

Speaker-Adapted Confidence Measures for ASR using Deep Bidirectional Recurrent Neural Networks

Miguel Ángel Del-Agua, Adrià Giménez, Albert Sanchis, Jorge Civera, and Alfons Juan

Abstract—In the last years, Deep Bidirectional Recurrent Neural Networks (DBRNN) and DBRNN with Long Short-Term Memory cells (DBLSTM) have outperformed the most accurate classifiers for confidence estimation in automatic speech recognition. At the same time, we have recently shown that speaker adaptation of confidence measures using DBLSTM yields significant improvements over non-adapted confidence measures. In accordance with these two recent contributions to the state of the art in confidence estimation, this paper presents a comprehensive study of speaker-adapted confidence measures using DBRNN and DBLSTM models. Firstly, we present new empirical evidences of the superiority of RNN-based confidence classifiers evaluated over a large speech corpus consisting of the English LibriSpeech and the Spanish poliMedia tasks. Secondly, we show new results on speaker-adapted confidence measures considering a multi-task framework in which RNN-based confidence classifiers trained with LibriSpeech are adapted to speakers of the TED-LIUM corpus. These experiments confirm that speaker-adapted confidence measures outperform their non-adapted counterparts. Lastly, we describe an unsupervised adaptation method of the acoustic DBLSTM model based on confidence measures which results in better automatic speech recognition performance.

Index Terms—Automatic speech recognition, confidence estimation, confidence measures, deep bidirectional recurrent neural networks, long short-term memory, speaker adaptation.

I. INTRODUCTION

CONFIDENCE Estimation (CE) aims at providing *Confidence Measures* (CM) of the Automatic Speech Recognition (ASR) output at a certain level of granularity such as sub-word, word or utterance [1]. CM are represented by scores usually between 0 and 1 which reflect the reliability of any recognition output. Considering CM as probabilities of correctness, CE has been largely addressed as a two-class (correct or incorrect) pattern recognition problem [1], [2], [3], [4], [5]. To this effect, a binary classifier is trained to map input features to class posterior probabilities. Under this approach, CE has been gradually improved by exploring novel features

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 761758 (X5gon) and Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 287755 (transLectures) and ICT Policy Support Programme (ICT PSP/2007-2013) as part of the Competitiveness and Innovation Framework Programme (CIP) under grant agreement no. 621030 (EMMA), as well as the Spanish government's TIN2015-68326-R (MINECO/FEDER) research project MORE.

The authors are with the Departament de Sistemes Informàtics i Computació, Universitat Politècnica de València, E-46022 València, Spain (e-mail: mideag@upvnet.upv.es; agimenez@dsic.upv.es; jcivera@dsic.upv.es; josanna@dsic.upv.es; ajuan@dsic.upv.es).

and by designing more and more accurate classifiers [1], [2], [3], [4], [5].

Recent significant improvements to word-level CE have come from the use of Recurrent Neural Networks (RNN). Other classifiers that were considered until recently to be very effective, such as Conditional Random Fields (CRF), Logistic Regression (LR) or naïve Bayes (NB), have been clearly outperformed by RNN [5], [6], [7]. In particular, both deep bidirectional RNN (DBRNN) and DBRNN with long short-term memory units (DBLSTM) have shown their superiority when compared with non-NN-based classifiers or even with deep feedforward NN (DNN) [5], [7].

At the same time, adaptation of CM has shown to be very effective in improving baseline performance [4], [7], [8], [9]. This is a key point, from our point of view, especially for tasks with limited training data, since adaptation allows us to easily obtain accurate task-specific models from generic models trained on large, non-specific data sets. Moreover, there is an increasing number of interesting applications in which relevant information for adaptation is available, such as speaker identity in video lecture repositories. However, to the best of our knowledge, thus far there have been very few contributions in adaptation of CM. To address this, we were the first to implement speaker adaptation for CM. In [4], we evaluated speaker-dependent features into an LR model with good results. Then, in a follow-up work [7], we obtained even better preliminary results by using speaker-adapted DBLSTM.

In this paper, following our previous work [7], new technical contributions are reported, including a new architecture for CE in which word embeddings and CE models are jointly trained, and also a novel CE-based unsupervised adaptation method for acoustic BLSTMs. Furthermore, a multi-task empirical evaluation setting is applied to achieve solid empirical results which confirm our previous preliminary results for a single task.

The content of this paper is organized as follows: a brief review of recent work in CE is given in Section II; the proposed speaker-adapted RNN architecture for CE is presented in Section III; empirical results are reported in Section IV; finally, the main conclusions of this work are summarized in Section V.

II. RECENT WORK IN CONFIDENCE ESTIMATION

CE has been largely addressed following three main approaches [1]. One of them, known as *Utterance Verification* (UV), formulates CE as a statistical hypothesis testing

problem [10]. The second one is based on word posterior probabilities computed over N -best lists, word lattices or confusion networks [11], [12]. The third approach considers CE as a two-class classification problem in which class posterior probabilities are estimated combining predictor features [2], [3], [4], [5]. The second approach is currently in wide use, since CM are computed in a straightforward manner from the ASR output. However, significantly better performances are generally reached using the third, classifier-based approach, mostly if word posteriors are used as input features [2], [3], [5].

In recent years, the classifier-based approach has directly benefited from the use of deep learning models outperforming the most accurate earlier classifiers such as CRF [5], [6], [7]. In a first proposal, DNN and kernel deep convex networks (K-DCN) were applied at the utterance level to discriminate between in-grammar and out-of-grammar utterances [13]. In later research, RNN have demonstrated outstanding performance in word-level CE [5], [6], [7], [14]. In particular, DBRNN and DBLSTM have confirmed their superiority over other classifiers such as CRF, DNN, DRNN and DLSTM [5], [7].

The performance of CM can be further improved by means of adaptation techniques [4], [7], [8], [9]. Significant performance gains have been reported by adapting generic CM using a small amount of transcribed adaptation data in a post-processing step called confidence calibration, based on different models such as maximum entropy, NN and deep belief networks [8]. Normalization of CM using adaptation data has also been proposed via confidence mapping to avoid decision threshold reselection when acoustic models are updated [9]. Very recently, we proposed speaker adaptation of LR models and DBLSTM for CE, showing that speaker-adapted models outperform their non-adapted counterparts [4], [7]. For instance, speaker-adapted DBLSTM produced relative reductions in *Classification Error Rate* (CER) of 4.6% when compared with non-adapted DBLSTM.

