NEURAL CODES TO FACTOR LANGUAGE IN MULTILINGUAL SPEECH RECOGNITION

Markus Müller[1]*, Sebastian Stüker*[1]*, and Alex Waibel*[1,2]

[1]Karlsruhe Institute of Technology, Karlsruhe, Germany
[2]Carnegie Mellon University, Pittsburgh PA, USA

ABSTRACT

In the past, we adapted neural network based multilingual acoustic models using language codes. In this work, we study the extracted language codes and the language properties they encode: We use the codes to generate language prototype vectors, which represent the features of a language. Computing distances between prototype vectors shows that languages from the same family have smaller distances. This structure found within the feature representation supports the assumption that language codes do encode language information and not other properties like, e.g. channel characteristics, and in addition providing a richer language representation than the language identity alone.

The network architecture of our system is based on a factorized model, which consists of multiple language dependent subnets. While we recently demonstrated that this approach enables multilingual setups to outperform monolingual ones, we here propose further optimizations. We evaluated using a) more language dependent subnets and b) wider BiLSTM layers. Our results indicate that using a larger number of language dependent subnets increases the system performance and renders phonetic pretraining superfluous. In addition, increasing the size of the hidden layers further improved the performance, with the system now outperforming the monolingual baseline by 6.3% relative.

Index Terms— Neural adaptation, multilingual, end-to-end, CTC, speech recognition

1. INTRODUCTION

Multilingual speech recognition is a challenging task, as each language requires its own acoustic model to achieve good performance. Training systems jointly on data from multiple languages introduces additional ambiguity because of increased linguistic variability. We recently proposed a method for rapid adaptation of neural networks to languages [1, 2]. By using *language codes (LC)*, applied via neural modulation, we stimulate networks to learn features depending on language properties.

The work leading to these results has received funding from the European Union under grant agreement Nº 825460 and the Federal Ministry of Education and Research (Germany) / DLR Projektträger Bereich Gesundheit under grant agreement Nº 01EF1803B.

In this work, we first discuss LCs in general. We show that these codes not only enable the rapid adaption of acoustic models, but also display similar features for languages within the same language family. Furthermore, we evaluate the use of different language combinations, thereby rendering phonetic pre-training abundant and requiring only training on graphemic targets. In addition, we evaluate if the use of larger hidden layers improves the recognition accuracy of our setup.

This paper is organized as follows: In the next Section 2, we outline related work, followed by Section 3, where we discuss language codes and the properties encoded therein. The neural adaptation technique is outlined in Section 4. We describe our experiments in Section 5 and discuss the results in Section 6. We conclude this paper with Section 7, where we also outline future work.

2. RELATED WORK

Multilingual speech recognition has been in the focus of research for many years. We here provide an overview of work that has been done to adapt classical HMM-based setups to multiple languages and also outline neural adaptation methods, as well as approaches towards building all-neural speech recognition systems.

2.1. Multi- and Crosslingual Speech Recognition Systems

Traditional speech recognition systems combine several explicitly modeled components and are based on a GMM/HMM or DNN/HMM approach. Explicitly modeled components are, e.g., the language model, or the pronunciation dictionary. Training these systems multilingually requires the adaptation of each of these models to achieve good performance. Several methods were proposed to train and/or adapt such systems to multi- and crosslingual scenarios [3, 4, 5]. One important step during training of these systems is the clustering of context-independent targets into context-dependent ones. This process can also be adapted to account for cross- and multilinguality [6].

2.2. Neural Network Adaptation

Supplying additional features to the network typically enables neural adaptation. In the regimen of automatic speech recognition, the use of i-vectors [7, 8] is a common approach for speaker adaptation. They are a low dimensional representation of speaker properties. Speaker adaptive networks can be trained by shifting acoustic features based on these vectors [9]. An approach alternative to using i-vectors is the use of bottleneck speaker vectors (BSVs), which are extracted using a neural network trained to discriminate speakers [10]. We proposed a similar method for adapting DNN acoustic models (AMs) to languages when trained jointly on multiple languages. Our first approach uses the language identity, encoded via one-hot encoding [11]. We refined it by using Language Feature Vectors (LFVs) [12], which encode language properties as low-dimensional code instead of using the language identity only. Similar to BSVs, LFVs are extracted by a neural network. In comparison to using just the language identity, LFVs enable better language adaptation, which results in lower WERs. Alternative methods for language adaptation were proposed as well [13].

