From Speech to Letters - Using a Novel Neural Network Architecture for Grapheme Based ASR

Florian Eyben [#1], Martin Wöllmer [#2], Björn Schuller [#3], Alex Graves [*4]

[#] *Institute for Human-Machine Communication, Technische Universität München, 80333 München, Germany*
[1] eyben@tum.de
[2] woellmer@tum.de
[3] schuller@tum.de

[*] *Institute for Computer Science VI, Technische Universität München, Germany*
[4] graves@in.tum.de

***Abstract*—Main-stream Automatic Speech Recognition systems are based on modelling acoustic sub-word units such as phonemes. Phonemisation dictionaries and language model based decoding techniques are applied to transform the phoneme hypothesis into orthographic transcriptions. Direct modelling of graphemes as sub-word units using HMM has not been successful. We investigate a novel ASR approach using Bidirectional Long Short-Term Memory Recurrent Neural Networks and Connectionist Temporal Classification, which is capable of transcribing graphemes directly and yields results highly competitive with phoneme transcription. In design of such a grapheme based speech recognition system phonemisation dictionaries are no longer required. All that is needed is text transcribed on the sentence level, which greatly simplifies the training procedure. The novel approach is evaluated extensively on the Wall Street Journal 1 corpus.**

I. Introduction

Up to now Automatic Speech Recognition (ASR) has been based on modelling acoustic sub-word units, i.e. phonemes. Few attempts have been made to model graphemes (i.e. letters, numbers, punctuation marks etc.) directly. Due to the HMM-based modelling used in most state-of-the-art speech recognition systems phoneme based modelling was preferred, since phonemes obviously are closely related to the acoustic observations. In order to model graphemes directly an HMM system must consider much more context, which is tricky, and requires advanced methods such as using context dependent n-gram models and long-term features. In the last few years a novel method for speech recognition has been published [1] which - without modifications - is in principle able to transcribe letters directly by modelling long-range context. The method uses Bidirectional Long Short-Term Memory (BLSTM) Recurrent Neural Networks (RNN) [2] with the Connectionist Temporal Classification (CTC) output layer [1]. We refer to the combined system as BLSTM-CTC. It has proven highly successful for sequence transcription tasks involving difficult, real-world data [3], [4], including phoneme recognition on the TIMIT corpus [5]. Similar network architectures have been used for other speech related recognition tasks where context information is beneficial, e. g. [6], [7].

To evaluate the ability of these networks to transcribe speech directly to graphemes we apply BLSTM-CTC to a grapheme based speech recognition task employing the 1993 Wall Street Journal 1 ($WSJ1$) corpus. We believe grapheme recognition using BLSTM-CTC is very interesting since the networks in theory can learn local acoustic modelling as well as long-range contextual information such as language modelling or vocabulary. The approach is principally language independent. For training recognisers for various languages only a set of written transcriptions of the recordings without phonemisations is required. Such an approach is especially advantageous for languages lacking a well defined phoneme set (cf. [8]).

This paper is structured as follows: section II gives a brief overview of related work of grapheme based speech recognition. The BLSTM-CTC architecture is explained in section III. The database and the evaluation procedure are presented in section IV. The results obtained with BLSTM-CTC for both phoneme and grapheme recognition are discussed in section V before we summarise our findings and provide an outlook in section VI.

II. Related Work

Up to now, only few - compared to phoneme based ASR - publications exist addressing direct grapheme transcription of spoken utterances (e.g. [9], [10], [11], [12]). This is not surprising since, for the English language at least, no direct relation between written text and acoustic realisation (pronunciation) exists. Since HMM locally model acoustic sub-word units, they are unable to handle the long-range context information required to interpret different context dependent pronunciations belonging to the same lexical representation. Killer investigates grapheme based ASR in [9], where an innovative decision tree based clustering method is used in order to build context dependent tri-grapheme models for using letters instead of phonemes as sub-word units. Moreover, [9] investigates HMM based grapheme recognition for multiple languages. The main conclusion is that grapheme recognition works well for languages such as German where the graphemes correspond closely to the phonemes, but is inferior to phoneme based recognition for languages such as English

that lack this close correspondence. A similar work is reported in [13].