III. SPEAKER-ADAPTED CONFIDENCE MEASURES USING DEEP BIDIRECTIONAL RECURRENT NEURAL NETWORKS

RNN have proven to be extremely successful in many related fields of speech processing, e.g, acoustic and language modelling, speech synthesis or spoken language understanding [15], [16], [17], [18]. RNN features recurrent connections which enable efficient modelling of temporal dependencies, outperforming other models without this capability. The most basic form of RNN was gradually improved to deal with some limitations such as the vanishing gradient problem and the use of context information in only one time direction [19], [20]. With regard to the former limitation, the LSTM architecture was proposed to overcome the vanishing gradient problem by which long-term dependencies make difficult the training of RNN [21]. Basically, LSTM differ from RNN in the use of hidden layers composed of built-in memory cells which are able to store information for long periods of time. As to the latter limitation, both past and future time directions were incorporated by extending RNN to BRNN [20]. In BRNN, hidden layers are composed of two separate forward and

backward layers which are responsible for the positive and negative time directions, respectively. It is worth mentioning that BRNN with hidden layers composed of LSTM cells result in the BLSTM architecture [22]. In general, better performance can be expected from deep architectures stacking multiple BRNN or BLSTM hidden layers [23]. In this section we describe our CE model based on deep BRNN and BLSTM architectures, and the speaker adaptation process.

The architecture of the proposed CE model is depicted in Fig.1. For simplicity, we show an architecture based on two bidirectional recurrent hidden layers. Both the DBRNN and DBLSTM architectures are represented in this single figure, since the only difference between them is the type of recurrent cell used in the hidden layers. The input layer is composed of a set of R word-level predictor features along with a word embedding representation. Predictor features are typically computed from the speech decoding, word-lattices and from the ASR models (the features used in this work are described in Sec. IV-B).

Word embeddings are also fed into the first hidden layer, since word identities have shown to be very useful in improving CE [2], [4], [5], [13], [14], [24], [25]. To this end, we have not used a conventional one-hot encoding, as this would make the number of parameters grow linearly with the vocabulary size V . Instead, we have used a more compact representation where each word is mapped to a real word vector of a fixed dimension F [26]. In the case of NN, this word representation is learned by adding an extra layer to the NN which takes as input the one-hot representation and outputs a fixed-length vector. This means learning a projection matrix of size $V \times F$, in which the i th row corresponds to the embedding representation of the i th word in the vocabulary. In this way, words with similar behaviour can be expected to be represented by similar word embeddings. The vocabulary is typically restricted to the most frequent words. In this way, an embedding representation for unknown words is learned by labelling low-frequency words as unknown. This parameter matrix is trained jointly with the rest of the neural network parameters.

Given a sequence of N input vectors $\mathcal{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$ representing N recognized words W_1^N , where each vector is composed of the R word-level predictor features along with the word embedding representation, the network produces a sequence of N output vectors $\mathcal{Y} = (\mathbf{y}_1, \dots, \mathbf{y}_N)$ defining a probability distribution over each class $c = \{\text{incorrect}(0), \text{correct}(1)\}$. These probabilities correspond to the network's estimation of observing each class c at word n given \mathcal{X} .

The network is trained to minimize the cross-entropy error of the targets using a softmax output layer with 2 output units that represent the two-category class based on the standard back-propagation through time algorithm (BPTT) [27]. Given a target sequence $\mathcal{Z} = (\mathbf{z}_1, \dots, \mathbf{z}_N)$, the network minimizes the negative log-probability of the target sequence given the input sequence:

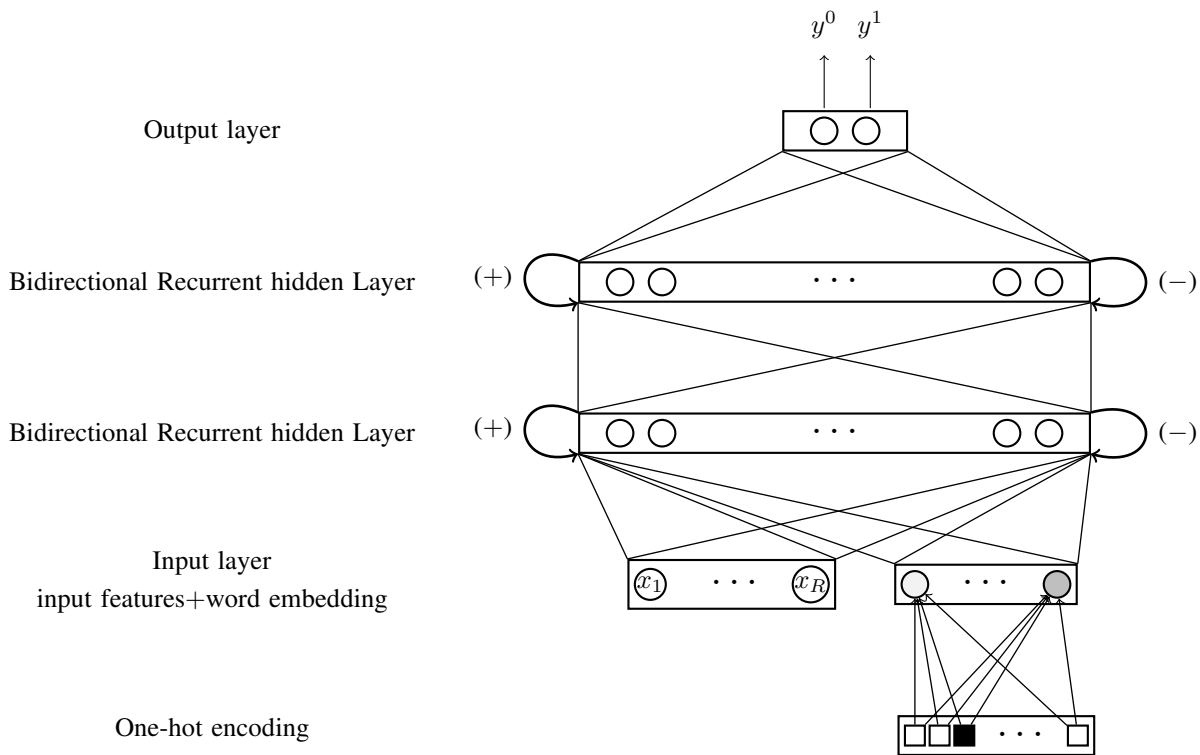


Fig. 1. DBRNN and DBLSTM architectures of two hidden layers for CE. Positive (forward states) and negative (backward states) time directions are indicated by (+) and (-), respectively.

$$-\log P(\mathcal{Z}|\mathcal{X}) = -\sum_{n=1}^N \log p(c = z_n | \mathbf{x}_n) = -\sum_{n=1}^N \log y_n^{z_n} \quad (1)$$

where $y_n^{z_n}$ is the probability estimated at word n by the output neuron that represents the target class z_n .

