ENHANCING MULTILINGUAL GRAPHEMIC RNN BASED ASR SYSTEMS USING PHONE INFORMATION

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

[1]*Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology*
[2]*Language Technology Institute, Carnegie Mellon University*
m.mueller@kit.edu

Abstract: In the past, we proposed the use of Language Feature Vectors (LFVs) to better adapt multilingual speech recognition systems to languages. Recently, we applied this method to RNN/CTC based systems. The recognition accuracy could be improved by modulating the network using LFVs. In this work, we propose an improvement to this approach by refining the network architecture as well as the training strategy. We first evaluated multiple methods for applying the modulation. As we are using bi-directional layers, each unit outputs two values, one per direction. Optimizing the combination of the outputs for each direction did improve the performance. In addition, we propose a method for including phonetic information into the training process of a graphemic system. By pre-training layers using phones as targets, the network did learn features to discriminate phones. Adding more layers and a two stage fine-tuning process using graphemes, we first forced the network map phonetic features to graphemes. In the second stage, we allowed the network to update the phonetic feature detectors as well. Both methods improved the performance of our setup. We evaluated our setup using a combination of 4 languages (English, French, German, Turkish), with a joint set of acoustic units.

1 Introduction

Lately, automatic speech recognition systems based on recurrent neural networks (RNNs), trained using the connectionist temporal classification (CTC) loss function [1], have gained substantial research interest. Due to the recurrent nature of the networks used, such systems are able to capture sequential dependencies implicitly, without the need to explicitly model context by, e.g., context-dependent phone states. In this work, we focus on multilingual systems, based on graphemes as acoustic modeling units. We aim at training a single, multilingual model which is able to recognize speech from multiple languages simultaneously. In the past, we presented an approach to train RNN/CTC systems jointly on multiple languages using a global set of either graphemes or phonemes as acoustic units. While using graphemes as acoustic units can be a challenging task for languages with complex pronunciation rules, e.g., English, training a system on a combination of languages is even more difficult: The network has to infer the pronunciation rules for multiple instead of a single language. In order to adapt networks to multiple languages, we proposed using Language Feature Vectors (LFVs). Extracted via a neural network, they encode language specific peculiarities as low dimensional vector. For RNNs, similar to feed-forward networks, appending the LFVs to the input features increases the performance. In a series of experiments, we determined that modulating the layers of RNNs does result in better performance. Akin to dropout, the modulation emphasizes or attenuates the outputs of neurons, based on encoded language properties. This forces the network to adapt

feature detectors based on language features which improves the performance in a multilingual setting.

In this work, we extended our approach in two ways: a) by optimizing the network architecture and b) integrating phonetic information into grapheme based systems. First, we optimized the network architecture. Since we are using bi-directional LSTM layers, the output of each layer has twice the size of the number of LSTM cells. As input to the next layer, these outputs could be passed along unchanged, or the outputs for each direction could be combined pairwise. In a series of experiments, we compared the performance of addition, multiplication or taking the maximum of the outputs (inspired by maxout networks [2]) to simply passing the outputs along unchanged.

In order to add phonetic information, we choose to train the networks in a two step approach: First, we pre-trained the network using phones as targets. Next, we replaced the output layer trained on phones, added two additional LSTM layers and fine-tuned the network again, using graphemes as targets. Pre-training with phonetic targets forced the network to extract features to discriminate phones. The fine-tuning process using graphemes then allowed for the network to learn a mapping from phonetic features to graphemes. This paper is organized as follows: In the next Section 2, we provide an overview of related work in the field. In Section 3, we outline our proposed method. The experimental setup is described in Section 4, followed by the results in Section 5. This paper concludes with Section 6, where we also provide an outlook to future work.

2 Related Work

Using GMM/HMM based systems was considered state of the art prior to the emergence of systems with neural networks. Data sparsity has been addressed in the past, by training systems multi- and crosslingually [3, 4]. Methods for crosslingual adaptation exist [5], but also methods for adapting the cluster tree were proposed [6].

