RWTH ASR Systems for LibriSpeech: Hybrid vs Attention

Christoph Lüscher¹, Eugen Beck¹,², Kazuki Irie¹, Markus Kitza¹, Wilfried Michel¹,², Albert Zeyer¹,², Ralf Schlüter¹, Hermann Ney¹,²

¹Human Language Technology and Pattern Recognition, Computer Science Department, RWTH Aachen University, 52074 Aachen, Germany
²AppTek GmbH, 52062 Aachen, Germany
{luescher, beck, irie, kitza, michel, zeyer, schlueter, ney}@cs.rwth-aachen.de

Abstract

We present state-of-the-art automatic speech recognition (ASR) systems employing a standard hybrid DNN/HMM architecture compared to an attention-based encoder-decoder design for the LibriSpeech task. Detailed descriptions of the system development, including model design, pretraining schemes, training schedules, and optimization approaches are provided for both system architectures. Both hybrid DNN/HMM and attention-based systems employ bi-directional LSTMs for acoustic modeling/encoding. For language modeling, we employ both LSTM and Transformer based architectures. All our systems are built using RWTH’s open-source toolkits RASR and RETURNN. To the best knowledge of the authors, the results obtained when training on the full LibriSpeech training set, are the best published currently, both for the hybrid DNN/HMM and the attention-based systems. Our single hybrid system even outperforms previous results obtained from combining eight single systems. Our comparison shows that on the LibriSpeech 960h task, the hybrid DNN/HMM system outperforms the attention-based system by 15% relative on the clean and 40% relative on the other test sets in terms of word error rate. Moreover, experiments on a reduced 100h-subset of the LibriSpeech training corpus even show a more pronounced margin between the hybrid DNN/HMM and attention-based architectures.

Index Terms: speech recognition, hybrid BLSTM/HMM, attention, LibriSpeech

1. Introduction

Over the last years, automatic speech recognition (ASR) systems have improved significantly. Especially the rise of deep neural networks (NNs) has accelerated this development immensely [1]. Convolutional NNs and recurrent NNs are the state-of-the-art architectures for most ASR tasks. State-of-the-art systems are largely based on the hybrid deep neural network (DNN) based standard architectures. However, the general progress in deep learning/machine learning also triggered a diversification of ASR architectures into a series of so-called end-to-end approaches. Most notably, this includes the attention-based encoder-decoder architecture, which for good performance has been reported on a number of tasks, including the LibriSpeech task.

The LibriSpeech task comprises English read speech data based on the LibriVox project [2]. Previous results on LibriSpeech using hybrid models are presented in [3, 4]. While [3] uses a Gaussian mixture model/hidden Markov model (GMM/HMM) as the basis for their system, further training is conducted with a hybrid DNN/HMM with a densely connected topology. The densely connected NNs in [3] are composed of different types of NN layers: convolutional NNs, time delay neural network and bi-directional long short-term memories (LSTMs). Lattice-free maximum mutual information (LF-MMI) is applied during training. A recurrent NN language model (LM) is used for rescoring. The final best result in [3] is achieved with a system combination of eight systems. In [4], a lattice-free state-level minimum bayes risk (sMBR) training method is used.

End-to-end results on LibriSpeech were presented in [5–9]. The end-to-end approach in [8] uses the raw waveform and a convolutional NN acoustic model with gated linear units. An end-to-end attention-based encoder-decoder approach with a pretraining scheme is presented in [5, 6]. In [7] a training procedure based on edit distance for sequence to sequence model optimization is presented. An exploration of target units (phoneme, grapheme and word-piece) in relation to training size was performed in [9]. A data augmentation method called SpecAugment was presented in [10]. So far, while end-to-end approaches show competitive performance, they are outperformed by hybrid approaches. We compare the conventional hybrid DNN/HMM approach on phone level to the encoder-decoder-attention model which directly operates on the word or sub-word level and is thus often referred to as an end-to-end model. In addition, we use word-level and subword-level neural language models to further improve the performance of both systems. We describe the development of our hybrid system and show which factors were especially important for the performance of the system. To the best of our knowledge, the results obtained on the LibriSpeech task reflect state-of-the-art performance for both hybrid and attention-based modeling, with a clear margin still for hybrid DNN/HMM modeling when no data augmentation scheme is applied.