Once the network has been estimated based on Eq. (1), M new training pairs $\{\mathcal{X}, \mathcal{Z}\}_1^M$ from one speaker are used for adaptation. Adaptation is performed by following a conservative training strategy in which a very small learning rate and early stopping are used [28]. Note that this strategy has become a conventional method for regularization in deep learning because of its effectiveness and simplicity. To actually adapt models, the M given training pairs are split into an adaptation set and a validation set. Adaptation data is used to update the speaker-independent network (or part of it), whereas validation data is used to set the error of the resulting speaker-adapted network. The adaptation process finishes when the validation error stops changing significantly. Then, the final speaker-adapted network is trained from all training pairs by running an “optimal” number of epochs, as determined by the early stopping procedure.

IV. EXPERIMENTS

A. Experimental Setup

The experimental study was conducted over several speech tasks involving the English and Spanish languages. Accordingly, a state-of-the-art ASR system was trained for each language using the transLectures-UPV toolkit (TLK) [7], [29],

[30], [31]. TLK is an open-source ASR toolkit developed at the Universitat Politècnica de València (UPV) by the MLLP research group within the framework of the EU-funded project transLectures¹. It comprises a set of tools for audio processing, feature extraction, HMM and DNN training and decoding. Its main features include multilingual and convolutional NNs, DNN sequence discriminative training based on Maximum Mutual Information (MMI), and different DNN speaker adaptation techniques such as output-feature discriminant linear regression (oDLR) [32] or Kullback-Leibler Divergence based techniques [33]. TLK has shown to provide competitive results in challenging and well-known tasks such as TED-LIUM, LibriSpeech, IWSLT or CHiME [7], [29], [30], [31].

The English ASR system was trained using the LibriSpeech training dataset, which contains almost 1000 hours of read speech recordings from the LibriVox project’s audio books [34] (statistics in Table I). On the other hand, the Spanish ASR system was trained using the poliMedia speech corpus enlarged to about 800 hours for training [35]. PoliMedia is a high-quality multimedia educational repository developed by the UPV. It includes more than 15,000 Spanish video lectures lasting up to 10 minutes each, created by more than 1800 lecturers, summing up a total amount of about 3000 hours. This speech corpus was developed within the EU-funded project transLectures (statistics in Table II).

The audio data was preprocessed with a Hamming window of 25 ms shifted at 10 ms intervals into 16 Mel-frequency cepstral coefficients (MFCC) plus deltas and accelerations,

¹<https://www.translectures.eu/web/>

TABLE I
STATISTICS OF THE LIBRISPEECH CORPUS.

Set	Duration (h)	Speakers	Words	Vocab	WER
Train	961	1210	9.4M	89K	4.7
Dev-other	5.3	33	51K	7.4K	12.5
Test-other	5.1	33	52K	7.6K	13.5

resulting into 48-dimensional feature vectors. Speaker-adapted features were then obtained by means of Cepstral Mean and Variance Normalization (CMVN) and applying a Constrained Maximum Likelihood Linear Regression transform following the simple target model approach (fMLLR) [36].

The acoustic models were based on hybrid models [37], [38], [39]. For hybrid training, forced alignments of the senone (tied-state) transcriptions to the acoustic features (MFCC and fMLLR) were obtained by training conventional context-dependent Gaussian mixture model hidden Markov models (CD-GMM-HMMs). CD-GMM-HMMs consist of three left-to-right tied-states estimated following a phonetic decision tree approach [40]. The resulting number of tied-states was 8.3K and 10K, reaching up to a total amount of 256K and 478K Gaussians for English and Spanish, respectively.

These baseline alignments were then used to train both speaker-independent and speaker-adapted CD-DNN-HMMs [38] for each language with a context window of 11 frames, 7 hidden layers with ReLU activation functions and 2048 units each. The trained speaker-adapted CD-DNN-HMMs were then used to further improve the state alignments. Using these DNN state realignments, we finally trained a speaker-adapted DBLSTM-HMM [39] for each language using the open source toolkit TensorFlow [41]. In both cases, the DBLSTM network had 5 bidirectional hidden layers with 1200 LSTM cells per layer, resulting in a total of 33.3M and 36.3M weights for English and Spanish, respectively. Relative improvements in WER of about 4.6% and 5.8% over the LibriSpeech and poliMedia test sets were achieved using DBLSTM-HMMs compared to CD-DNN-HMMs.

For the English language model (LM), we used the freely available pre-built 4-gram model released as part of the LibriSpeech corpus [34]. As for Spanish, we used the 4-gram LM built by UPV within the transLectures project [42], [43]. Both models had a vocabulary size of about 200K words, and the test set perplexities were 146 and 205, respectively.

Speech processing was carried out following a two-pass decoding setup. The speaker-independent CD-DNN-HMM ASR system was used primarily to obtain a transcription which in conjunction with a simple “target” HMM allowed for the transformation of acoustic features into speaker-adapted features. A word-lattice was then generated feeding the speaker-adapted features into the hybrid DBLSTM-HMM ASR system. Both recognition steps were carried out using a pruned version of the LMs to allow for very fast decoding. The final transcription was produced by rescoring the word-lattice with the whole LM.

TABLE II
STATISTICS OF THE POLIMEDIA SPEECH CORPUS.

Set	Duration (h)	Videos	Speakers	Words	Vocab	WER
Train	813	9.5K	205*	8.3M	36.6K	14.5
Dev	3.4	26	5	35K	2.6K	11.3
Test	3.2	23	5	30K	2.4K	12.5

(*) Lower estimate, since training set is not wholly speaker-annotated.

