Second Order Factorized Model Adaptation for Short Duration Language Identification

Sarith Fernando[1,2], Vidhyasaharan Sethu[1], Eliathamby Ambikairajah[1,2] and Haizhou Li[3]
[1]School of Electrical Engineering and Telecommunications, UNSW Sydney
[2]DATA61, CSIRO, Sydney, Australia
[3]Department of Electrical & Computer Engineering, National University of Singapore
E-mail: sarith.fernando@unsw.edu.au

***Abstract*— Adaptation of deep neural network (DNN) based language identification models is still a challenging area of research. Recently, state-of-the-art approaches to short duration language identification task have made use of bidirectional long short-term memory (BLSTM) recurrent neural network (RNN) language identification models. Although this enables the effective modelling of sequential information, significant mismatch due to different conditions such as speaker, channel, duration and background noise between training and testing data still exists. An adaptation of BLSTM systems can help to reduce such mismatches between training and testing data. In this paper, a transformation to the existing BLSTM layer is proposed, using learning of a second order factorization matrix called a *compensation layer*. The condition-dependent parameters of the factorization matrix are estimated to adapt the BLSTM layer weights. Experiments on the AP17-OLR database show that utterance level adaptation helps to achieve relative improvements of 28% in terms of Cavg over a traditional BLSTM for utterances of '1s' duration.**

I. INTRODUCTION

Mismatch between training and testing data is a long-standing problem in language identification [1-3]. In practice, it is common that sufficient data for long duration utterances are available for system training. However, it is important to note that test utterances may be significantly smaller (~1s). In language identification, these short duration utterances are the most affected due to mismatches in recordings such as speaker, channel and background noise [4]. Although there are a number of ways to enhance the noise robustness of speech features [1, 5, 6], practical systems still fail to produce state-of-the-art performance if the training and testing environments do not match acoustically [1]. A language model trained with sufficient data leads to enhanced language discriminating ability with a better representation of the language's space. Previous work provides evidence for mismatch compensation, including Gaussian Mixture Models (GMMs) [2], the total variability transform (i-vector) approach [7] and techniques such as Probabilistic Linear discriminant analysis (GPLDA) [8], which was designed specifically for channel mismatch conditions. However, since most of these approaches are based on feature statistics, testing with short duration utterances tends to exhibit higher intra-class variability with lower inter-class distance. Consequently, the system performance is significantly degraded [4].

On the other hand, deep neural network (DNN) based approaches, specifically bidirectional long-short term memory (BLSTM) recurrent neural networks (RNNs) have outperformed state-of-the-art approaches and have proven to be effective for short duration language identification [4, 9]. Although, this elegant framework achieves superior performance by capturing sequential information of the input features, these are also vulnerable to mismatch conditions that lead to significant performance degradations, similar to all other prevailing machine learning approaches. This issue can be mitigated with adaptation techniques that adapt an existing model to match better testing conditions [10, 11].

Although there are very few works on the adaptation of language identification models, many efforts have been proposed for speaker adaptation in speech recognition systems [12-14]. The most common way of adapting DNNs is to introduce a linear layer to the input, hidden or output layer in the existing model [15]. Some researchers have also invested great effort in combining DNNs with GMMs in the training of tandem systems [16]. In tandem systems, DNNs are used to extract bottleneck features (where the features form a narrow hidden layer) in order to train GMM models. Instead of model adaptation, input features have been adapted to train DNNs for various applications [8, 17]. Besides these techniques, speaker aware training is one of the most popular adaptation techniques in speech recognition systems [12]. In this approach, speaker information is provided to the network in order to perform speaker normalization in the adaptation stage. The adaptation of DNNs commonly introduces a huge number of parameters, leading to overfitting. However, subspace method [18, 19] based adaptation is only performed on a subset of model parameters, which avoids overfitting. Further, regularization based adaptation methods minimize the distance between training and testing data by training the DNN with an additional error [20].

In our previous work [11], we address the issue of mismatch compensation for short duration language identification by introducing a factorized hidden variability subspace (FHVS) to adapt the existing DNN framework, and showed that significant gains can be achieved over existing adaptation techniques. In contrast to existing adaptation techniques, we introduced a transformation matrix called the hidden variability subspace (HVS) to capture the variability between training and testing utterances. The factorization was