2. Hybrid system

2.1. Acoustic model

2.1.1. GMM/HMM system

We use 16-dim Mel frequency cepstral coefficients (MFCC) adding first and second order derivatives, and additionally energy features as input for the GMM/HMM system. The transition probabilities are set manually and applied to all HMMs. The first step is linear time alignment where the features are uniformly distributed over the audio. We iterate the repetition of the parameter estimation based on the linear time alignment five times. Afterwards, we perform a non-linear time alignment to improve the alignment. Afterwards we perform parameter estimation. Initially, this process was iterated 10 times. Increasing the number of iterations showed constant improvement. Therefore we continued adding training iterations until word error rate (WER) convergence. Training of a state-
tied triphone GMM/HMM model is the following step. The states are tied using a phonetic classification and regression tree (CART). We experimented with different numbers of CART labels ranging from 9 – 200 plus silence. All state-tied triphones use three HMM states. We switch the input features from 16-dim. MFCC with derivatives to a context window of MFCC features resulting in a 144-dim. feature vector on which linear discriminant analysis (LDA) is performed. The LDA output has a dimension of 48. After training the state-tied triphone GMM/HMM model, vocal tract length normalization (VTLN) is applied, followed by speaker adaptive training (SAT) with constrained maximum likelihood linear regression to adapt the Gaussian mixture model parameters to a speaker. After adapting the parameters, a realignment is performed.

2.1.2. Hybrid DNN/HMM system

The NN acoustic model architecture is a bi-directional LSTM [11, 12]. This architecture achieves good performance in acoustic modelling [12, 13]. For the hybrid DNN/HMM system we extract several different features: 16-dim MFCC with derivatives and energy, 48-dim. features from the triphone system and Gammatone filters [14] with 25, 50 or 100-dim. The extracted 50-dim. Gammatone filters had the best performance. All features are used as input into the bi-directional LSTM along with the generated alignments from the GMM/HMM system. We continue to use CART labels for the state-tied phones. The same range of CART labels was used for experimentation. The network topology consists of six bi-directional LSTM layers with 1000 units for backward and forward direction each. We experimented with smaller bi-directional LSTM sizes (number of layers and number of units per layer) but found them to be worse in performance. The output layer is comprised of a softmax layer with output units corresponding to the number of the CART labels. Frame-wise cross-entropy loss criterion is applied, followed by speaker adaptive training (SAT) with constrained maximum likelihood linear regression to adapt the Gaussian mixture model parameters to a speaker. After adapting the parameters, a realignment is performed.

2.2. Language model

We report performance of hybrid systems using both a 4-gram count based language model [20] and an LSTM language model [21] in the first pass decoding [22]. We use the 4-gram count model officially distributed with the LibriSpeech dataset [2]. For the LSTM language model, we train our own model using our toolkit RETURNN [23]. Two training datasets are available for language modeling: 800M-word text only data and 960B of audio transcriptions which corresponds to 10M-word text data. These two sets are merged to form one training dataset for language model training. Our LSTM language model has two recurrent layers with 4096 LSTM nodes in each layer, an input projection layer of size 128, and a output softmax layer over the full 200k vocabulary. We train the model using the stochastic gradient descent with gradient norm clipping and Newbob learning rate scheduling.

In addition, we carry out rescoring of lattices generated by the LSTM language model using a Transformer [24] language model. Our Transformer model has 96 layers with the self-attention total dimension of 512 using 8 heads and the inner feed-forward dimension of 2048 in each layer, which gave the best development perplexity in our preliminary experiments [25]. We use push-forward algorithm [26] with recombination pruning of order 9. We linearly interpolate the two models with interpolation weights optimized on the development perplexity. We found 0.71 to be the optimal weight on the Transformer model which gave the development perplexity of 52.3, while the LSTM and Transformer models have the individual development perplexity of 60.2 and 53.7 respectively.

