test sets used are shown in table 2. The OOV rates for the 260K word-list are far larger than for the 350K. This illustrates that some of the words for which Buckwalter could not derive pronunciations are relatively common.

Training data	Test Acc. (%)
200 hours	63.55
1350 hours	65.16

Table 1. Frame accuracies of the 200 hours and the 1350 hours MLP.

Wordlist	dev07	eval07	dev08
260k	2.68	3.39	2.03
350k	1.19	1.26	1.14

Table 2. Out-of-vocabulary rates for the 260k and 350k wordlists

A multi-pass adaptation framework was used to evaluate the systems. This is a three-stage process. The P1-stage is a fast decoding run with GI-PLP graphemic models. The P2-stage uses GD-PLP graphemic models adapted using Least Square Linear Regression (LSLR) and variance scaling using the P1 supervision. The P2-stage generates trigram lattices which are expanded using a 4-gram language model and then rescored in the P3 stage. The P3-stage models are again GD models (both graphemic and phonetic, PLP and PLP+MLP), adapted using 1-best CMLLR and lattice-MLLR as discussed in [10]. Confusion network decoding is then performed on this output with optional Confusion Network Combination (CNC) to combine two or more branches. For the PLP-system adaptation full CMLLR and MLLR transforms were used. For the PLP+MLP-systems block diagonal transforms were used, one block for the PLP features one for the MLP features.

5.2. MLP Phonetic Modelling

Table 3 shows the P3 CN-decoding outputs for both the phonetic and graphemic systems. The PLP+MLP graphemic system was built using the rapid approach described in section 3. This used MLP features derived from phonetic targets as a method for incorporating information about the short vowels. Comparing the G1 PLP system with the G2 PLP+MLP system shows WER reductions of between 0.5% and

System		Front End	WER		
			dev07	eval07	dev08
Graphemic	G1	PLP	13.2	14.1	14.9
	G2	PLP+MLP	12.6	13.4	14.4
Phonetic	V1	PLP	11.4	12.9	14.0
	V2	PLP+MLP	11.3	12.4	13.7
	V2†	PLP+MLP	11.7	13.0	14.3
G1+V1		CNC	11.0	12.4	12.9
G2+V2			11.0	12.1	12.9
G1+V2			10.7	12.1	12.7
G1+G2+V1+V2			10.5	11.7	12.5

Table 3. Final P3 decoding results contrasting the PLP-front-end versus the mixed PLP/MLP-front-end for graphemic and phonetic systems using 350K vocabulary or 260K vocabulary (indicated with †).

0.7%, but this is still worse than the phonetic PLP system (V1) performance. Similarly the PLP+MLP system (V2) is better than the PLP V1 system. However the gains from the MLP features in case of the phonetic system are smaller than those for the graphemic system. This indicates that the MLP features have incorporated some short-vowel and nunation information into the graphemic system. Though the higher gains for the G2 system are not enough to outperform the explicit short vowel modelling of the phonetic system, the fact that the G2 system applies the 200 hours MLP instead of the much more powerful 1350 hours MLP further emphasises the potential of the implicit short vowel modelling by the MLP of the graphemic system.

nunation information into the graphemic system, though not enough to overcome the explicit modelling of the phonetic system.

Given the possible issues with deriving test-set vocabulary pronunciations for the phonetic systems, the performance of the 260K Buckwalter PLP+MLP phonetic system was examined (V2†). Compared to the 350K V2 system the WER is 0.4% to 0.6% higher. This shows that the rule derived pronunciations are robust even with the more complex acoustic models of the PLP+MLP system.

The combination of various systems was then compared: the PLP systems (G1+V1); PLP+MLP systems (G2+V2); both PLP and PLP+MLP systems (G1+V2, G1+G2+V1+V2). Gains over the best individual system are shown for all configurations. However, no consistent gains were obtained comparing the "pure" PLP (G1+V1) and PLP+MLP (G2+V2) systems, though gains were obtained combining the two frontend ends, G1+V2. The lowest error rate was obtained by combining all four systems together using CNC.