Another interesting, yet only loosely related, ASR approach where grapheme recognition is used was published in [14]. Here, grapheme based modelling is used in addition to phoneme based modelling. Both, grapheme and phoneme hypothesis are used in the decoding stage. The authors showed that adding grapheme based modelling to a phoneme-only system improved recognition accuracy.

III. BLSTM-CTC

Traditional feed-forward neural networks, such as multilayer perceptrons, are static classifiers that consider fixed-size input windows irrespective of surrounding context. This makes them poorly suited to transcribing connected time-series such as speech. Recurrent neural networks (RNN), where one or more of the hidden network layers is connected to itself, have proven more effective for speech transcription [2]. RNN can learn to model past events by adjusting the weights of the feedback connection, allowing them to make use of previous context.

However, analysis of the error flow in traditional RNN revealed that long-range context is inaccessible to them because the backpropagated error either blows up or decays over time (the vanishing gradient problem [15]). This led to the introduction of the Long Short-Term Memory (LSTM) RNN architecture, which is able to store information in linear memory cells over a long period of time. An LSTM hidden layer is composed of recurrently connected memory blocks, each of which contains one or more recurrently connected memory cells (see figure 1), along with three multiplicative "gate" units: the input, output, and forget gates.

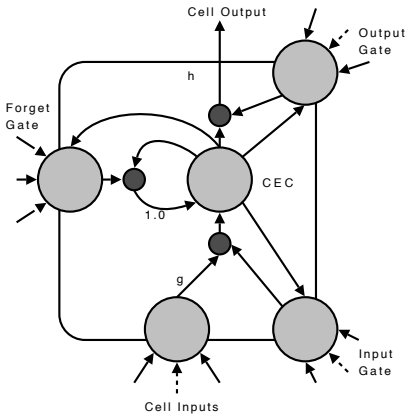

Fig. 1. LSTM memory block with one memory cell whose contents are protected and controlled by input, output and forget gates.

The cell input is multiplied by the activation of the input gate, the cell output by that of the output gate, and the previous cell values by the forget gate. The combined effect of the gates is to allow the network to store and retrieve information over long periods of time, thereby giving access to long-range context information. By adjusting the weights of the gates during training, the network learns "how much" context to consider, thus theoretically being able to learn even word level semantic context from the training transcriptions. However, this has not been precisely researched yet. The first few publications where LSTM networks are applied to phoneme recognition tasks show good results, which is an early indicator, that these networks are indeed able to model higher level context.

LSTM can be further extended to bidirectional networks, resulting in Bidirectional Long Short-Term Memory Recurrent Neural Networks (BLSTM-RNN) [16]. Here, two separate hidden layers are used to process the input data fowards and backwards. Both layers are connected to the same output layer, which can therefore make classifications based on both past and future context (see figure 2). When applied to speech recognition [2], [5], BLSTM-RNN has the advantage of being able to model anticipatory co-articulation effects. However, one major drawback is that the entire input sequence must be available beforehand, which makes on-line classification impossible. Yet, on-line classification is not required for many applications, such as off-line transcription of broadcast speech.

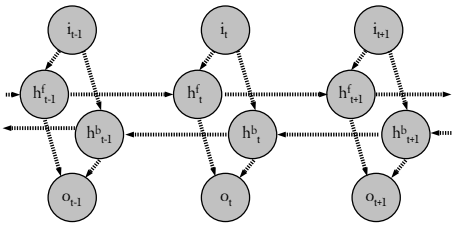

Fig. 2. Structure of a bidirectional network with input i, output o, as well as two hidden layers (h^f and h^b)

In the past, time-series transcription with RNN has been severely hampered by the fact that the target labels required for training have to be pre-aligned with the inputs. For phoneme recognition, this means that phonetic alignments must be provided for the training data. HMM-based systems perform alignments of the training sequences iteratively during training and thus do not have this limitation. Recently, a novel technique called Connectionist Temporal Classification (CTC) has been published [1], which allows RNN to be trained on unsegmented sequence data. The basic idea of CTC is to interpret the network outputs as a probability distribution over all possible label sequences, conditioned on a given input sequence. Given this probability distribution, an objective function can be derived that directly maximises the probabilities of the correct labellings. The objective function is differentiable and thus the network can be trained with standard backpropagation through time [17].