3. Encoder-Decoder-Attention system

The encoder-decoder framework with attention has initially been introduced for machine translation where it dominates the field now [27–29]. Recent investigations have shown promising results by applying the same approach for speech recognition [5, 7, 30–33]. Among end-to-end approaches for ASR, the attention model seems to perform best [6]. Our model operates on sub-word units via byte-pair encoding [34]. As input 40-dim MFCC feature vectors are used. Our presented results outperform the best LibriSpeech attention system presented in [6]. Compared to the system in [6] we use an extended pretraining variant where we not only grow the encoder depth but also grow the hidden dimension of the LSTMs. Specifically, we start with 2 layers in the encoder of dimension 512 and increase to 6 layers with dimension 1024. Additionally, we train the first pretrain construction step first without dropout. We improved upon that model by tuning the curriculum learning schedule slightly, i.e. we have these 4 steps with different portions of the dataset:

1. from 25% of the whole data, take only train-clean, and filter randomly such that the max mean number of characters in the transcriptions of each sequence is 50,
2. from the next 25% of the whole data, take only train-clean, and filter randomly such that the max mean number of characters is 75,
3. from the next 50% of the whole data, take only train-clean,
4. from now on, take everything.

Also, in the pretraining, we repeat the first step once more, with 2 layers of dimension 512, without dropout. The next improvement came from just training longer, i.e. we trained with our learning rate scheduling until it converged, then took the best
model, and continued training with a reset learning rate scheduling. We repeated this twice. In the first iteration, we went over the whole data 12.5 times, then another 6.0 times and finally another 8.3 times, i.e. in total 27.4 times.

To further enhance end-to-end system’s performance, we train byte-pair encoding (BPE)-level language models and apply them to the system by shallow fusion [35, 36]. We report the performance of LSTM based and Transformer based language models separately. Our LSTM model has 4 recurrent layers with 2048 LSTM nodes. We use a 24-layer Transformer model with 8-head self-attention and feed-forward dimensionality of 1024 and 4096 respectively, which we obtained in [25]. We select the language model checkpoints for the recognition experiments based on the development perplexity. For shallow fusion, we apply a single weight on the language model score (the weight on the score of the attention model is 1) and we use a beam size of 64 as well as an end-of-sentence penalty [37]. We optimize the weights separately on the dev-clean and dev-other sets, then respectively apply them to the test-clean and test-other sets. We found optimal weights to be similar for both models; 0.5 and 0.56 for the LSTM language model, and 0.52 and 0.54 for the Transformer model, respectively on the clean and other sets.

### 4. Experimental setup

The two systems, a hybrid-DNN and an attention-based encoder-decoder are both trained on the 960h training data from the LibriSpeech corpus. For comparison, also a 100h subset is used. Unless specified otherwise, the training was performed using the full training set of 960h. The data is in English but the content ranges from different time periods and different English speaking countries. Having the consequence of different English styles being within the corpus.

The hybrid model was trained and decoded with RASR [38] and RETURNN [23, 39]. The monophone and triphone system to generate the alignments was built in RASR while the NN model was trained in RETURNN. The decoding process was setup in RASR. Our encoder-decoder-attention model was trained and decoded using RETURNN [23]. Both toolkits are open-source. All the config files used for training and recognition of all our results are publicly available online [40].

We evaluate the models on the dev and test sets provided with the LibriSpeech corpus: dev-clean, dev-other, test-clean and test-other. The difference between clean and other is the quality of the audio and its corresponding transcription. The clean quality is higher than the other.

### 5. Experimental results

The development stages of our acoustic model are shown in Table 1. We start the training of the GMM/HMM model from scratch using linear alignments. Afterwards we utilize non-linear alignments. To further improve the GMM/HMM model we introduce triphones. Adding VTLN on top of the triphone system only shows improvements on clean but degradation on other. However adding SAT to the triphone system improves the WER. Combining VTLN and SAT gives mixed WER: clean improves, other degrades. Introducing an hybrid DNN/HMM improves the system WER results. Continuing with sequence discriminative training improves the performance even further.