2.1 Multilingual BNFs

Deep Neural Networks (DNNs) are a data-driven model with many parameters to be trained, failing to generalize if trained on only a limited data set. Different methods have been proposed to train networks on data from multiple source languages. Training DNNs typically involves a pre-training and a fine-tuning step. It has been shown, that the pre-training is language independent [7]. Several approaches exist to fine-tune a network using data from multiple languages. One method is to share hidden layers between languages, but use language specific output layers [8, 9, 10]. Combining language specific output layers into one layer is also possible [11]. By dividing the output layer into language specific blocks, the setup uses language dependent phone sets. Training DNNs simultaneously on data from multiple languages can be considered multi-task learning [12, 13].

2.2 Neural Network Adaptation

Several methods for adapting neural networks to different speakers have been proposed. Using i-Vectors [14] is the most common method. Based on this low dimensional representation of speaker properties, speaker adaptive neural networks [15] can be built. An alternative method for adaptation are Bottleneck Speaker Vectors (BSVs) [16]. These methods show that neural networks benefit from additional input modalities. Similar to BSVs, we proposed an adaptation method for adapting neural networks to different languages when trained multilingually. We first proposed using the language identity information via one-hot encoding [17]. But this

method does not supply language characteristics to the network. Language Feature Vectors (LFVs) [18, 19] have shown to encode such language properties, even if the LFV net was not trained on the target language.

2.3 CTC Based Systems

Recently, CTC-based [1] systems have gained in popularity. Such systems can be trained on either phones or graphemes as acoustic units, but also jointly together [20, 21]. Multi-task learning has also been proposed [22, 23, 24]. It also has been demonstrated that CTC based systems are able to outperform HMM based setups [25]. We proposed a first approach towards training CTC systems multilingually [26], and refined it by using more languages [27] and modulation [28].

3 Multilingual RNN/CTC Based Systems

There are multiple ways of building multilingual systems. Here, we built systems using a global set of acoustic units. This enables the acoustic model to recognize speech independent of the language. Traditional speech recognition systems are built to recognize speech of a single language. The performance of a system having a language dependent set of acoustic units is typically better. We therefore aim at developing adaptation techniques to close the gap between systems with multi- and monolingual sets of acoustic units.

3.1 Network Architecture

Our network architecture is based on Baidu's Deepspeech 2 [29]. As input features, we use multilingual bottleneck features (ML-BNFs). We therefore omitted the convolutional layers of Baidu's architecture, as adjacent dimensions in ML-BNFs have so spacial relation. The architecture, shown in Figure 1, uses 4 layers with bi-directional LSTM (BLSTM) cells, with a final feed-forward output layer. After pre-training, two additional layers were added.

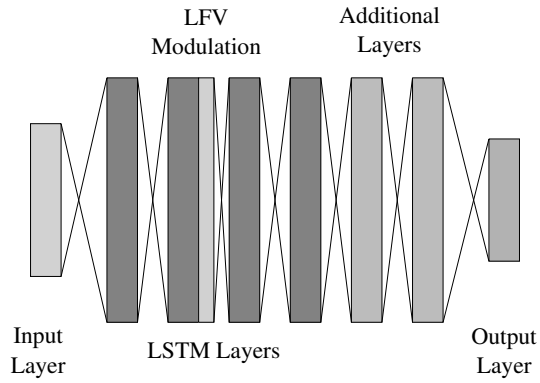

Figure 1 – Network layout with modulation of the output after the second hidden layer with LFVs and additional layers added after pre-training.

3.2 Neural Network Modulation

One method for adapting the acoustic model to multiple languages is the use of LFVs [18]. Extracted via a neural network, they encode language properties which enable adaptation to multiple languages. A similar technique based on i-Vectors is used to adapt networks to the properties of speakers. For applying the LFVs, we used a method called "modulation" instead of simply appending the features to the acoustic input vectors. The modulation, introduced as part of Meta-PI networks [30], multiplies the output of a unit with a coefficient. The number of LSTM cells in the hidden layers was chosen to be a multiple of the dimensionality of LFVs. The output of each layer could therefore be split into groups of equal size and each group was multiplied with one LFV coefficient. Modifying the output of units is similar to dropout training, where connections between units are being dropped on a random basis. The method proposed here could be considered an "intelligent dropout", where the outputs of units are emphasized or attenuated based on LFV coefficients. This forces the network to learn features based on language properties. Each group of units is modulated with a dimension of the LFVs, which forces networks to learn features depending on the language representation. As we are using bi-directional layers, each layer outputs two values for each LSTM cell. There are multiple possibilities of forwarding these values from one layer to the next. With respect to the modulation, we evaluated multiple configurations of combining the outputs for both directions.