A more detailed description of the CTC algorithm can be found in [3]. A CTC output layer has $L + 1$ units, where L is the number of labels to be recognised (e.g. the total number of phonemes or graphemes). The additional output is required to specify a *blank* label, which will be output for

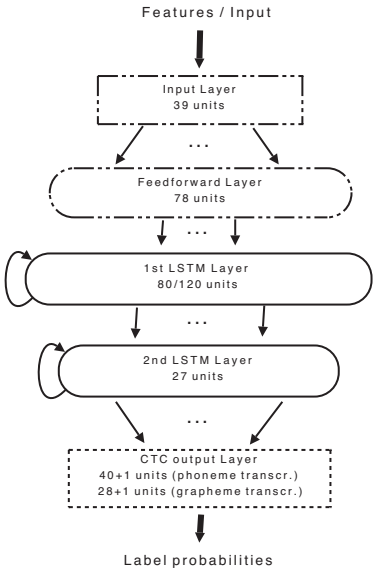

Fig. 3. Schematic topology of our multi-layer BLSTM-CTC networks used for grapheme transcription.

no label or an unknown label. At each timestep the $L + 1$ outputs corresponding to the L labels and the *blank* label are used to estimate the probabilities of observing the respective labels. The combined sequence of outputs estimates the joint probability of all possible alignments of all possible labellings with the input sequence. The total probability for a particular labelling can then be found by summing the probabilities of all the alignments corresponding to it [1]. By analogy with HMM, this process is referred as decoding. Using the CTC token passing algorithm [3], the decoding procedure can be extended to incorporate dictionaries and language models.

IV. Evaluation

To evaluate our proposed grapheme recognition approach we use the Wall-Street Journal 1 ($WSJ1$) corpus, which contains a large number of read English speech utterances spoken by American native English speakers. The $WSJ1$ subset used in this work contains 38,275 turns from the 1993 SI200 training set. The turns have an average length of 7.6 s, resulting in a total length of over 80 hours of speech material. For phoneme transcription experiments a phonemisation using 39 phonemes was obtained using the CMU Pronouncing Dictionary[1]. Thus, including an additional short pause label marking word boundaries, 40 labels are used for phoneme recognition. For grapheme recognition 28 labels are used, the 26 letters of the alphabet, the short pause (`SP`) label (for blanks between words), and a label for the apostrophe (e.g. in words like *it's*).

The label string output by the net is compared to the corresponding ground truth phonetic and graphemic transcriptions and a Label Error Rate (LER) is computed based on the number of insertions, substitutions and deletions according to the following equation:

$$LER = \frac{D - S - I}{N} \tag{1}$$

where D is the total number of deletions in the recognised strings, S is the total number of substitutions, I the total number of insertions, and N the total number of labels in the test set.

As acoustic low-level features 12 Mel-Frequency Cepstral Coefficients (MFCC) and log. energy (E) are used. Both are extracted every 10 ms over a window of size 25 ms. For HMM-based recognisers it is common practice to append δ and $\delta\delta$-coefficients to the MFCC. Although BLSTM-CTC theoretically are able to learn to internally generate the additional information gained by those coefficients, our experience has shown that feeding the BLSTM-CTC net those features directly reduces training time and gives slightly better results. Thus, we append δ and $\delta\delta$ coefficients resulting in 39 features in total. All features were normalised to have zero mean and unit variance using statistics computed from the training set.

We evaluate the proposed BLSTM-CTC on both grapheme and phoneme transcription tasks on $WSJ1$. Evaluation is performed using pre-defined speaker independent training, validation and test sets. The training set contains 27,570 utterances from 144 speakers. The validation set contains 3,059 utterances from 16 speakers, and the test set contains the 7,646 utterances from the remaining 40 speakers. Since a larger test set yields more stable and significant results we decided not to use the rather small standard evaluation sets, as used in [18], for example. The validation set is used to determine when training of the network should be aborted. After every 5 training epochs an error test on the validation set is conducted. If no improvement of LER on the validation or training set is observed for more than 10 error tests, the training is aborted. The best results reported are those obtained on the test set with the network giving the best LER on the validation set.