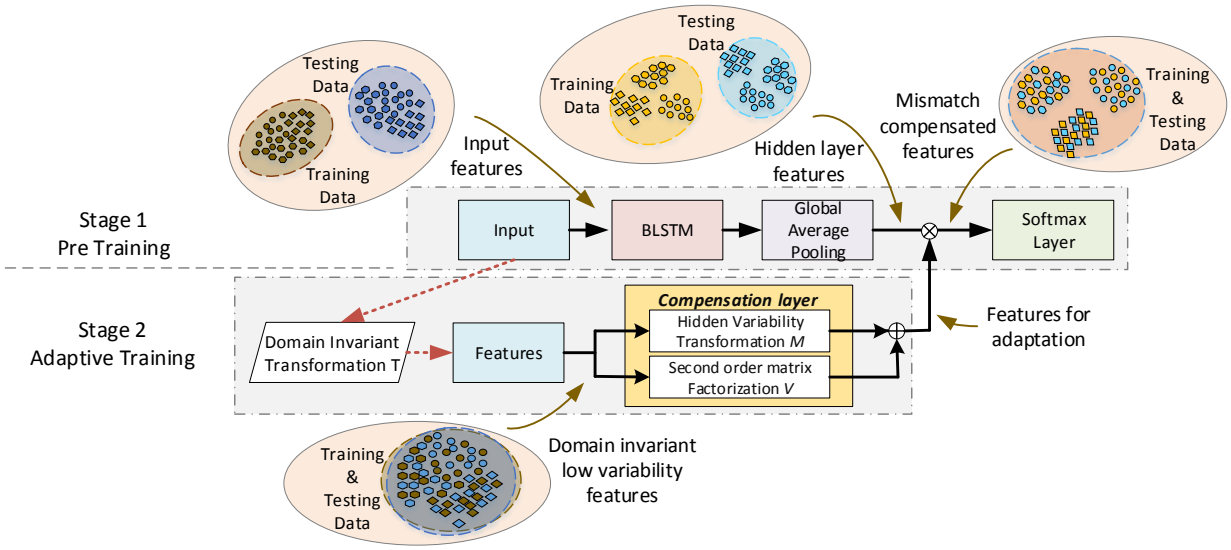

Fig. 1 Proposed second order factorized *compensation layer* for adaptation of BLSTM layer. The marker shapes represent the instance labels and colours represent the original domains. Both training and testing domains are matched together using the unsupervised domain invariant transformation T (low variability subspace) where T is trained using i-vectors. In the *compensation layer* the metric M and V defined the subspace and second order factorization matrix respectively which are learned to minimize the mismatch and to maximize the discriminative power between samples in the BLSTM network. Domain distributions are indicated by dashed ellipsoids. Our learning scheme non-linearly identifies the transformation M and V. This figure is best viewed in colour.

then conducted prior to the subspace learning using linear factorization methods such as linear discriminant analysis (LDA) or singular value decomposition (SVD). Although these factorization techniques provide slight improvements over several other adaptation techniques, these classic linear algebraic methods may not be so effective when embedded in a large-scale nonlinear model. For this reason, in the current work we propose a learning scheme for the factorized model adaptation using second order information (Section III) to perform mismatch compensation. This framework is introduced as a novel layer called the *compensation layer*, (Section II) which can compensate for the mismatch in the existing BLSTM language model.

II. Adaptation of BLSTMs with Utterance Representations

In language identification, bidirectional long short-term memory (BLSTM) recurrent neural networks (RNNs) are used to generate framewise predictions as in [4]. Unlike traditional LSTM networks, the underlying principle of BLSTMs can be thought of as capturing the temporal information of input features in both backward and forward directions. A BLSTM therefore has access to both past and future information in a speech sequence in order to classify a given speech utterance to a specific language. For a length T input vector sequence $\boldsymbol{x_t} = [x_1, x_2, \dots, x_T]$, a conventional BLSTM output $\boldsymbol{y_t} = [y_1, y_2, \dots, y_T]$ can be computed as

$$\boldsymbol{y_t} = W_{\vec{h}y}\vec{\boldsymbol{h}}_{\boldsymbol{t}} + W_{\overleftarrow{h}y}\overleftarrow{\boldsymbol{h}}_{\boldsymbol{t}} \quad (1)$$

$$\vec{\boldsymbol{h}}_{\boldsymbol{t}} = \mathcal{H}\left(W_{x\vec{h}}\boldsymbol{x_t} + W_{\vec{h}\vec{h}}\vec{\boldsymbol{h}}_{\boldsymbol{t-1}} + \boldsymbol{b}_{\vec{h}}\right) \quad (2)$$

$$\overleftarrow{\boldsymbol{h}}_{\boldsymbol{t}} = \mathcal{H}\left(W_{x\overleftarrow{h}}\boldsymbol{x_t} + W_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{\boldsymbol{h}}_{\boldsymbol{t-1}} + \boldsymbol{b}_{\overleftarrow{h}}\right) \quad (3)$$

where $\vec{\boldsymbol{h}}_{\boldsymbol{t}}$ and $\overleftarrow{\boldsymbol{h}}_{\boldsymbol{t}}$ are the forward and backward sequences of the BLSTM hidden states at time t respectively. W and $\boldsymbol{b}$ are the weights and biases of the BLSTM layer. The recurrent hidden layer function $\mathcal{H}$ is derived as in [21] for each LSTM memory block in conventional manner. After the BLSTM layer, global average pooling is conducted over the whole sequence T to transform frame level features to utterance level features yielding the BLSTM output $\boldsymbol{k}$ as,

$$\boldsymbol{k} = \sum_{\forall t \in T} \boldsymbol{y_t} \quad (4)$$

This output $\boldsymbol{k}$ is then given to the next layer (a softmax layer in this paper) to perform the language classification task. In order to compensate for mismatch in this basic model, we introduce a novel *compensation layer* in between this output $\boldsymbol{k}$ and the final softmax layer.