5.3. Speaker Adaptation

To investigate LIN adaptation, a simplified P3 adaptation with the V2 system was initially explored, only CMLLR was used rather than CMLLR plus lattice-MLLR. For LIN adaptation a full network transform (475×475) was trained. A separate LIN transform was estimated for each speaker cluster in the same fashion as CMLLR and MLLR. These initial speaker adaptation results are given in Table 4. The use of LIN adaptation shows gains over not adapting the MLP features, although it is slightly worse than using CMLLR. When the two approaches are combined there is a slight performance improvement on two of the test sets, though the gains are not significant. Table 5 shows the performance when using the full adaptation process with the V2 system. LIN adaptation shows WER reductions over not adapting the features for two of the test-sets. However the WER is higher than when using the CMLLR+latMLLR adaptation. Combining LIN and CMLLR shows no improvements over the standard approach.

PLP Adapt	MLP Adapt	WER dev07	eval07	dev08
P2-Supervision		13.8	15.0	15.6
CMLLR	—	12.0	13.1	14.3
	CMLLR	11.7	12.8	13.9
	LIN	11.8	13.0	14.1
	LIN+CMLLR	11.6	12.7	13.9

Table 4. Evaluation results of LIN adaptation without lattice MLLR.

6. CONCLUSION

This paper has explored the training and adaptation of MLP features in the context of a state-of-the-art large vocabulary Arabic STT system. Three schemes have been investigated. First, the use of MLP features to incorporate short-vowel information into the graphemic system. Though this simplifies decoding as only test set graphemic pronunciations are required, by itself it does not match the performance of PLP or PLP+MLP phonetic systems. However, when used in combination with these phonetic systems, performance gains can be obtained. Second, a rapid training approach for use with the PLP+MLP system was described. This allows the lattices from PLP systems to be re-used. This was the form of training used in all PLP+MLP experiments where gains of around 0.5% were obtained compared to the PLP systems. Finally the use of LIN adaptation as an alternative to the usual HMM-based linear adaptation was described. Though WER reductions using LIN adaptation were obtained compared to the unadapted system, in the configuration investigated, performance was no better than the standard HMM-based adaptation.

PLP Adapt	MLP Adapt	WER dev07	eval07	dev08
CMLLR +latMLLR	—	11.6	12.7	13.8
	CMLLR+latMLLR	11.3	12.4	13.7
	LIN+latMLLR	11.4	12.6	13.8
	LIN+CMLLR+latMLLR	11.3	12.5	13.6

Table 5. Evaluation results of LIN adaptation with lattice MLLR.

7. REFERENCES

[1] Q. Zhu, A. Stolcke, B.Y. Chen, and N. Morgan, "Using MLP features in SRI's conversational speech recognition system," in *Proc. INTERSPEECH*, 2005, Lisbon.

[2] H. Hermansky, D.P.W. Ellis, and S. Sharma, "Connectionist feature extraction for conventional HMM systems," in *Proc. ICASSP 2000*, Istanbul.

[3] G. František, P. Fousek, "Optimizing bottle-neck features for LVCSR," in *Proc. ICASSP*, 2008, Las Vegas.

[4] R. Gemello, F. Mana, and D. Albesano, "Linear input network based speaker adaptation in the Dialogos system," in *Proc. IJCNN*, 1998, Anchorage.

[5] M.J.F. Gales and P.C. Woodland, "Mean and variance adaptation within the MLLR framework," *Computer Speech and Language*, vol. 10, pp. 249–264, 1996.

[6] M.J.F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[7] P. Fousek, personal communication.

[8] P. Fousek, "Extraction of features for automatic recognition of speech based on spectral dynamics," *Ph.D Thesis, Czech Technical University in Prague*, March 2007.

[9] F. Grézl, M. Karafiát, S. Kontár, and J. Černocký, "Probabilistic and bottle-neck features for LVCSR of meetings," in *Proc. ICASSP*, 2007, Hawaii.

[10] M.J.F. Gales, F. Diehl, C.K. Raut, M. Tomalin, P.C. Woodland, and K. Yu, "Development of a phonetic system for large vocabulary Arabic speech recognition," in *Proc. ASRU*, 2007, Kyoto.

[11] F. Diehl, M. J. F. Gales, M. Tomalin, & P.C. Woodland "Phonetic pronunciations for Arabic speech-to-text systems" in *Proc. ICASSP*, 2008, Las Vegas.