TABLE I
Four BLSTM-CTC hidden-layer topologies.

Topology	Description
NET_0	1 layer, 100 LSTM units, 1 cell each.
NET_1	1 layer, 150 LSTM units, 1 cell each.
NET_{0H}	1st layer, 78 feedforward units
	2nd layer, 80 LSTM units, 1 cell each.
	3rd layer, 27 LSTM units, 1 cell each.
NET_{1H}	1st layer, 78 feedforward units
	2nd layer, 120 LSTM units, 1 cell each.
	3rd layer, 27 LSTM units, 1 cell each.

Four BLSTM-CTC hidden layer topologies are investigated, as detailed in table I. Common to all networks is a feedforward input layer of size 39 (one input for each feature) as well as a CTC output layer of size 41 (including blank) for phoneme transcription networks and size 29 (including blank) for grapheme transcription networks. A novel architecture for both phoneme and grapheme recognition is the hierarchical

[1]http://www.speech.cs.cmu.edu/cgi-bin/cmudict

topology with three hidden layers as depicted in figure 3. For training of all the networks in our experiments standard backpropagation through time with a learn rate of 10^{-5} and a momentum of 0.9 is used.

V. Discussion of Results

Table II shows the results obtained on $WSJ1$ for grapheme and phoneme transcription with BLSTM-CTC after a fixed number of 100 training epochs.

TABLE II
Label Error Rate (LER) obtained on $WSJ1$. Four BLSTM-CTC topologies for both phoneme and grapheme transcription. Speaker independent evaluation using given train and test partitions. No. of training epochs fixed at 100.

[LER %]	Grapheme	Phoneme
NET_0	30.8	23.4
NET_1	26.6	19.9
NET_{0H}	22.1	17.5
NET_{1H}	**19.9**	**15.7**

Among the four network topologies great differences in transcription performance are observable. The multi-layer network topologies perform far better than the single layer topologies. I.e. approx. 6-8% absolute for grapheme recognition and 4-6% absolute for phoneme recognition. Networks with more units in the hidden-layer outperform smaller networks. Multiple layers might mimic the speech processing hierarchy found in human speech perception, i.e. acoustic modelling, word units, words, and grammar. HMM-based systems also employ a similar hierarchy by including acoustic models (HMM), dictionaries and language models. Other phoneme recognition approaches, e.g. [19] also implement multiple hierarchies. The benefit of BLSTM-CTC, however, is that an arbitrary number of layers can be specified, where each layer is able to automatically learn what and how much context to model. Hierarchical LSTM topologies are already successfully applied in other fields, e.g. handwriting recognition [3].

Overall it is clear that phoneme transcription yields slightly better results than grapheme transcription. Even after training the networks for more than 100 epochs, there is still a gap between phoneme and grapheme transcription results. However, we feel it is quite remarkable that BLSTM-CTC grapheme recognition performance is close to phoneme recognition performance, with no modification of the algorithm. The HMM-based systems for phoneme recognition and grapheme recognition must be tuned separately and show considerable conceptual differences.

The phoneme recognition result of 15.7% is the best obtained so far using BLSTM-CTC. 24.6% LER are reported on the TIMIT corpus in [5]. TIMIT - similarly to $WSJ1$ - contains read speech, however $WSJ1$ is larger than TIMIT, which seems to be the main reason for the boost in recognition performance. At the same time this shows that more training data is highly beneficial, which is not new and also applies to HMM-based systems.