A. Proposed compensation layer: a joint nonlinear adaptation learning scheme

In this work, we propose adapting the obtained model output of the BLSTM $\boldsymbol{k}$, by introducing a nonlinear learning scheme that uses second order utterance aware information (Section III). This facilitates learning more abstract information with the aim of compensating mismatch. An utterance dependent (UD) feature transformation is employed to $\boldsymbol{k}$ in equation (4) as,

$$\boldsymbol{k}' = \boldsymbol{k} \odot \boldsymbol{C} \quad (5)$$

where $\odot$ denotes elementwise multiplication, and $\boldsymbol{C}$ is the proposed mismatch *compensation layer* with nonlinearity $\mathcal{H}$ ('tanh' in this paper),

$$\boldsymbol{C} = \mathcal{H}(\boldsymbol{\omega}^{\boldsymbol{T}} V\boldsymbol{\omega} + M\boldsymbol{\omega} + \boldsymbol{\varphi}) \quad (6)$$

which is constructed from subspace M and the factorization matrix V, and where $\boldsymbol{\omega}$ is the utterance level feature vector, an i-vector in this work. This adapted feature representation $\boldsymbol{k}'$ is passed to the next layer in the adaptation stage instead of $\boldsymbol{k}$. Therefore, during the adaptation process, the M and V matrices are trained to capture the information of the utterance mismatch in the pretrained model. It is worth highlighting that unlike previous work, the factorization matrix V is learned automatically in this learning scheme, and is also able to capture the second order information that is highly significant to the mismatch conditions. This provides a unified and efficient way to capture the mismatch information without any constraints.

III. CONSTRUCTION OF SECOND ORDER FACTORIZATION MODEL

DNNs have proven highly effective at classification tasks with advances in recognition accuracy when the features and classifier are jointly learned [4]. In this end-to-end process, the parameters (weights) of each layer act as feature transformation of the output of proceeding one, when stacking multiple layers. Even though these layer outputs are followed by nonlinearities, the computation of such linear combinations can be thought as extracting first-order statistics for the input features [22]. Therefore, it can be argued that second order statistics such as covariances may not be directly extracted using such networks. However, the usefulness of higher order information in neural networks learning has been shown in [23]. Several studies have been conducted on image classification tasks, extracting covariance based features [22, 24, 25], which also showed the effectiveness of second order statistics. Covariance plays an important role in data mismatch compensation [5], similar to the above-mentioned speech application, and many features can be found that are extracted based on covariance.

In our previous work we showed the effectiveness of employing an adaptation layer within a BLSTM framework using classic covariance based transformations such as SVD and LDA, applied to the low variability feature space (i-vectors). The transformed features increased the robustness of the feature space and could also be used as a factorization technique to reduce the number of dimensions. However, these factorization techniques are linear and applying such transformations on a large nonlinear network may not be as effective. Therefore, in this work, we train a unified second order nonlinear factorization model within the network as part of the adaptation process.

A. Proposed learning of second order information

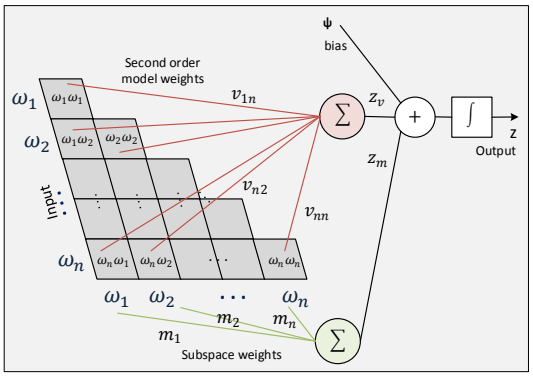

Fig. 2 Joint nonlinear adaptation learning scheme of *compensation layer* for a neuron. The $\boldsymbol{m}$ and $\boldsymbol{v}$ are the weights of subspace M and second order factorization matrix V respectively for a given input feature vector $\boldsymbol{\omega}$.

Learning the second order factorization model is quite challenging due to large number of weight parameters. In this work, we construct the factorization matrix V and force it to learn the covariance information using a second order function of the input vector. The main intuition behind second order factorization is to capture the covariance information between the feature vector dimensions. This is a similar idea to that conveyed by the conventional linear factorization algorithms of SVD and LDA. We argue that this second order factorization facilitates a better learning of the mismatch feature space based on second order statistics, empowering feature separability in individual neurons. We learned the subspace M in the standard form [10]. As shown in Fig. 2, the output z of a single neuron in the *compensation layer* $\boldsymbol{C}$, for an n-dimentional input feature vector $\boldsymbol{\omega} = [\omega_1, \omega_2, \dots, \omega_n]$ can be computed as

$$z = z_m + z_v + \psi \quad (7)$$

where z_m, z_v are subspace output and second order output for a single neuron respectively. The ψ denotes the bias for the same neuron. Equation (7) can be further expanded as

$$z = \sum_{i=1}^{n} m_i\omega_i + \sum_{i=1}^{n}\sum_{j=i}^{n} v_{ij}\omega_i\omega_j + \psi \quad (8)$$

where m_i and v_{ij} are the weights of the subspace M and factorization matrix V respectively. ψ represents the bias for the neuron. Further, $\omega_i\omega_j$ are elements of the outer product of feature vector $\boldsymbol{\omega}$.