B. Word-level predictor features

A number of $R = 20$ common word-level predictor features have been used in this work. These features have been computed from the speech decoding, word-lattices and from the ASR models. We briefly enumerate them here:

- (i) Features based on speech decoding and ASR models:
 1. Decoding score: Word score produced jointly by the acoustic and language models during decoding.
 2. Acoustic log-score: As in 1, but considering only the acoustic model.
 3. Normalized acoustic log-score: As in 2, but normalized per time frame (10 ms).
 4. Duration: Word length in ms.
 5. Language model probability: N-gram language model probability for the decoded word.
 6. Length of the N-gram in which the word was decoded.
 7. Average number of alternative hypothesis within the decoding word boundaries.
 8. Binary feature, equals 1 if the word appears in both the first and second decoding hypotheses.
- (ii) Features based on word-lattices:
 - 9-11. Forward, backward and edge posterior probabilities: The forward-backward algorithm is applied to the word-lattice to compute forward, backward and posterior probabilities for every edge in the lattice. As usual, edges in a word-lattice are associated with words occurring at specific intervals along the time axis; and probabilities are computed from acoustic and language model scores by using the (meta-)parameters set during the decoding phase. It is worth noting that edge posterior probabilities are probability sums of all paths including the given edge (normalized by the probability mass of all paths in the lattice).
 - 12-14. Three variants of word posterior probabilities [11]: More precise word posterior probabilities can be computed by summing up the posterior probabilities of all edges containing the word in approximately the same interval time. Moreover, an appropriate scaling of acoustic model probabilities during the forward-backward algorithm is really needed to prevent (nearly) all posterior probability mass from concentrating in a few word-lattice hypotheses. In this case, given a word w which occurs at a specific point in time $t \in [s, e]$, its *accumulated* posterior probability at time t , $A(w, t)$, is computed by summing the posterior probabilities over all edges intersecting word w at time t . From this, three different variants of word posterior probabilities are computed:

12. Intersection: $P_{sec}(w, [s, e]) = \sum_{t=s}^e A(w, t)$
13. Maximum: $P_{max}(w, [s, e]) = \max_{t \in [s, e]} A(w, t)$
14. Average: $P_{avg}(w, [s, e]) = \frac{1}{e-s+1} \sum_{t=s}^e A(w, t)$

15-17. As in 12-14, but using only acoustic scores during the forward-backward algorithm.

18-20. As in 12-14, but using only language model probabilities during the forward-backward algorithm.

C. Evaluation metrics

We have used three metrics to evaluate CE performance: (i) the area under a ROC curve (AUC), (ii) the classification error rate (CER), and (iii) the normalized cross entropy (NCE). We briefly explain them in this section.

Let us assume that the ASR output results in C correctly recognized words and I misrecognized words. Let *False Rejection* be the number of correctly recognized words with confidence lower than a decision threshold τ ($FRR(\tau)$) and, equivalently, let *True Rejection* be the number of misrecognized words with confidence lower than τ ($TRR(\tau)$). The *False Rejection Rate* ($FRR(\tau)$) and the *True Rejection Rate* ($TRR(\tau)$) for a decision threshold τ are computed as:

$$FRR(\tau) = \frac{FR(\tau)}{C} \quad TRR(\tau) = \frac{TR(\tau)}{I} \quad (2)$$

A *Receiver Operating Characteristic* (ROC) curve represents $TRR(\tau)$ against $FRR(\tau)$ for different values of τ . The AUC provides an adequate overall estimation of the classification accuracy, 100 being a perfect classification and 50 a random classification (diagonal ROC curve).

The *Classification Error Rate* (CER) for a decision threshold τ is computed as:

$$CER(\tau) = \frac{FR(\tau) + (I - TR(\tau))}{C + I} \cdot 100 \quad (3)$$

A *baseline CER* can be computed by classifying all the words as correct (i.e., $\tau = 0$):

$$CER(0) = \frac{I}{C + I} \cdot 100 \quad (4)$$

Clearly, $\tau = 0$ is not necessarily optimal in the sense of minimizing Eq. (3). Therefore, it is convenient to consider the classification threshold $\tau = \tau^*$, which minimizes the CER criterion (usually that which provided the minimum CER in a *development set*):

$$\tau^* = \arg \min_{\tau} CER(\tau) \quad (5)$$

We have also used the *Normalized Cross Entropy* (NCE) as proposed by NIST [44]:

$$NCE = \frac{H_{max} + \sum_{w \in correct} \log(cm(w)) + \sum_{w \in incorrect} \log(1 - cm(w))}{H_{max}} \quad (6)$$

where $cm(w)$ is the CM of word w and $H_{max} = -(p \log p + (1-p) \log(1-p))$, p being the prior probability for a word to be correct. Note that the higher the NCE, the better

the CM performance, with optimal classification being reached when NCE equals one. It is worth mentioning that NCE score is lower unbounded, as the logarithm of low values can occur in samples with high scores on their opposite class.

D. Experiments on CE

We performed experiments on CE using the LibriSpeech and poliMedia speech tasks. For each task, the training data were used to estimate DBRNN and DBLSTM models with TensorFlow following the architecture described in Sec. III. The optimal numbers of hidden layers, neurons per hidden layer and word embedding size were tuned using the development set. The characteristics of the optimal topologies for each task are shown in Table III.

TABLE III
 Characteristics of the optimal DBRNN and DBLSTM topologies for the LibriSpeech and poliMedia speech tasks.

	LibriSpeech		poliMedia	
	DBRNN	DBLSTM	DBRNN	DBLSTM
# hidden layers	3	4	2	2
# neurons per layer	512	512	64	512
Word embedding size	80	20	80	10

Table IV summarizes the results obtained in terms of the different metrics presented in Section IV-C. $CER(\tau^*)$ figures in Table IV correspond to the classification error attained in the test set using a threshold τ^* providing the minimum CER in validation. The performance of the RNN was comparatively evaluated with respect to word posterior probabilities (WP) and CRF [3], [11]. A linear interpolation of the RNN models was also tested aiming to further improve their individual performance (BRNN+BLSTM). The interpolation weights were tuned on the corresponding development set and fixed to 0.5 in the case of LibriSpeech and 0.3 (BRNN) and 0.7 (BLSTM) in the case of poliMedia. The experiments with CRF were carried out using the Wapiti toolkit [45]. The best CRF models were obtained using the training algorithm *rprop*- and modelling dependencies between consecutive words.

TABLE IV
 AUC [%], NCE, CER [%] and 95% Confidence Interval (CI) of CER for the different CM on the LibriSpeech and poliMedia evaluation data. The baseline CERs ($CER(0)$) are 11.99 and 10.90 for LibriSpeech and poliMedia, respectively.