3.3 Training Strategy

We trained our networks using a multi step approach. First, we pre-trained 4 BLSTM layers using phones as targets. Two additional layers were added and the network was trained using graphemes as targets, while keeping the weights of the first 4 layers fixed. A final fine-tuning step updating all network weights was then performed. Training a network in this manner forces it to first learn features to discriminate between phones. Keeping the first 4 layers fixed during the first round of fine tuning causes the two additional layers to learn a mapping between phonetic features and graphemes. The final fine-tuning step updating all the weights enables the network to further optimize the feature detectors in the first layers for discriminating graphemes.

4 Experimental Setup

For our experiments, we used data from the Euronews Corpus [31], which contains TV broadcast news recordings covering multiple languages. In total, this corpus includes data from 10 languages. We filtered the data, excluding all utterances being shorter than 1s, or having a transcript longer than 639 symbols[1]. After filtering, approx. 50h of data per language remained. As test set, 5h were selected randomly on a speaker basis.

As input features to our systems, we used ML-BNFs. The network for their extraction was trained on data from 5 languages (French, German, Italian, Russian, Turkish). During training, the hidden layers were shared between languages, with the output layers being language dependent. Training data was obtained using traditional DNN/HMM based systems to force align the reference transcriptions to the recordings and to obtain frame-level state labels. As input features to the network, a combination of logMEL and tonal features (FFV and pitch) was used.

[1] Internal limitation within the implementation of CUDA/warp-ctc, see: https://github.com/baidu-research/warp-ctc, accessed 2018-01-23

4.1 Acoustic Units

We used two types of acoustic units to train our networks: Phones and graphemes. In order to create a pronunciation dictionary, MaryTTS [32] was used. As we were using data from multiple languages in a multilingual setting, we needed to establish a mapping of phone symbols across languages to ensure that the same phone is represented by the same symbol independent of the language. This mapping was established using the definitions of articulatory features within each of MaryTTS' language definition files.

4.2 Neural Network Training

The network for the extraction of LFVs is trained using data from 9 languages (Arabic, French, German, Italian, Polish, Portuguese, Russian, Spanish, Turkish). ML-BNFs were used as input features. The network featured a similar configuration as BNF networks by having 6 hidden layers and a narrow bottleneck layer as second last layer. In contrast to speech recognition systems, this network requires a much larger context as the language information is long-term in nature. We used a context window covering $+/-$ 33 frames in each direction, resulting in a total context of 690ms. After training, all layers after the bottleneck were discarded and the output activations of the bottleneck were used as LFVs.

The BLSTM networks of the acoustic modeling were trained on 4 languages (English, French, German, Turkish) using stochastic gradient descent and Nesterov momentum with a factor of 0.9. Weight updates were applied using mini-batches with a size of 15. During the first epoch, the training utterances were sorted ascending by length to stabilize the network training, as, in general, shorter utterances are easier to align.

4.3 Evaluation

We evaluated our setup using two error metrics: Token Error Rate (TER) as well as Word Error Rate (WER). TER was used to evaluate the performance of the RNN without decoding with a language model (LM). WER was used in combination with an character based RNN-LM and greedy decoding using the best path. The language model was trained on the training utterances only and is therefore not a strong language model. But it should provide an indication, if the improvements reported in TER are also reflected in the WER of the system.

5 Results

5.1 Combining layer outputs

When using bi-directional networks, each unit outputs one output for each direction which doubles the outputs of each layer. For applying the modulation, we evaluated ways of combining the outputs for each direction by either simply appending the outputs for each direction or adding, multiplying, taking the maximum value pairwise for each direction. The results using graphemes are shown in Table 1 and phones in Table 2. We reported the results for all 4 languages using both graphemes and phones.

When using graphemes, the pairwise multiplication of the outputs leads to the worst performance. Taking the maximum value of each pair does result in the best performance, except for English and German where it is en par with summation. Using phones, the maximum does result in the best performance for English, German and Turkish, whereas appending the elements shows the best results for French. The multiplication here also produces the worst results. In general, taking the maximum for each direction does improve the performance in almost all

cases and only reduces the performance for French using phones. This idea was first introduced as maxout networks [2], where the neuron would output the maximum value of the inputs.