The layer optimization can be formulated for the adaptation network output $\mathcal{H}(.)$ in the following manner. The weight parameters v_{ij}, m_i and ψ can be updated using the gradient decent optimization method for a training set with inputs $\omega = \{\omega^1, \omega^2 \dots, \omega^u, \dots\}$ and outputs $o = \{o^1, o^2 \dots, o^u, \dots\}$, where gradients can be calculated as

$$\frac{\partial E}{\partial m_i} = (\mathcal{H}(\omega^u) - y^u)\frac{\partial \sigma}{\partial \omega}\omega_i \quad (9)$$

$$\frac{\partial E}{\partial v_{ij}} = (\mathcal{H}(\omega^u) - y^u)\frac{\partial \sigma}{\partial \omega}\omega_i\omega_j \quad (10)$$

$$\frac{\partial E}{\partial \psi} = (\mathcal{H}(\omega^u) - y^u)\frac{\partial \sigma}{\partial \omega} \quad (11)$$

Therefore, this second order framework can be trained similarly to the standard approach. However, it must be noted that the number of training parameters in the second order factorization model is $a = n(n+1)/2$, while there are only n first order parameters. Furthermore, it is expected that this second order model contains highly correlated parameters due to the consideration of the outer product and higher dimensionality of the feature vectors.

B. *Low-rank matrix factorization*

Low-rank factorization is used to form an abstract representation of the second order information and to reduce the number of parameters and by reducing the training complexity. In the training process, low-rank factorization ensures that there is minimal loss, thereby generally reducing the feature dimensionality as shown in Fig. 3. Also, by discarding features that correspond to covariances that are not relevant to classification/noise, this facilitates generalization of the input feature vectors.

For a d-dimensional layer, it can be seen that $n \times d$ and $a \times d$ parameters exist for M and V matrices respectively, where $a = n(n+1)/2$. The two matrices M and V were introduced in section III and illustrated in Fig. 2. In this paper we aim to represent the weight matrix V as a low-rank matrix as shown in Fig. 3. If V has rank r, then as in [26] there exist a factorization $V = P \times Q$ where P and Q are full rank matrices of size $a \times r$ and $r \times d$ respectively. It must be noted that this decomposition allows the single large matrix V to be implemented as two matrices P and Q without any non-linearity between them using a much lower number of parameters in the second order model in the *compensation layer*. To satisfy the requirement of lower number of parameters than the original layer, $r(a+d) < ad$ we can select an appropriate parameter r. In the experimental Section V, several low rank factorization matrix experiments are run using different ranks, as well as with other constraints such as symmetricity and diagonality to the weight matrices.

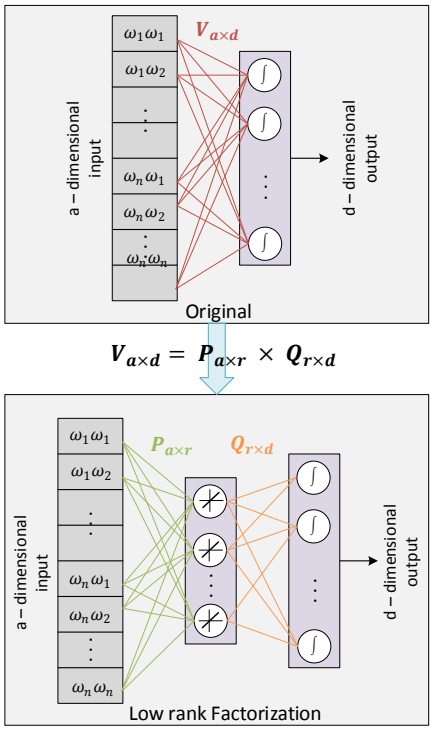

Fig. 3 Schematic of low rank matrix factorization of the second order model.

IV. Feature Extraction and Experimental Setup

Experiments are conducted on AP17-OLR database [27] and Fig. 4 depicts the complete experimental setup. This database is chosen for two main reasons. In order to test our proposed hypothesis, the experimental data should come from different mismatch conditions, and include short duration utterances. AP17-OLR dataset satisfies both of these conditions, as it contains a large number of utterances for training and testing purposes. The dataset contains 10 different languages developed for short duration language identification tasks. The test data has 3 different duration conditions of '1s', '3s' and 'all', where each subset contain 17964, 16404 and 17964 utterances respectively. Additionally, this database utilizes data from two different recording conditions: clean and noisy environmental conditions. In this work, three languages that were recorded in both above conditions (Japanese, Russian and Korean) are designated as 'mismatched', whereas all other languages are 'matched'. Each language contains around 10 hours of training data