2.3. RNN Based ASR Systems

Speech recognition systems based entirely on neural networks gained a lot of research interest in recent years. With the emergence of increased computing capabilities and special purpose hardware like GPUs, more complex network architectures can be trained using larger datasets. One method to train neural networks for speech recognition without bootstrapping by a classical GMM/HMM system is the use of the connectionist temporal classification (CTC) loss function [14]. Recurrent neural networks (RNNs) are powerful tools for modeling sequential dependencies. In contrast to traditional systems, no context-dependent targets need to be modeled as the network learns context implicitly. A variety of acoustic modeling units can be used. Like in traditional systems, phones, graphemes, or both can be use to model the acoustics [15]. In addition, using whole words is also possible, given enough training data [16]. Recently, the use of subword units, so-called BPE (byte pair encoding) units were proposed [17] and do perform better than graphemes.

Another family of systems is based on an approach originating from the regimen of machine translation: Listen, Attend, Spell [18]. It factorizes the model into an encoder, decoder and attention. Based on this approach, speech recognition systems can be trained on multiple languages as well [19]. A more recent approach is using self-attention, combined with CTC [20].

3. LANGUAGE CODES

Key to our proposed language adaptation method are *language codes (LC)*, which are used to *modulate* the acoustic model of our speech recognition system. To extract LCs, we train an ancillary feed-forward neural network on the auxiliary task of language identification. This network contains a bottleneck layer as second-to-last layer. As input, we use multilingual bottleneck features extracted by a typical speech recognition pipeline using a window size of 32ms with a frame shift of 10ms. The network for extracting these features is trained on log Mel scaled and tonal features, using data from 5 languages (German, French, Italian, Russian, Turkish). The language identification network is then trained on 9 languages (Arabic, French, German, Italian, Polish, Portuguese, Russian, Spanish, Turkish). We use an output layer with 9 output units, each representing one language.

All layers after the bottleneck are discarded when the training is done and the output activations of the bottleneck layer are used as continuous representation of language properties. We consider the language properties to be stationary, longer-duration features. Therefore, a large context window is input into to the language identification network, covering 700ms. This enables the network to capture these longer-duration properties. For each time step, the network outputs one language feature vector. To smooth the output and generate a robust representation, LCs are extracted by averaging the language feature vectors on utterance level.

3.1. Language Prototype Vectors

To study if LCs do encode language properties instead of other, non language related differences between recordings of different languages, we seek to derive higher order language features. One aspect is the determination of the closeness of languages. Hence we create *language prototype vectors (LPVs)* for each of the training languages we consider for the acoustic model. For this, we extract LCs on the training data and average them on a per language basis. Note that we also extract LCs for English, although this language was not seen during training of the language code network. To determine the distances between the LPVs, we used the Euclidean distance. Figure 1 shows the distances between the prototype vectors of the various languages. Lower values represent a smaller difference. The values are normalized to be in the range of $[0, 1]$.

Except for English, the distances between training languages are close to 1.0, with Spanish and Italian (both languages from the Italic-Romance family) showing the highest closeness with a distance of 0.76. For English, which has not been seen during training, lower distances for all other languages are observed. The rationale behind this is that the LC network attempts to represent the properties of English using a combination of the source languages. Comparing LPVs, the distance from English to Russian (a Slavic language) is the highest, whereas the distance to German (a Germanic language like English) is the lowest. This is an indication that LCs do not only encode the language identity, but higher-

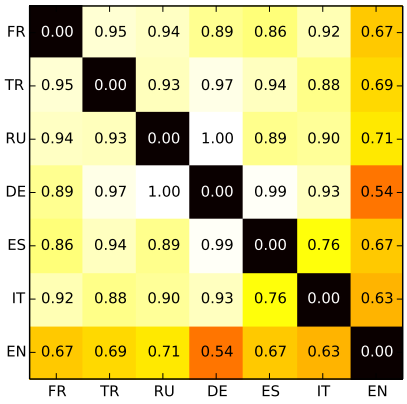

Fig. 1. Euclidian distances between language prototype vectors (LPVs).

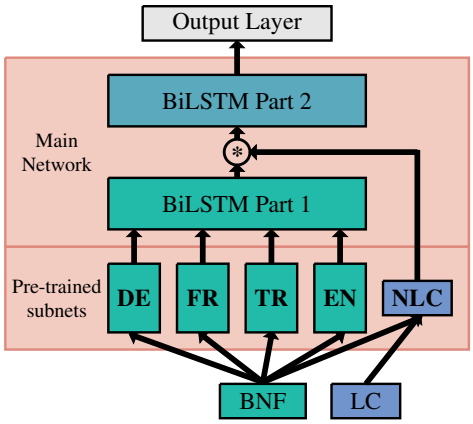

Fig. 2. Combinational superstructure using adaptive neural language codes (NLC) for network modulation. 4 language dependent subnets are shown.

order language related features.