Table 1 – Evaluation of merging strategies, TER on grapheme based systems

Strategy	DE	EN	FR	TR
Append	7.8	11.2	8.9	6.1
Sum	**7.7**	**11.0**	9.0	6.2
Multiply	7.9	11.7	9.2	6.2
Max	**7.7**	**11.0**	**8.8**	**6.0**

Table 2 – Evaluation of merging strategies, TER on phone based systems

Strategy	DE	EN	FR	TR
Append	**6.7**	11.8	**9.5**	5.8
Sum	7.0	12.1	10.1	6.0
Multiply	7.2	12.8	10.5	6.0
Max	**6.7**	**11.7**	9.8	**5.7**

5.2 Phonetic Pre-Training

Next, we evaluated performing a pre-training step using phonetic targets with the best merging strategy as determined in the previous section. As shown in Table 3, pre-training the network on phonemes and adding two additional layers does result in better performance. As contrastive experiment, we included results using a 6 layer network. Simply using more layers does not improve the performance. WERs are shown in Table 4 and show that improvements can also be observed when decoding with an LM.

Table 3 – Evaluation of phonetic pre-training, TER on grapheme based systems

Strategy	DE	EN	FR	TR
Baseline (4L)	7.7	11.0	8.8	6.0
Baseline (6L)	9.0	12.7	10.3	7.5
Pre-training	**6.0**	**10.0**	**8.7**	**5.3**

Table 4 – Evaluation of phonetic pre-training, WER

Strategy	EN
Baseline	26.3
Pre-training	**25.4**

6 Conclusion

We presented two refinements to language adaptation using LFVs. By the pairwise combination of outputs from the LSTM cells for each direction, we could improve the TER. In addition, pre-training networks using phonetic targets does improve the performance for graphemic systems. The networks first learn to discriminate phonetic features and are then able to learn pronunciation rules based on these features more easily. Future work includes additional optimization of the network architecture and training strategies, as well as the expansion to more languages.

References

[1] GRAVES, A., S. FERNÁNDEZ, F. GOMEZ, and J. SCHMIDHUBER: *Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks*. In *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376. ACM, 2006.

[2] MIAO, Y., F. METZE, and S. RAWAT: *Deep maxout networks for low-resource speech recognition*. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pp. 398–403. IEEE, 2013.

[3] WHEATLEY, B., K. KONDO, W. ANDERSON, and Y. MUTHUSAMY: *An evaluation of cross-language adaptation for rapid hmm development in a new language*. In *Acoustics, Speech, and Signal Processing, 1994. ICASSP-94., 1994 IEEE International Conference on*, vol. 1, pp. I–237. IEEE, 1994.

[4] SCHULTZ, T. and A. WAIBEL: *Fast bootstrapping of lvcsr systems with multilingual phoneme sets.* In *Eurospeech*. 1997.

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

[6] SCHULTZ, T. and A. WAIBEL: *Language-independent and language-adaptive acoustic modeling for speech recognition*. *Speech Communication*, 35(1), pp. 31–51, 2001.

[7] SWIETOJANSKI, P., A. GHOSHAL, and S. RENALS: *Unsupervised cross-lingual knowledge transfer in DNN-based LVCSR*. In *SLT*, pp. 246–251. IEEE, IEEE, 2012.

[8] GHOSHAL, A., P. SWIETOJANSKI, and S. RENALS: *Multilingual training of Deep-Neural networks*. In *Proceedings of the ICASSP*. Vancouver, Canada, 2013.

[9] SCANZIO, S., P. LAFACE, L. FISSORE, R. GEMELLO, and F. MANA: *On the use of a multilingual neural network front-end*. In *Proceedings of the Interspeech*, pp. 2711–2714. 2008.

[10] VESELY, K., M. KARAFIAT, F. GREZL, M. JANDA, and E. EGOROVA: *The language-independent bottleneck features*. In *Proceedings of the Spoken Language Technology Workshop (SLT), 2012 IEEE*, pp. 336–341. IEEE, 2012.

[11] GRÉZL, F., M. KARAFIÁT, and K. VESELY: *Adaptation of multilingual stacked bottleneck neural network structure for new language*. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 7654–7658. IEEE, 2014.