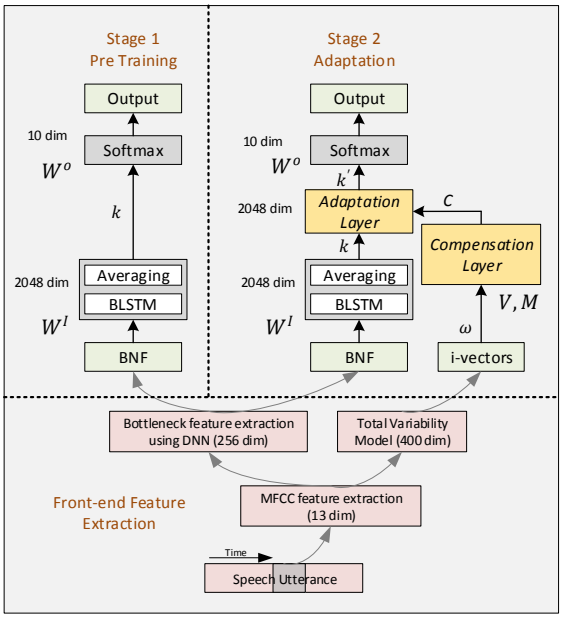

Fig. 4 Language identification framework for use *compensation layer* for adaptation of BLSTM network. The front-end feature extraction of BNF and i-vectors are shown in bottom followed by Stage 1 pre training and Stage 2 adaptation process.

sampled at 16kHz. Since all the utterances used for system training and testing are short duration utterances, voice activity detection was not employed in the frontend feature extraction process.

A. *Bottleneck and i-vector feature extraction*

BNF features were extracted to train the BLSTM model in Stage 1. The BNF extraction is based on a time-delay neural network (TDNN) [27] phonotactic model that was trained on the THCHS30 database. The 40-dimensional raw Mel-filter bank coefficients with a symmetric 4-frame window are given to TDNN as input features. The TDNN has 6 hidden layers and uses a p-norm activation function. The TDNN layers are set to be 2048 neurons except for the last hidden layer which only contains 256 units.

In Stage 2, the low variability feature space was derived using i-vectors [7] and the covariance statistics were learned during the joint nonlinear adaptation learning scheme (Section II A) for these i-vector elements. The universal background model containing 2048 Gaussians is trained with 13-dimensional Mel-frequency cepstral coefficients (MFCC) features using 25ms window and 10ms frame shift. The extracted i-vectors contain 400 dimensions.

B. *Model pre training and adaptation*

As described in Section II, the backend is a simple BLSTM network with a single hidden layer followed by global average pooling and a classification softmax layer. The BLSTM layer contains 1024 neurons for each forward and backward layer with the total of 2048 while the softmax layer has only 10 (the number of languages). A global average pooling layer averages frame level features to utterance level features as in equation (4). Training is carried out for '1s' duration utterances with truncated backpropagation through time.

The initial model in Stage 1 (shown in Fig. 4) is first trained with bottleneck features (BNF) that are extracted as explained in Section III A. In the adaptation process in Stage 2 the second order factorization matrix of V and subspace matrix M is learned after fixing the initial model weight parameters using extracted i-vectors. Finally, classification is conducted for the adapted system using both BNF and i-vectors for the test utterances in the testing phase.

V. Feature Analysis

A. *Comparison of 'matched' and 'mismatched' feature spaces*

First, the feature discriminability of 'matched' and 'mismatched' languages was investigated using **t**-distributed stochastic neighbor embedding (t-SNE) [28] scatter plots (Fig. 5). The t-SNE is a data visualization technique particularly well suited for dimensionality reduction of higher dimension embedding data. This method maps similar and dissimilar points in a higher dimensional space to near and distant points respectively in feature visualizing space (two-dimensional in this case). In this paper, t-SNE scatter plots have been used to illustrate the feature space of the 'Russian' language (a 'mismatch' language) for training and testing data. The 2048-

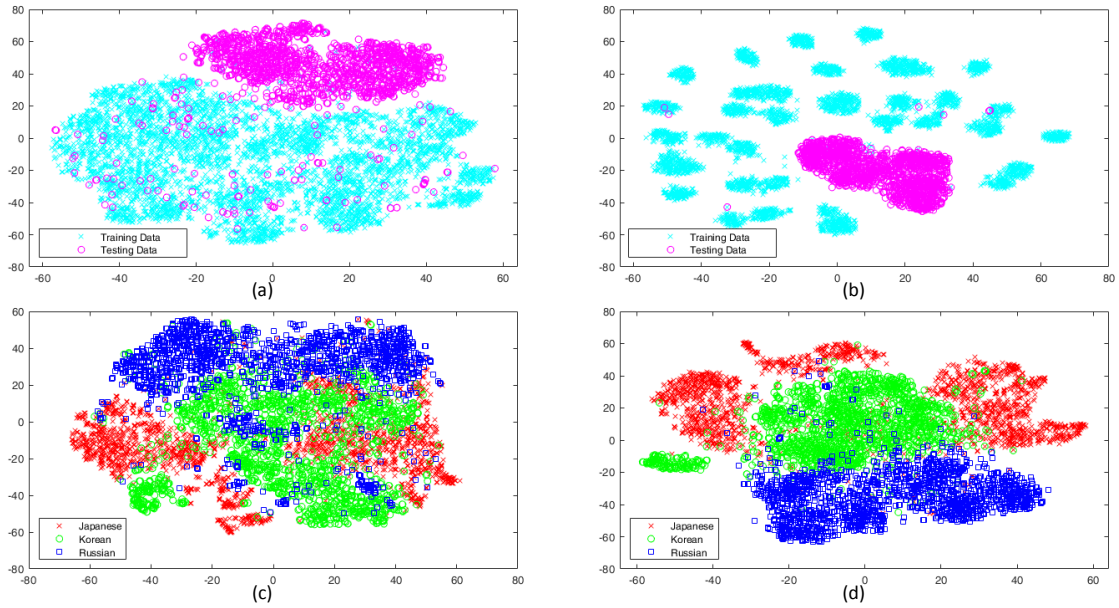

Fig. 5 t-SNE scatter plots of features from mismatch languages before and after adaptation. (a) and (b) compare the training and testing data from the Russian language for 1s duration utterances, before and after adaptation. (c) and (d) show the test features for all three mismatch languages (Japanese, Russian, Korean) for '1s' duration utterances, before and after adaptation.