4. LANGUAGE ADAPTED SPEECH RECOGNITION

We recently proposed a novel approach to adapt all-neural CTC-based ASR systems to languages called "modulation" [2]. We use LCs to enable language adaptation of the neural network. The network architecture is shown in Figure 2. Our system uses a factorized model which is assembled out of multiple, pre-trained networks. Each subnetwork is trained on a single language and features 105 BiLSTM cells per layer and 3 layers in total. The main network is split into two parts with 2 layers and 420 BiLSTM units per layer. While auxiliary features are typically added at the input of networks, we opted for an approach that incorporates language properties deeper into the network architecture and in an adaptive manner using an additional network with 2 BiLSTM layers.

The purpose of this neural language codes (NLCs) subnetwork is to transform the LCs into a representation that is more favorable regarding the recognition accuracy of the system: as part of our combinational superstructure, its parameters will be updated during the joint training and the extracted codes will be updated according to the global objective function. The network consists of 2 BiLSTM layers and is pre-trained to output the LCs unaltered using BNFs and LCs as input features. It will therefore initially learn to ignore the BNFs and to simply forward LCs. But during the joint optimization, it can take advantage of the additional features. We added NLCs after the first BiLSTM part by modulation, this method gates the outputs of neurons based on external features [1].

5. EXPERIMENTAL SETUP

Our experiments are based on the Euronews corpus [21], containing recordings of TV broadcast news from the Euronews TV station. The program of this station is dubbed in many languages. By using this data, we ensure that the recordings feature to a large extend the same acoustic conditions across languages. We use 45h of transcribed acoustic training data per language. As acoustic features, we use multilingual bottleneck (ML-BNF) features, extracted using a feed-forward neural network trained on a combination of 5 languages (German, French, Italian, Russian, Turkish). To evaluate the performance of our system, we report WERs on English.

5.1. Combination of Languages

While we previously used only 4 language dependent subnets (English, German, French, Turkish), we here extend the pool of languages to 7 by adding networks pretrained on Italian, Russian and Spanish. As target languages, we keep our set of 4 languages fixed (English, German, French, Turkish). In case of only 4 language dependent subnets, we pretrained the networks for German, French and Turkish using phonemic targets. For the 7-language case, the networks were pretrained solely on graphemic targets.

Condition	Languages
Baseline	DE, EN, FR, TR
Enhanced	DE, EN, FR, TR, ES, IT, RU

Table 1. Language combinations used for language dependent subnets

5.2. Network training

The training schedule for the combinational superstructure consists of multiple steps. First, we trained the language dependent subnets, as well as the NLC net. After all subnets are trained, we combine them and add two BiLSTM blocks,

as shown in Figure 2. This superstructure will then be jointly trained, allowing updates to the weights of all networks. The NLC network can thereby take advantage of additional BNF input to further refine the LCs, enabling better language adaptation and higher recognition accuracies.

6. RESULTS

We first evaluate the combination of different language dependent subnets. Based on the best configuration, we then increase the size of the main network and compare the performance.

6.1. Language Dependent Subnets

We start with the evaluation of different subnet combinations, with results shown in Table 2. Our baseline uses a combination of subnets from 4 languages, where the networks for German, French and Turkish are pretrained using phonetic targets. As the first experiment, we use subnets from the same combination of languages, but train them using solely graphemic targets. This results in a drop in performance from 24.2% WER to 25.6% WER. Next, we evaluate using more subnets by adding subnets from three more languages; all pretrained using only graphemic targets.

Using more language dependent subnets yields to the same performance (24.2% WER vs. 24.3% WER) as using fewer subnets, but without the need for pretraining them on phonemic targets.

Condition	Languages	WER
Baseline	DE, EN, FR, TR	24.2%
Grapheme only	DE, EN, FR, TR	25.6%
Grapheme only	DE, EN, FR, TR, IT, RU, TR	24.3%

Table 2. Language combinations used for language dependent subnets

6.2. Network Size

As last evaluation, we compare different layer sizes of the main network. Starting with 420 BiLSTM cells per layer as baseline, we doubled the size to 840. To account for the increase in GPU memory requirements, we reduced the mini-match size and accumulated the weight updates over several mini-batches before applying them to keep the number of parameter updates the same as if the mini-batch size would remain unchanged.

Starting with our baseline setup using 420 BiLSTM cells per layer, our multilingual setup shows an improvement of 3.4% relative to the monolingual target, as shown in Table 3. Doubling the number of BiLSTM cells reduces the WER of both our adapted setup, as well as the monolingual one. In comparison, using larger layers improves the relative improvement to 6.3%. This shows not only the effectiveness of our adaptation method, but also that using larger networks enables better language adaptation. While using even larger layers should potentially lead to even larger improvements, practical restrictions regarding training time do require additional optimization of the network training as in, e.g., the use of multiple GPUs in parallel. This will be subject of future work.