[12] CARUANA, R.: *Multitask learning*. *Machine learning*, 28(1), pp. 41–75, 1997.

[13] MOHAN, A. and R. ROSE: *Multi-lingual speech recognition with low-rank multi-task deep neural networks*. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pp. 4994–4998. IEEE, 2015.

[14] SAON, G., H. SOLTAU, D. NAHAMOO, and M. PICHENY: *Speaker Adaptation of Neural Network Acoustic Models Using i-Vectors*. In *ASRU*, pp. 55–59. IEEE, 2013.

[15] MIAO, Y., H. ZHANG, and F. METZE: *Towards Speaker Adaptive Training of Deep Neural Network Acoustic Models*. 2014.

[16] HUANG, H. and K. C. SIM: *An Investigation of Augmenting Speaker Representations to Improve Speaker Normalisation for DNN-based Speech Recognition*. In *ICASSP*, pp. 4610–4613. IEEE, 2015.

[17] MÜLLER, M. and A. WAIBEL: *Using Language Adaptive Deep Neural Networks for Improved Multilingual Speech Recognition*. *IWSLT*, 2015.

[18] MÜLLER, M., S. STÜKER, and A. WAIBEL: *Language Adaptive DNNs for Improved Low Resource Speech Recognition*. In *Interspeech*. 2016.

[19] MÜLLER, M., S. STÜKER, and A. WAIBEL: *Language Feature Vectors for Resource Constraint Speech Recognition*. In *Speech Communication; 12. ITG Symposium; Proceedings of*. VDE, 2016.

[20] CHEN, D., B. MAK, C.-C. LEUNG, and S. 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*, pp. 5592–5596. IEEE, 2014.

[21] RAO, K. and H. SAK: *Multi-accent speech recognition with hierarchical grapheme based models*. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 4815–4819. IEEE, 2017.

[22] KIM, S., T. HORI, and S. WATANABE: *Joint ctc-attention based end-to-end speech recognition using multi-task learning*. *arXiv preprint arXiv:1609.06773*, 2016.

[23] LU, L., L. KONG, C. DYER, and N. A. SMITH: *Multi-task learning with ctc and segmental crf for speech recognition*. *arXiv preprint arXiv:1702.06378*, 2017.

[24] SAK, H. and K. RAO: *Multi-accent speech recognition with hierarchical grapheme based models*. 2017.

[25] MIAO, Y., M. GOWAYYED, X. NA, T. KO, F. METZE, and A. WAIBEL: *An empirical exploration of ctc acoustic models*. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pp. 2623–2627. IEEE, 2016.

[26] MÜLLER, M., S. STÜKER, and A. WAIBEL: *Language adaptive multilingual ctc speech recognition*. In *International Conference on Speech and Computer*, pp. 473–482. Springer, 2017.

[27] MÜLLER, M., S. STÜKER, and A. WAIBEL: *Phonemic and graphemic multilingual ctc based speech recognition*. *arXiv preprint arXiv:1711.04564*, 2017.

[28] MÜLLER, M., S. STÜKER, and A. WAIBEL: *Multilingual adaptation of rnn based asr systems*. *arXiv preprint arXiv:1711.04569*, 2017.

[29] AMODEI, D., R. ANUBHAI, E. BATTENBERG, C. CASE, J. CASPER, B. CATANZARO, J. CHEN, M. CHRZANOWSKI, A. COATES, G. DIAMOS ET AL.: *Deep speech 2: End-to-end speech recognition in english and mandarin*. *arXiv preprint arXiv:1512.02595*, 2015.

[30] HAMPSHIRE, J. B. and A. WAIBEL: *The Meta-Pi Network: Building Distributed Knowledge Representations for Robust Multisource Pattern Recognition*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(7), pp. 751–769, 1992.

[31] GRETTER, R.: *Euronews: A Multilingual Benchmark for ASR and LID*. In *Fifteenth Annual Conference of the International Speech Communication Association*. 2014.

[32] SCHRÖDER, M. and J. TROUVAIN: *The german text-to-speech synthesis system mary: A tool for research, development and teaching*. *International Journal of Speech Technology*, 6(4), pp. 365–377, 2003.