Table 1. Performance of the proposed system compared to a BLSTM system for AP17-OLR '1s' duration for matched and mismatched conditions.

Condition		Cavg [%]		Improvement [%]
		BLSTM	Proposed	
1	Matched	9.23	7.33	20.59
2	Mismatched	14.97	9.81	**34.47**
Overall		12.14	**8.75**	27.92

dimensional feature vectors $\boldsymbol{k}$ and $\boldsymbol{k}'$ from equations (4) and (5) (before and after the adaptation respectively) were extracted from the adapted model. The effectiveness of the mismatch adaptation process can be visualised in Fig. 5(a) and Fig. 5(b). The adaptation makes the training and testing data more similar where the mismatch is comparatively reduced, i.e. for the Russian language, Fig. 5(b) testing data is surrounded by training data and this may leads to a better classification compared to Fig. 5(a). Further, it is interesting to see that the training data (cyan) is more tightly clustered in the adapted feature space in Fig. 5(b), which is believed to be due to utterance level information in the i-vectors that can relate specifically to individual speakers. In particular, there are 24 speakers who speaks Russian language and we can see there are 24 separate clusters. To validate the observations, the J-measure is obtained [4]. The J-measure is the ratio between inter-class scatter to intra-class scatter, and the larger the value of the J-measure, the higher the mismatch in the feature space. In this instance, the training and testing data are considered as two separate classes.

The resulting J-measure values of 0.92 and 0.91 before and after the adaptation for the Russian language demonstrate that there is a lower mismatch in the adapted feature space compared to the original BLSTM feature output. Likewise, the J-measure was calculated for the other languages individually. These results are not presented here, though it is worth highlighting that, while all the languages showed an improvement in J-measure, the highest improvements were gained in Japanese, Korean and Russian languages. This suggests that the adaptation of the *compensation layer* is more effective when there is a channel mismatch between training and testing data.

Table 1 gives the performance comparison for a standard BLSTM system and the proposed system using a *compensation layer* for the adaptation It is clear that the proposed system has significant relative improvement of around 35% for '1s' duration utterances. Additionally, this improvement is highly significant for 'mismatched' languages compared to 'matched' languages.

B. *Feature comparison of BLSTM output before and after mismatch compensation*

For this study, we aim to illustrate the language discrimination capabilities of the '1s' duration testing data for

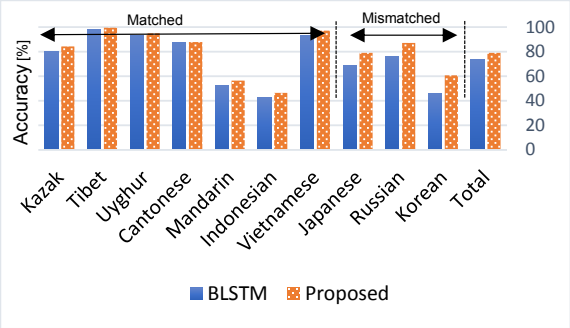

Fig. 6 System performance comparison of BLSTM and proposed systems for AP17-OLR '1s' condition in terms of accuracy for each language.

'mismatch' languages before and after the adaptation process. Similar to the above analysis the t-SNE scatter plots of feature vectors before ($\boldsymbol{k}$) and after ($\boldsymbol{k}'$) the adaptation are analyzed. Fig. 5(c) and 5(d) shows the effectiveness of language separability between the testing data language classes before and after adaptation. Further, this shows that there is a significant improvement in separability in the adapted space compared to unadapted. To validate these observations, similar to above task the J-measure was computed for each feature type. In this case, it can be argued that the larger the value of J-measure, the higher the separability between languages. The calculated J-measure is highest for the adapted feature space with an improvement of 1.69 to 1.71. Fig. 6 shows the individual performance for each language in terms of identification accuracy for the BLSTM and proposed systems. The mismatched languages again show the trend of having the higher improvements compared to the matched languages.

VI. Language Identification Evaluations

For all language evaluation tasks two evaluation metrics of average cost (Cavg) and equal error rate (EER) were used, as originally proposed in AP17-OLR database evaluation plan [27]. First, we explore the behavior of the *compensation layer* with the second order low-rank factorization model and find an appropriate choice of rank r (Section III B), which is required for the adaptation process. The full rank second order model contains $(400 \times 401)/2 \times 2048 \sim 164\text{M}$ parameters. Consequently, these experiments are hard to be run because of computational cost associated with this large number of parameters. In the low-rank second order factorization experiments we replace the above matrix with two matrices, one of size $(400 \times 401)/2 \times r$ and other of size $r \times 2048$ where $r \leq 100$ attaining significantly lower number of parameters than before. Table 2 shows the Cavg and EER for different choices of rank r, and the percentage reduction in parameters compared to a full rank second order model.