Task	CM	AUC	NCE	$CER(\tau^*)$	95%-CI CER
LibriSpeech	WP	85.3	-0.74	10.71	[10.44, 10.97]
	CRF	89.6	0.36	9.29	[9.04, 9.54]
	BRNN	91.1	0.40	8.82	[8.58, 9.07]
	BLSTM	91.0	0.38	8.85	[8.60, 9.09]
	BRNN+BLSTM	91.5	0.41	8.65	[8.41, 8.89]
poliMedia	WP	83.6	-0.57	9.67	[9.33, 10.00]
	CRF	90.0	0.40	7.69	[7.39, 7.99]
	BRNN	91.6	0.44	7.00	[6.71, 7.29]
	BLSTM	92.0	0.44	6.77	[6.48, 7.05]
	BRNN+BLSTM	92.1	0.45	6.75	[6.47, 7.04]

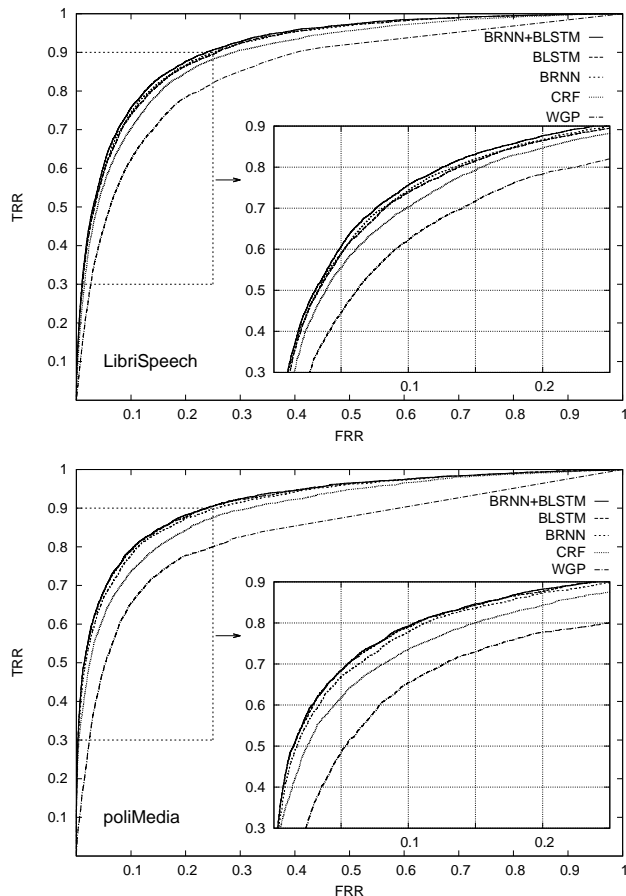


Fig. 2. ROC curves of the different CM for the LibriSpeech (at the top) and poliMedia (at the bottom) evaluation data. TRR is the *True Rejection Rate* and FRR is the *False Rejection Rate*.

From the results in Table IV, it can be stated that RNN models clearly outperform CRF and WP, confirming previous results [5], [7]. Better performance is consistently achieved in all the evaluation measures using RNN models. The improvement in CER of the RNN models over CRF and WP is statistically significant at the 95% confidence level to a great extent, especially in the case of poliMedia. This better overall performance is depicted in Fig. 2, where the ROC curves of the RNN models clearly outperform the CRF and WP models for all decision thresholds τ .

On the other hand, the different RNN models present very similar behaviour, with no statistically significant differences visible between their performance. Even so, better figures are obtained in general using BRNN+BLSTM interpolation. This is depicted in Fig. 2, where small improvements can be observed when the linear interpolation of RNN models is compared to their individual performance.

E. Experiments on speaker-adapted CM

The evaluation of the speaker-adapted CM was conducted considering a practical scenario in which both ASR and confidence models may be used in multiple speech tasks. With this purpose, the ASR models and the BRNN+BLSTM confidence estimator trained with LibriSpeech were used to

obtain the transcriptions and confidence scores of talks of eight speakers chosen from the TED-LIUM corpus [46]. The selection of speakers was made on the basis of having at least 4 talks per speaker, in order to perform a 4-fold cross-validation evaluation and also to cover a reasonable range of error between 10% and 30% of WER. The main characteristics of these talks are summarized per speaker in Table V, where each speaker set is composed of exactly 4 talks.

TABLE V
Global statistics of the 4 talks per speaker extracted from the TED-LIUM corpus.

Speaker	Duration (h:mm:ss)	Running words (k)	WER [%]	CER(0) [%]
1	0:56:26	9.4	21.95	18.61
2	0:40:37	9.0	19.34	16.00
3	0:39:34	7.8	23.56	19.74
4	0:45:59	7.4	22.95	19.03
5	0:45:56	8.6	13.33	12.07
6	0:34:41	8.4	14.90	12.06
7	0:31:55	8.5	25.51	20.19
8	0:33:42	6.1	26.79	22.03
<i>All</i>	5:28:53	65.1	20.78	17.35

As mentioned, speaker adaptation of CM was evaluated following the k -fold cross-validation method [47]. In this way, $k = 4$ experiments were performed per speaker, with the supervised transcriptions of 3 talks being used for adapting the LibriSpeech BRNN+BLSTM CE network, while the remaining talk was used for testing. With this strategy, each talk was used three times for adaptation and only once for testing. Moreover, the non-adapted LibriSpeech BRNN+BLSTM CE network was used also to establish the baseline performance of CM without speaker adaptation.

Comparative results in terms of AUC and CER between non-adapted and adapted CM are shown in Table VI. It is worth noting that the non-adapted model corresponds to the BRNN+BLSTM CE network achieving the best performance on the LibriSpeech corpus in the CE experiments reported above. Also, this network was used to derive a speaker-adapted model as described in Section III. CER figures were obtained using the same decision threshold τ for both non-adapted and adapted experiments. The operative τ^* for each speaker was tuned over the adaptation data.

In general, it can be stated that speaker-adapted CM outperform their non-adapted counterparts for all the speakers. Slightly better performance is achieved in terms of AUC, with the only exception of speaker number 8, for which no differences were found. The overall superiority of adapted CM is observed in Fig. 3, where ROC curves obtained considering all the speakers as a whole are plotted comparatively. Similarly, relative improvements in CER of about 2–8% are produced by using adapted CM, except in the case of speaker number 7, for which CER figures were nearly identical. Overall, considering all the speakers as a whole, the improvement in CER is statistically significant at the 95% confidence level to a great extent, since the confidence intervals are [11.94 – 12.47] and [12.39 – 12.93] for adapted and non-adapted CM, respectively. It is worth noting that improvements notably depend on the speaker. A possible explanation of this phenomenon is that model improvement is very much dependent on the quality and amount of the speaker-dependent adaptation data.