Ground truth annotation:

```
AS OF APRIL FIRST ALL INTEREST
INCOME WILL BE TAXED AT TWENTY
PERCENT.
```

BLSTM-CTC transcribed grapheme string:

```
a s SP o f SP a i p r o l f i r s
t SP a l SP i n t e r e s t SP i
n c o m e SP w i l l SP b e SP t
a x t SP i t SP t w e n t i n SP
p e r c e n t SP
```

Final output after decoding:

AS OF APRIL FIRST AL **INTEREST INCOME WILL BE** TAX T. IT **TWENTY PERCENT.**

Fig. 4. $WSJ1$ example sentence 1. Average recognition case. Manual annotation (top), BLSTM-CTC grapheme output - SP denotes *short-pause*, i.e. a word boundary (middle), BLSTM-CTC grapheme output after dictionary based (unweighted, no language model) decoding (bottom), see section III. Correctly recognised words are printed in bold-face.

Ground truth annotation:

```
FINE ANSWERED HIS FRIEND JOHN
REILLY.
```

BLSTM-CTC transcribed grapheme string:

```
f i n SP a n s e r a g e s SP f r
SP a n j o n SP r i l y SP
```

Final output after decoding:

FINE AN SURGES FOR AN JON RILE.

Fig. 5. $WSJ1$ example sentence 2. Bad recognition case. Manual annotation (top), BLSTM-CTC grapheme output - SP denotes *short-pause*, i.e. a word boundary (middle), BLSTM-CTC grapheme output after dictionary based (unweighted, no language model) decoding (bottom), see section III. Correctly recognised words are printed in bold-face.

For the reader to get an impression of the types of recognition errors the networks commonly make, we provide the recognition result of two selected utterance from the test set in figures 4 (example 1) and 5 (example 2). Example 1 is an example of an average recognition result, where example 2 shows a result with more errors, below average. When looking at example 1 it can be seen that the words INTEREST and PERCENT are both correctly recognised on the grapheme level, even though the 'c' in PERCENT is pronounced as an 's', just like the 's' in INTEREST. This indicates that BLSTM-CTC networks are indeed very flexible and are able to model higher level context and thus being able to distinguish the different orthographies from the surrounding graphemes.

However, this does not seem to work as expected in all cases, e.g. if we look at APRIL in example 1, we see that it is transcribed as a i p r o l, which is pretty close to the phonetic representation. Such errors might be related to the lack of training data for specific words. This is a notable drawback of the proposed approach: Words occurring very seldom in the acoustic training data will not be learnt as well as more frequently occurring ones. It is not possible to train separate language and acoustic models as with HMM based approaches, where the language model can be trained from a far larger collection of texts and transcripts. However, when a large amount of transcribed acoustic training data is available, the presented approach is very easy to apply, and a recogniser can be built fully data-driven for any corpus and any language. Applying a language model on top of the BLSTM-CTC is another option. However, we believe that a phoneme based BLSTM-CTC system is better suited for decoding with a language model, since acoustically similar words can be very different on the grapheme level, not so on the phoneme level.

VI. Conclusion and Outlook

We have demonstrated a novel approach for recognition of large vocabulary read English speech, which is capable of recognising graphemes (i.e. letters) directly. No phonemisation dictionary is required, thus the approach is language independent and can be applied without modifications to any orthographically transcribed speech corpus. Recognition performance for graphemes was compared to phoneme recognition performance using the proposed approach. Phoneme recognition performance still remains superior to grapheme recognition performance, however, the gap is very small, especially for multi-layer BLSTM-CTC neural network topologies. These were found to be superior to single layer topologies on both tasks (phoneme and grapheme).

In future work we will investigate other topologies using more layers including sub-sampling feed-forward layers between the recurrent LSTM layers, as the results herein show that grapheme recognition benefits slightly more from hierarchical networks than phoneme recognition. We will also compare decoding results of grapheme and phoneme strings and report corresponding Word Error Rates. Furthermore, we plan to investigate a combined phoneme/grapheme BLSTM-CTC system to increase overall performance. Finally, we will investigate grapheme recognition on multi-lingual corpora and databases containing natural, spontaneous, and emotionally coloured speech.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 211486 (SEMAINE).

References

[1] A. Graves, S. Fernandez, F. Gomez, , and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the International Conference on Machine Learning, ICML*, Pittsburgh, USA, June 2006.