The trend is similar in the '1s' and '3s' duration utterances. Based on these observations r was chosen to be 20 for later experiments. Furthermore, as previously mentioned it can be seen that the system performance degrades with increasing number of parameters. The reason for this may be that it introduces highly correlated parameters to the matrix V, due to the inclusion of the outer product of the input feature vector $\boldsymbol{\omega}$. It should be noted that there are only 1.64M parameters in this low rank representation, a reduction of 99.0%. To explore the behavior at this operating point further, symmetric and diagonal constraints were also applied to the V matrix, based on the idea that the weights relevant to the outer product of $\boldsymbol{\omega}$ tend to be symmetric in theory. The diagonal constraint was an extension of the symmetric and further reduced the number of unique parameters to train the second order factorized model. However, these constraints did not perform well in our evaluations as shown at the bottom of Table 2.

Finally, system evaluations were carried out for the BLSTM baseline and the proposed technique of *compensation layer* based adaptation including comparisons with some existing systems in the literature. Although many adaptation techniques have been proposed for many different applications, the goal of this paper was to propose the use of the second order factorization model, or the *compensation layer*, for mismatch compensation in a short duration language identification task. In consideration of this aim, and since adaptation is novel to language identification tasks, results for a LSTM based language identification system proposed in [27] are included for comparison, along with the baseline BLSTM approach described in Section II A. The benefit of factorised hidden variability subspace (FHVS) adaptation for a BLSTM layer was shown in [11], and as such this is also included as one of the comparisons. Table 3 shows that the proposed second order factorized adaptation technique is able to outperform all other systems with 28% relative improvement in terms of Cavg, and 22% relative reduction in EER compared to the baseline BLSTM system,

Table 2. Performance for '1s' and '3s' duration utterances with different choices of rank r on the second order model in the *compensation layer*, and the number of parameters and % reduction compared to a full rank V matrix.

r = rank	Performance [%]				# of Parameters (% Reduction)
	1s		3s		
	Cavg	EER	Cavg	EER	
Full Rank	~	~	~	~	164M
r = 9	8.88	8.77	3.41	3.40	0.74M (99.6)
r = 10	8.94	8.84	3.49	3.47	0.82M (99.5)
r = 20	**8.75**	**8.46**	**3.15**	**3.32**	**1.64M (99.0)**
r = 50	9.65	9.66	3.78	3.94	4.11M (97.5)
r = 100	10.2	9.63	5.68	5.35	8.22M (95.0)
r = 20, Symmetric	9.80	9.33	3.78	3.76	1.64M (99.0)
r = 20, Diagonal	9.68	9.25	3.61	3.65	1.64M (99.0)

Table 3. Performance of the proposed system compared to the baseline for AP17-OLR '1s', '3s' and 'all' duration conditions.

System	Performance [%]					
	1s		3s		all	
	Cavg	EER	Cavg	EER	Cavg	EER
LSTM [26]	11.5	11.8	7.27	8.24	6.89	8.15
BLSTM	12.1	10.8	6.68	6.12	5.89	5.24
FHVS [10]	8.82	8.47	3.77	3.79	3.30	3.19
Proposed	**8.75**	**8.46**	**3.15**	**3.32**	**2.36**	**2.48**

confirming the effectiveness of proposed technique. Further, the proposed joint nonlinear adaptation learning scheme of second order information shows better performance compared to linear factorization based methods, such as FHVS proposed in [11]. This emphasises the significance of capturing the second order information in the mismatch conditions. Finally, the results in Table 3 show similar performance gains across the different utterance durations, showing that the proposed technique is generalizable.

VII. Conclusions

In this paper, we have proposed a *compensation layer* for mismatch adaptation. The significance of this BLSTM adaptation using second order information is studied using the analysis of the feature space and language identification experiments. When second order information of the input vectors is integrated into the adaptation *compensation layer*, it introduces a large number of parameters. The low-rank factorization method was found to be effective in reducing the number of parameters and obtaining an abstract representation of the information for the purposes of adaptation. The joint nonlinear adaptation learning scheme of the *compensation layer* showed promising results compared to some existing linear factorization techniques for adaptation. The proposed *compensation layer* introduced utterance dependant parameters using i-vectors and connected these to the BLSTM layer as a new set of adaptively trained weights. The proposed technique was evaluated with the AP-OLR17 database, which is designed for the short duration language identification task. For all test data durations of '1s', '3s' and 'all', the proposed compensation layer was able to achieve the superior performance compared to the baseline BLSTM system.

References

[1] F. Castaldo, D. Colibro, E. Dalmasso, P. Laface, and C. Vair, "Compensation of nuisance factors for speaker and language recognition," *IEEE Transactions on Audio, Speech, and Language Processing,* vol. 15, no. 7, pp. 1969-1978, 2007.