TABLE VI

AUC [%] and CER [%] for the adapted and non-adapted CM per speaker of the TED-LIUM corpus. The baseline CERs and the relative improvements (R.I.) in CER over the non-adapted CM are also shown.

Speaker	AUC		CER(τ^*)			R.I. [%]
	-Adapt	Adapt	CER(0)	-Adapt	Adapt	
1	87.4	88.4	18.61	13.86	13.21	4.7
2	88.4	89.4	16.00	12.23	11.53	5.7
3	88.1	88.3	19.74	14.42	14.16	1.8
4	88.8	89.0	19.03	13.21	12.81	3.0
5	90.4	91.0	12.07	9.03	8.29	8.2
6	89.8	90.2	12.06	9.04	8.79	2.8
7	86.6	87.1	20.19	14.09	14.06	0.2
8	87.4	87.4	22.03	15.87	15.48	2.5
All	88.6	89.1	17.35	12.66	12.21	3.6

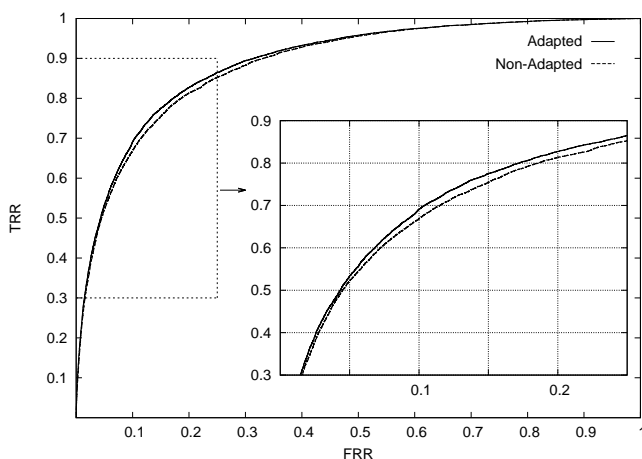


Fig. 3. ROC curves for the adapted and non-adapted LibriSpeech BRNN+BLSTM CE networks using the 8 speakers of the TED-LIUM corpus as a whole.

In practice, it might not be realistic to assume that perfect transcriptions are available for at least three talks. Therefore, one could argue that results in Table VI are optimistic. In order to study the CE performance in a more realistic setting, additional experiments were conducted in which the proposed adaptation approach was tested as a function of the amount of available adaptation data. We used the same 4-fold cross-validation procedure described above, though in this case it was repeated for an increasing percentage of perfect transcriptions available.

Figure 4 shows the CER for each speaker using increasing percentages of available adaptation data (0, 10, 25, 50, 75 and 100). Note that 0% and 100% would correspond with CER results of non-adapted and adapted models, respectively, showed in Table VI. As expected, Figure 4 confirms that the more adaptation data we use, the better CER we achieve. Although this holds visibly for speakers 1, 2, 4, 5 and 6, the results for speakers 3, 7 and 8 do not follow this pattern so clearly. It is worth mentioning that, for almost all the speakers, the CER improves already from the point where we use just 10% of the available adaptation data, and thus we can conclude

that the proposed adaptation approach is really effective even when adaptation data is scarce.

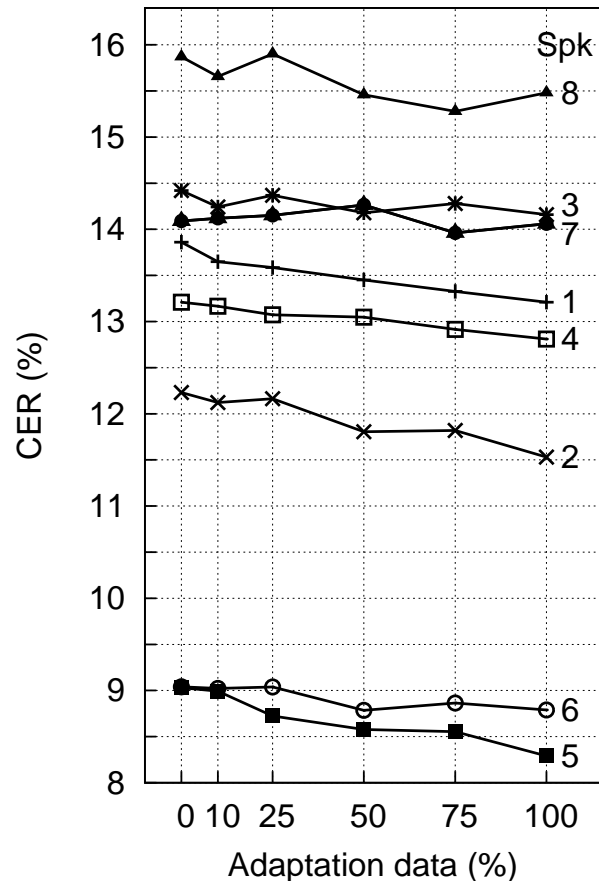


Fig. 4. CER for different percentages of the whole adaptation data.

F. Experiments on improving ASR performance

As mentioned before in Sec. IV-A, we followed a two-pass recognition strategy in which unsupervised speaker adaptation is implemented in a second step based on fMLLR transformed features. Further refinements of the second decoding hypothesis can be produced by means of an additional unsupervised adaptation step. In this extra step, the layers of the acoustic DBLSTM used in the second pass are retrained based on the senone alignments corresponding to the second decoding hypothesis. The retraining is carried out following a conservative training approach using a very small learning rate and early stopping [33]. In particular, given T acoustic vectors of fMLLR features $\mathcal{X} = (\mathbf{x}_1, \dots, \mathbf{x}_T)$ and the senone-level alignments from the second pass hypothesis $\mathcal{S} = (s_1, \dots, s_T)$, the parameters of the acoustic DBLSTM are retrained to maximize the negative cross entropy

$$\mathcal{C}(\mathcal{X}, \mathcal{S}) = -\frac{1}{T} \sum_{t=1}^T \log p(s_t | \mathbf{x}_t) = -\frac{1}{T} \sum_{t=1}^T \log y_t^{s_t} \quad (7)$$

where $y_t^{s_t}$ is an estimation of the probability at frame t given by the output neuron associated with the target class s_t .

TABLE VII
 WER [%] USING DIFFERENT RECOGNITION SETTINGS OVER
 LIBRISPEECH, POLIMEDIA TEST SETS AND THE 8 SPEAKERS OF THE
 TED-LIUM CORPUS.