[2] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, June 2005.

[3] A. Graves, S. Fernandez, M. Liwicki, H. Bunke, and J. Schmidhuber, "Unconstrained online handwriting recognition with recurrent neural networks," in *Advances in Neural Information Processing Systems 20, NIPS 2008*, J. C. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA: MIT Press, January 2008.

[4] M. Wöllmer, F. Eyben, S. Reiter, B. Schuller, C. Cox, E. Douglas-Cowie, and R. Cowie, "Abandoning emotion classes - towards continuous emotion recognition with modelling of long-range dependencies," in *Proceedings Interspeech*, Brisbane, Australia, 2008.

[5] S. Fernandez, A. Graves, and J. Schmidhuber, "Phoneme recognition in timit with blstm-ctc," IDSIA, Tech. Rep., 2008.

[6] ——, "An application of recurrent neural networks to discriminative keyword spotting," in *Proceedings of ICANN*, Porto, Portugal, 2007, pp. 220–229.

[7] A. Graves, S. Fernandez, and J. Schmidhuber, "Bidirectional lstm networks for improved phoneme classification and recognition," in *Proceedings of ICANN*, vol. 18, Warsaw, Poland, 2005, pp. 602–610.

[8] P. Charoenpornsawat, S. Hewavitharana, and T. Schultz, "Thai grapheme-based speech recognition," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL*. New York, NY, USA: Association for Computational Linguistics, June 2006, pp. 17–20.

[9] M. Killer, S. Stüker, and T. Schultz, "Grapheme based speech recognition," in *Proceedings of the 8th European Conference on Speech Communication and Technology , Eurospeech 2003, Genf, Schweiz*, September 2003.

[10] S. Stüker and T. Schultz, "A grapheme based speech recognition system for russian," in *Proceedings of the 9th International Conference "Speech And Computer" SPECOM'2004*. Saint-Petersburg, Russia: Anatolya, September 2004, pp. 297–303.

[11] S. Kanthak and H. Ney, "Context-dependent acoustic modeling using graphemes for large vocabulary speech recognition," in *Proceedings the 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'02)*, vol. 1. Orlando, Florida, USA: IEEE, 2002, pp. 845–848.

[12] E. G. Schukat-Talamazzini, H. Niemann, W. Eckert, T. Kuhn, and S. Rieck, "Automatic speech recognition without phonemes," in *Proceedings of the 3rd European Conference on Speech Communication and Technology EUROSPEECH'93*. Berlin, Germany: ISCA, September 1993, pp. 129–132.

[13] B. Mimer, S. Stüker, and T. Schultz, "Flexible decision trees for grapheme based speech recognition," in *Elektronische Sprachsignalverarbeitung Tagungsband der 15. Konferenz*, ser. Studientexte zur Sprachkommunikation, no. 30. Cottbus, Germany: w.e.b. Universitätsverlag, September 2004, pp. 79–86.

[14] M. M. Doss, T. A. Stephenson, H. Bourlard, and S. Bengio, "Phoneme-grapheme based speech recognition system," *Automatic Speech Recognition and Understanding, 2003. ASRU '03. 2003 IEEE Workshop on*, pp. 94–98, 2003.

[15] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," in *A Field Guide to Dynamical Recurrent Neural Networks.*, S. C. Kremer and J. F. Kolen, Eds. IEEE Press, 2001.

[16] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, pp. 2673–2681, November 1997, introduction of BLSTM.

[17] P. Werbos, "Backpropagation through time: What it does and how to do it," *Proceedings of the IEEE*, vol. 78, pp. 1550–1560, 1990.

[18] K. Vertanen, "Baseline wsj acoustic models for htk and sphinx: Training recipes and recognition experiments," University of Cambridge, Cavendish Laboratory, Madingley Road, Cambridge, CB3 0HE, UK, Tech. Rep., 2006.

[19] O. Dekel, J. Keshet, and Y. Singer, "Online algorithm for hierarchical phoneme classification," in *Workshop on Multimodal Interaction and Related Machine Learning Algorithms*, Martigny, Switzerland, 2004, pp. 146–159.