BiLSTM Block Size	WER
Monolingual 420	24.3%
Multilingual 420	**23.5%**
Monolingual 840	23.7%
Multilingual 840	**22.3%**

Table 3. Language combinations used for language dependent subnets

7. CONCLUSION

We introduced a language adaptation method for multilingual, all-neural CTC-based speech recognition systems based on language codes (LCs) as auxiliary features. In this work, we analyzed the properties of these codes. We combine extracted LCs to LPVs (language prototype vectors), which encode features of languages. Computing distances between LPVs shows, that languages of the same family have smaller distances than languages from different families. This is strong evidence that language codes are a low-dimensional representation of language features.

Key to our adaptation method is the factorization of the model by pre-training subnets on different languages. Here, we further refined our approach by analyzing the importance of the source languages chosen for a given set of target languages. While we previously required some language dependent subnets to be trained using phonetic targets, we here demonstrated that training entirely on graphemic targets is possible if the pool of source languages is increased. Using the best combination of languages, we could further show improvements by increasing the number of BiLSTM units per layer.

In total, we achieved an improvement of 6.3% relative (instead of 3.4% in our previous work) over the monolingual target without the need of phonetic pre-training by increasing both the number of source languages as well as the size of the hidden layers. Future work includes the optimization of the training procedure, including the parallelization of the training using multiple GPUs.

8. REFERENCES

[1] Markus Müller, Sebastian Stüker, and Alex Waibel, "Multilingual adaptation of RNN based ASR systems," in *Acoustics, Speech and Signal Processing (ICASSP), 2018 IEEE International Conference on*. IEEE, 2018.

[2] Markus Müller, Sebastian Stüker, and Alex Waibel, "Neural language codes for multilingual acoustic models," in *Interspeech*, 2018.

[3] Tanja Schultz and Alex Waibel, "Fast bootstrapping of LVCSR systems with multilingual phoneme sets.," in *Eurospeech*, 1997.

[4] Tanja Schultz and Alex Waibel, "Multilingual and crosslingual speech recognition," in *Proc. DARPA Workshop on Broadcast News Transcription and Understanding*. Citeseer, 1998, pp. 259–262.

[5] Sebastian Stüker, *Acoustic modelling for under-resourced languages*, Ph.D. thesis, Karlsruhe, Univ., Diss., 2009, 2009.

[6] Sebastian Stüker, "Modified polyphone decision tree specialization for porting multilingual grapheme based ASR systems to new languages," in *Proceedings of the 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Las Vegas, NV, USA, April 2008, pp. 4249–4252, IEEE.

[7] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[8] George Saon, Hagen Soltau, David Nahamoo, and Michael Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *ASRU*. IEEE, 2013, pp. 55–59.

[9] Yajie Miao, Hao Zhang, and Florian Metze, "Towards speaker adaptive training of deep neural network acoustic models," 2014.

[10] Hengguan Huang and Khe Chai Sim, "An investigation of augmenting speaker representations to improve speaker normalisation for DNN-based speech recognition," in *ICASSP*. IEEE, 2015, pp. 4610–4613.

[11] Markus Müller and Alex Waibel, "Using language adaptive deep neural networks for improved multilingual speech recognition," *IWSLT*, 2015.

[12] Markus Müller, Sebastian Stüker, and Alex Waibel, "Language adaptive DNNs for improved low resource speech recognition," in *Interspeech*, 2016.

[13] Ngoc Thang Vu, David Imseng, Daniel Povey, Petr Motlicek, Tanja Schultz, and Hervé Bourlard, "Multilingual deep neural network based acoustic modeling for rapid language adaptation," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 7639–7643.

[14] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.

[15] Dongpeng Chen, Brian Mak, Cheung-Chi Leung, and Sunil Sivadas, "Joint acoustic modeling of triphones and trigraphemes by multi-task learning deep neural networks for low-resource speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 5592–5596.

[16] Hagen Soltau, Hank Liao, and Hasim Sak, "Neural speech recognizer: Acoustic-to-Word LSTM model for large vocabulary speech recognition," *arXiv preprint arXiv:1610.09975*, 2016.

[17] Thomas Zenkel, Ramon Sanabria, Florian Metze, and Alex Waibel, "Subword and crossword units for CTC acoustic models," *arXiv preprint arXiv:1712.06855*, 2017.

[18] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4960–4964.

[19] Shinji Watanabe, Takaaki Hori, and John R Hershey, "Language independent end-to-end architecture for joint language identification and speech recognition," in *Automatic Speech Recognition and Understanding Workshop (ASRU), 2017 IEEE*. IEEE, 2017, pp. 265–271.

[20] Julian Salazar, Katrin Kirchhoff, and Zhiheng Huang, "Self-attention networks for connectionist temporal classification in speech recognition," *arXiv preprint arXiv:1901.10055*, 2019.

[21] Roberto Gretter, "Euronews: A multilingual benchmark for ASR and LID," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.