Recognition setting	LibriSpeech	poliMedia	TED-LIUM
2-pass	13.50	12.53	20.78
3-pass	13.06	12.37	20.02
3-pass+CM	13.05	12.06	19.63

Unsupervised adaptation of the acoustic DBLSTM in the additional pass can benefit from CE by adjusting the influence of the training samples as a function of CM. Formally, we propose to apply a modified cross entropy training criterion for this kind of adaptation. Following this idea, Eq. (7) becomes

$$\mathcal{C}(\mathcal{X}, \mathcal{S}) = -\frac{1}{T} \sum_{t=1}^T \log p(s_t | \mathbf{x}_t) \cdot cm(s_t) = -\frac{1}{T} \sum_{t=1}^T \log y_t^{s_t} \cdot cm(s_t) \quad (8)$$

where $cm(s_t)$ is the word-level CM of senone s_t .

Once the adapted acoustic DBLSTM has been retrained, a third-pass decoding is performed to produce the final hypothesis.

Table VII shows the WER obtained for three different recognition settings on the LibriSpeech and poliMedia test sets and the 8 speakers of the TED-LIUM corpus. The “2-pass” setting corresponds to the baseline performance without performing the third adaptation pass. The “3-pass” setting implies performing the third pass based on Eq. (7). Finally, the “3-pass+CM” setting corresponds to applying Eq. (8) in the third pass.

As we can see, relative reductions in WER of 3.3%, 1.3% and 3.7% are obtained in LibriSpeech, poliMedia and TED-LIUM, respectively, by performing this additional adaptation pass. Moreover, in the case of poliMedia and TED-LIUM, further improvements are achieved by using the proposed third pass based on CM. These improvements reach relative reductions in WER of 2.5% and 2% in poliMedia and TED-LIUM, respectively, with respect to the “3-pass” setting. As a result, relative reductions in WER of 3.3%, 3.8% and 5.5% are achieved in LibriSpeech, poliMedia and TED-LIUM, respectively, by performing this third pass based on CM.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a comprehensive study of speaker adaptation of DBRNN and DBLSTM models for confidence estimation. The study has confirmed the superiority of RNN-based models over the CRF and WP approaches. In particular, a linear interpolation of DBRNN and DBLSTM models has obtained the best performance. Furthermore, we have shown that speaker adaptation of confidence measures is an effective approach for improving confidence estimation. This is an important practical outcome, since general-purpose confidence measures have to be applied frequently in multiple applications and adaptation becomes necessary. As a final contribution, we have proposed a novel unsupervised adaptation of the acoustic DBLSTM based on confidence measures.

Relative reductions in WER in the range of 3% – 5.5% have been achieved in different speech tasks by adding an extra recognition pass of adaptation based on confidence measures into a classical two-pass ASR decoder.

As future work, we plan to apply the same approach to estimate speaker-adapted confidence measures at different levels, such as sub-word or utterance. The idea is to use a bottom-up approach (from sub-word to utterance) where class probabilities generated by lower-level RNN models are used as additional input features by RNN models at higher levels. Moreover, based on previous works [48], [49], we plan to investigate different adaptation approaches in which reestimation of specific parts of the network would be performed depending on the amount of adaptation data.

REFERENCES

- [1] H. Jiang, “Confidence measures for speech recognition: A survey,” *Speech Communication*, vol. 45, no. 4, pp. 455–470, 2005.
- [2] A. Sanchis, A. Juan, and E. Vidal, “A Word-Based Naïve Bayes Classifier for Confidence Estimation in Speech Recognition,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 565–574, 2012.
- [3] M. S. Seigel, “Confidence estimation for automatic speech recognition hypotheses,” Ph.D. dissertation, Department of Engineering, University of Cambridge, 2013.
- [4] I. Sanchez-Cortina, J. Andrés-Ferrer, A. Sanchis, and A. Juan, “Speaker-adapted confidence measures for speech recognition of video lectures,” *Computer Speech & Language*, vol. 37, pp. 11–23, 2016.
- [5] A. Ogawa and T. Hori, “Error detection and accuracy estimation in automatic speech recognition using deep bidirectional recurrent neural networks,” *Speech Communication*, vol. 89, no. 4, pp. 70–83, 2017.
- [6] —, “ASR error detection and recognition rate estimation using deep bidirectional recurrent neural networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4370–4374.
- [7] M. A. del Agua, S. Piqueras, A. Giménez, A. Sanchis, J. Civera, and A. Juan, “ASR confidence estimation with speaker-adapted recurrent neural networks,” in *Interspeech*, 2016, pp. 3464–3468.
- [8] D. Yu, J. Li, and L. Deng, “Calibration of confidence measures in speech recognition,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 8, pp. 2461–2473, 2011.
- [9] K. Kumar, C. Liu, and Y. Gong, “Normalization of ASR confidence classifier scores via confidence mapping,” in *Interspeech*, 2014, pp. 1199–1203.
- [10] E. Lleida and R. C. Rose, “Utterance verification in continuous speech recognition: decoding and training procedures,” *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 2, pp. 126–139, 2000.
- [11] F. Wessel, R. Schlüter, K. Macherey, and H. Ney, “Confidence measures for large vocabulary continuous speech recognition,” *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 3, pp. 288–298, 2001.
- [12] L. Mangu, E. Brill, and A. Stolcke, “Finding consensus in speech recognition: word error minimization and other applications of confusion networks,” *Computer Speech & Language*, vol. 14, no. 4, pp. 373 – 400, 2000.
- [13] P.-S. Huang, K. Kumar, C. Liu, Y. Gong, and L. Deng, “Predicting speech recognition confidence using deep learning with word identity and score features,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7413–7417.
- [14] K. Kalgaonkar, C. Liu, Y. Gong, and K. Yao, “Estimating confidence scores on ASR results using recurrent neural networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4999–5003.
- [15] O. Vinyals, S. V. Ravuri, and D. Povey, “Revisiting recurrent neural networks for robust ASR,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 4085–4088.
- [16] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, “Recurrent neural network based language model,” in *Interspeech*, vol. 2, 2010, pp. 1045–1048.
- [17] Y. Fan, Y. Qian, F.-L. Xie, and F. K. Soong, “TTS synthesis with bidirectional LSTM based recurrent neural networks,” in *Interspeech*, 2014.