#### Table 1: GMM/HMM and hybrid DNN/HMM results on LibriSpeech with 12k CART labels and evaluated with the official 4-gram LM.

<table>
<thead>
<tr>
<th>phoneme context</th>
<th>acoustic model</th>
<th>VTLN</th>
<th>SAT</th>
<th>sMBR</th>
<th>WER [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>mono</td>
<td>GMM</td>
<td>no</td>
<td>no</td>
<td></td>
<td>24.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>no</td>
<td></td>
<td>12.1</td>
</tr>
<tr>
<td>tri</td>
<td>LSTM</td>
<td>yes</td>
<td>-</td>
<td>no</td>
<td>8.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>yes</td>
<td>yes</td>
<td></td>
<td>4.0</td>
</tr>
</tbody>
</table>

#### Table 2: Hybrid DNN/HMM results on LibriSpeech with different numbers of CART labels. For all systems the official 4-gram word LM is used.

<table>
<thead>
<tr>
<th># of CART labels</th>
<th>dev</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>clean</td>
<td>other</td>
</tr>
<tr>
<td>9001</td>
<td>6.2</td>
<td>14.9</td>
</tr>
<tr>
<td>12001</td>
<td>4.0</td>
<td>9.6</td>
</tr>
<tr>
<td>20001</td>
<td>4.9</td>
<td>11.3</td>
</tr>
</tbody>
</table>

#### Table 3: Comparison between hybrid DNN/HMM and encoder-decoder-attention model results on LibriSpeech with different training corpus sizes. train-clean-100 is a official subset of the training corpus. train-960 is the complete training corpus. (Clustered) context-dependent phones (CDPs) are utilized for the hybrid model, and sub-word units for the attention model.

<table>
<thead>
<tr>
<th>training set</th>
<th>model</th>
<th>LM</th>
<th>WER [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>clean</td>
<td>other</td>
</tr>
<tr>
<td>train-clean-100</td>
<td>hybrid</td>
<td>4-gram</td>
<td>5.0</td>
</tr>
<tr>
<td></td>
<td>attention</td>
<td>none</td>
<td>14.7</td>
</tr>
<tr>
<td>train-960</td>
<td>hybrid</td>
<td>4-gram</td>
<td>4.0</td>
</tr>
<tr>
<td></td>
<td>attention</td>
<td>none</td>
<td>4.7</td>
</tr>
</tbody>
</table>

We compare the hybrid model with the encoder-decoder-attention model. We trained both models on the train-clean-100 training subset and on the train-960 complete training set.
These are not the best models but utilize a baseline model for both approaches. The hybrid DNN/HMM model outperforms the encoder-decoder-attention model constantly. But the difference in performance shrinks substantially with the much larger training set.

Our encoder-decoder-attention model in combination with a Transformer LM gives a WER of 3.2% on test-clean and 9.9% on test-other (Table 4). Evaluating our sequence discriminatively trained acoustic model with our LSTM LM results in a WER of 2.6% on test-clean and 5.5% on test-other. Rescoring with a Transformer language model further improves the performance of our hybrid DNN/HMM system resulting in a WER of 2.3% on test-clean and 5.0% on test-other. The previous best hybrid system was presented in [3] while the best end-to-end system without data augmentation was presented in [8, 9] (Table 4). Additionally we present the best end-to-end system with data augmentation [10]. Our best encoder-decoder-attention model improves the state-of-the-art for end-to-end models without data augmentation by 17.6% relative WER on test-clean and by 3.9% relative WER on test-other. Our best hybrid DNN/HMM system without Transformer LM rescoring improves the state-of-the-art by 18.8% relative WER on test-clean and by 27.0% relative WER on test-other. If we add rescoring with a Transformer LM we improve further by 11.5% relative WER on test-clean and by 9.1% relative WER on test-other. In comparison, the hybrid DNN/HMM system still outperforms the encoder-decoder-attention system by over 15% relative WER on test-clean and by over 40% relative WER on test-other. Our best hybrid model even outperforms the end-to-end attention model with SpecAugment [10] by 8% relative WER on test-clean and by 13.8% relative WER on test-other. These results reflect the state-of-the-art performance for both hybrid and attention-based models on LibriSpeech, to the best of the authors’ knowledge.