[2] R. Travadi, M. Van Segbroeck, and S. S. Narayanan, "Modified-prior i-vector estimation for language identification of short duration utterances," in *INTERSPEECH*, 2014, pp. 3037-3041.

[3] M.-G. Wang, Y. Song, B. Jiang, L.-R. Dai, and I. McLoughlin, "Exemplar based language recognition method for short-duration speech segments," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013, pp. 7354-7358: IEEE.
[4] S. Fernando, V. Sethu, E. Ambikairajah, and J. Epps, "Bidirectional Modelling for Short Duration Language Identification," in *INTERSPEECH*, 2017, pp. 2809-2813.
[5] S. Fernando, V. Sethu, and E. Ambikairajah, "Eigenfeatures: An alternative to Shifted Delta Coefficients for Language Identification," presented at the SST2016, Parramatta, Australia, 06 - 09 December 2016, 2016.
[6] S. Fernando, V. Sethu, and E. Ambikairajah, "A Feature Normalisation Technique for PLLR Based Language Identification Systems," in *INTERSPEECH*, 2016, pp. 2925-2929.
[7] N. Dehak, P. A. Torres-Carrasquillo, D. A. Reynolds, and R. Dehak, "Language Recognition via i-vectors and Dimensionality Reduction," in *INTERSPEECH*, 2011, pp. 857-860.
[8] Y. Lei, L. Burget, L. Ferrer, M. Graciarena, and N. Scheffer, "Towards noise-robust speaker recognition using probabilistic linear discriminant analysis," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, 2012, pp. 4253-4256: IEEE.
[9] S. Fernando, V. Sethu, and E. Ambikairajah, "Sub-band Envelope Features Using Frequency Domain Linear Prediction for Short Duration Language Identification," in *INTERSPEECH*, 2018, pp. 1818-1822.
[10] S. Fernando, V. Sethu, and E. Ambikairajah, "Hidden variability subspace learning for adaptation of deep neural networks," *Electronics Letters*, vol. 54, no. 3, pp. 173-175Available: http://digital-library.theiet.org/content/journals/10.1049/el.2017.4027
[11] S. Fernando, V. Sethu, and E. Ambikairajah, "Factorized Hidden Variability Learning for Adaptation of Short Duration Language Identification Models," presented at the ICASSP, Calgary, Alberta, Canada, 15-20 April, 2018.
[12] L. Samarakoon and K. C. Sim, "Learning factorized feature transforms for speaker normalization," in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, 2015, pp. 145-152: IEEE.
[13] L. Samarakoon and K. C. Sim, "On combining i-vectors and discriminative adaptation methods for unsupervised speaker normalization in dnn acoustic models," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, 2016, pp. 5275-5279: IEEE.
[14] H. Zen *et al.*, "Statistical parametric speech synthesis based on speaker and language factorization," *IEEE transactions on audio, speech, and language processing,* vol. 20, no. 6, pp. 1713-1724, 2012.
[15] B. Li and K. C. Sim, "Comparison of discriminative input and output transformations for speaker adaptation in the hybrid NN/HMM systems," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.
[16] F. Richardson, D. Reynolds, and N. Dehak, "A Unified Deep Neural Network for Speaker and Language Recognition," *arXiv preprint arXiv:1504.00923,* 2015.
[17] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*, 2011, pp. 24-29: IEEE.
[18] J. Xue, J. Li, D. Yu, M. Seltzer, and Y. Gong, "Singular value decomposition based low-footprint speaker adaptation and personalization for deep neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 6359-6363: IEEE.
[19] T. Tan, Y. Qian, M. Yin, Y. Zhuang, and K. Yu, "Cluster adaptive training for deep neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, 2015, pp. 4325-4329: IEEE.
[20] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, "KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013, pp. 7893-7897: IEEE.
[21] A. Zeyer, P. Doetsch, P. Voigtlaender, R. Schlüter, and H. Ney, "A comprehensive study of deep bidirectional LSTM RNNs for acoustic modeling in speech recognition," *arXiv preprint arXiv:1606.06871,* 2016.
[22] K. Yu and M. Salzmann, "Second-order convolutional neural networks," *arXiv preprint arXiv:1703.06817,* 2017.
[23] C. L. Giles and T. Maxwell, "Learning, invariance, and generalization in high-order neural networks," *Applied optics,* vol. 26, no. 23, pp. 4972-4978, 1987.
[24] K. Cremanns and D. Roos, "Deep Gaussian Covariance Network," *arXiv preprint arXiv:1710.06202,* 2017.
[25] X. Xu, N. Mu, X. Zhang, and B. Li, "Covariance descriptor based convolution neural network for saliency computation in low contrast images," in *Neural Networks (IJCNN), 2016 International Joint Conference on*, 2016, pp. 616-623: IEEE.
[26] G. Strang, *Introduction to Linear Algebra*. Wellesley Cambridge Press, 2009.
[27] Zhiyuan Tang, Dong Wang, Yixiang Chen, and Q. Chen, "AP17-OLR Challenge: Data, Plan, and Baseline," *arXiv:1706.09742,* 2017.
[28] L. v. d. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research,* vol. 9, no. Nov, pp. 2579-2605, 2008.