- [18] K. Yao, G. Zweig, M.-Y. Hwang, Y. Shi, and D. Yu, "Recurrent neural networks for language understanding," in *Interspeech*, 2013, pp. 2524–2528.
- [19] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.
- [20] M. Schuster and K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
- [21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [22] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional LSTM networks for improved phoneme classification and recognition," in *International Conference on Artificial Neural Networks: Formal Models and Their Applications*, 2005, pp. 799–804.
- [23] 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," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2462–2466.
- [24] S. Ghannay, Y. Esteve, and N. Camelin, "Word embeddings combination and neural networks for robustness in ASR error detection," in *Signal Processing Conference (EUSIPCO)*, 2015, pp. 1671–1675.
- [25] S. Ghannay, Y. Esteve, N. Camelin, and P. Deléglise, "Acoustic word embeddings for ASR error detection," in *Interspeech*, 2016, pp. 1330–1334.
- [26] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, 2013, pp. 3111–3119.
- [27] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cognitive modeling*, vol. 5, no. 3, p. 1, 1988.
- [28] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [29] M. del Agua, A. Giménez, N. Serrano, J. Andrés-Ferrer, J. Civera, A. Sanchis, and A. Juan, "The translectures-UPV toolkit," in *Advances in Speech and Language Technologies for Iberian Languages*, 2014, pp. 269–278.
- [30] M. A. del Agua, A. Martínez-Villaronga, A. Giménez, A. Sanchis, J. Civera, and A. Juan, "The MLLP system for the 4th CHiME challenge," in *Proc. of the 4th CHiME Speech Separation and Recognition Challenge (CHiME-4)*, San Francisco (USA), 2016, pp. 57–59.
- [31] M. A. del Agua, A. Martínez-Villaronga, S. Piqueras, A. Giménez, A. Sanchis, J. Civera, and A. Juan, "The MLLP ASR systems for IWSLT 2015," in *Proc. of 12th IWSLT*, Da Nang (Vietnam), 2015.
- [32] K. Yao, D. Yu, F. Seide, H. Su, L. Deng, and Y. Gong, "Adaptation of context-dependent deep neural networks for automatic speech recognition," in *Proc. of the SLT*, 2012, pp. 366–369.
- [33] 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 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7893–7897.
- [34] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.
- [35] J. A. Silvestre-Cerdà et al., "transLectures," in *IberSPEECH*, 2012, pp. 345–351.
- [36] G. Stemmer, F. Brugnara, and D. Giuliani, "Adaptive training using simple target models," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, 2005, pp. 997 – 1000.
- [37] H. A. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*. Norwell, MA, USA: Kluwer Academic Publishers, 1993.
- [38] G. Hinton et al., "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.
- [39] A. Graves, N. Jaitly, and A. r. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013, pp. 273–278.
- [40] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modelling," in *Proceedings of the Workshop on Human Language Technology*, 1994, pp. 307–312.
- [41] A. Martín et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: <http://tensorflow.org/>
- [42] RWTH, UPVLC, XEROX, and EML, "D3.1.1: First report on massive adaptation," transLectures, Tech. Rep., 2012. [Online]. Available: <http://www.translectures.eu/wp-content/uploads/2013/05/transLectures-D3.1.1-18Nov2012.pdf>
- [43] UPVLC, XEROX, JSI-K4A, RWTH, and EML, "D3.1.3: Final report on massive adaptation," transLectures, Tech. Rep., 2014. [Online]. Available: <https://www.translectures.eu/wp-content/uploads/2015/01/transLectures-D3.1.3-31Oct2014.pdf>
- [44] M. Siu and H. Gish, "Evaluation of word confidence for speech recognition systems," *Computer Speech & Language*, vol. 13, no. 4, pp. 299 – 319, 1999.
- [45] T. Lavergne, O. Cappé, and F. Yvon, "Practical very large scale CRFs," in *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2010, pp. 504–513.
- [46] A. Rousseau, P. Deléglise, and Y. Estève, "Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks," in *LREC*, 2014, pp. 3935–3939.
- [47] U. M. B. Neto and E. R. Dougherty, *Error estimation for pattern recognition*. John Wiley & Sons, 2015.
- [48] Y. Miao and F. Metze, "On speaker adaptation of long short-term memory recurrent neural networks," in *Interspeech*, 2015.
- [49] Z. Huang, J. Tang, S. Xue, and L. Dai, "Speaker adaptation of RNN-BLSTM for speech recognition based on speaker code," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5305–5309.



Miguel Ángel Del-Agua received his Degree in Computer Science (2010), his MSc in Artificial Intelligence, Pattern Recognition and Digital Imaging (2012), and now he is a PhD candidate at the Universitat Politècnica de València. He has been co-author of several articles in international conferences, and has been actively involved in several R&D projects (transLectures, EMMA, MORE). His current research interests include deep learning for natural language processing, and specifically speech recognition and language modelling.



Adrià Giménez is a postdoctoral researcher at the Universitat Politècnica de València (UPV). He received his PhD in Computer Science from UPV in 2014, and he is currently a member of the UPV's Machine Learning and Language Processing research group (MLLP). He has published over 20 articles in international journals and conferences, and he has been actively involved in 2 EU research projects and several Spanish research projects. His current research is mainly focused on deep learning for speech recognition.



Jorge Civera is an associate professor of Computer Science at the Universitat Politècnica de València (UPV). He received his PhD from UPV in 2008. He is a member of the UPV's Machine Learning and Language Processing research group (MLLP). He has participated in more than 15 research projects and has published over 40 articles in international journals and conferences. His most recent work includes his participation in the EC Network of Excellence PASCAL2, and the EC research projects transLectures (as project manager) and EMMA (as

leader at the UPV).



Albert Sanchis is a PhD associate professor of Computer Science at the Universitat Politècnica de València (UPV). He obtained his PhD in 2004, and he is a member of the UPV's Machine Learning and Language Processing research group (MLLP). He is the co-author of more than 50 articles in international journals and conferences. He has participated in 5 EU research projects and more than 10 Spanish research projects. He is now leading the Spanish government-funded project MORE on fostering open education by providing multilingual access to open

educational resources.



Alfons Juan is a Professor of Computer Science at the Universitat Politècnica de València (UPV), where he obtained his PhD in 2000, and now leads a research group on Machine Learning and Language Processing (MLLP). He has participated in more than 30 research projects and has published more than 130 articles in international journals and conferences. He has also been an advisor for 11 PhD theses on different MLLP topics. His most recent work includes the participation in the EU projects transLectures, EMMA and X5gon.