WERs become very small, especially for dev-clean and test-clean. When analyzing the errors, it is noticeable that some of the errors would not be recognized as primary errors by a human. These can be categorized as, for example: word contractions or American vs British English spelling. Examples of such errors are: I am ↔ I’m, tyrannise ↔ tyrannize, color ↔ colour, oh ↔ o. So far we have not employed a normalization strategy for these errors.

6. Conclusions

In this paper we presented two ASR systems for the LibriSpeech task. One System was a hybrid DNN/HMM system based on a GMM/HMM system, the other system was an attention-based encoder-decoder system.

We described how we built the systems and described how to incrementally improve the systems to get competitive results. For the hybrid DNN/HMM system a large NN acoustic model, the sequence discriminative training and the employment of an LSTM LM was important for the good performance. The encoder-decoder-attention approach utilized an extended pre-training variant and a tuned curriculum learning schedule. This enabled the model to achieve competitive results in comparison to other end-to-end approaches.

The presented encoder-decoder-attention system showed state-of-the-art performance on the LibriSpeech 960h task in comparison with end-to-end systems without data augmentation. But our comparison shows that on the LibriSpeech 960h task, the hybrid DNN/HMM system outperforms the attention-based system by 15% relative on the clean and 40% relative on the other test sets. Our hybrid system even outperforms previous results presented in the literature. Moreover, experiments on a reduced 100h-subset of the LibriSpeech training corpus even show a more pronounced margin between the hybrid DNN/HMM and attention-based architectures. To the best knowledge of the authors, the results obtained when training on the full LibriSpeech training set, are the best published currently, both for the hybrid DNN/HMM and the attention-based systems presented in this work.

Acknowledgements

This work has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 694537, project “SEQCLAS”) and from a Google Focus Award. The work reflects only the authors’ views and none of the funding parties is responsible for any use that may be made of the information it contains.

Experiments were partially performed with computing resources granted by RWTH Aachen University under project nova0003.

We thank Wei Zhou for help with generating lattices.

Table 4: The WER results from our most interesting models and important results from other papers on LibriSpeech 960 h. CDp are (clustered) context-dependent phones. BPE are sub-word units. 4-gr LM is the official 4-gram word LM. GCNN are gated convolutional NN. RNN are recurrent NN.

<table>
<thead>
<tr>
<th>paper</th>
<th>model</th>
<th>label unit</th>
<th>LM</th>
<th>WER [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>AM</td>
<td>dev</td>
</tr>
<tr>
<td>Han et al. [3]</td>
<td>hybrid, seq. disc., single</td>
<td>CDp</td>
<td>RNN</td>
<td>3.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>LSTM</td>
<td>3.3</td>
</tr>
<tr>
<td>Zeghidour et al. [8]</td>
<td>end-to-end GCNN</td>
<td>chars</td>
<td>GCNN</td>
<td>3.2</td>
</tr>
<tr>
<td>Irie et al. [9]</td>
<td>end-to-end attention</td>
<td>BPE</td>
<td>LSTM</td>
<td>3.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>None</td>
<td>4.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>LSTM</td>
<td>2.9</td>
</tr>
<tr>
<td>Zeyer et al. [5]</td>
<td>end-to-end attention</td>
<td>BPE</td>
<td>Transformer</td>
<td>4.6</td>
</tr>
<tr>
<td></td>
<td>hybrid</td>
<td>CDp</td>
<td>4-gr</td>
<td>4.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+ LSTM</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Transformer resc.</td>
<td>1.9</td>
</tr>
<tr>
<td>Park et. al. [10]</td>
<td>end-to-end attention/SpecAugment</td>
<td>Word Piece Model</td>
<td>LSTM</td>
<td>-</td>
</tr>
</tbody>
</table>
References


[34] R. Sennrich, B. Haddow, and A. Birch, “Neural machine translation of rare words with subword units,” in ACL, Berlin, Germany, August 2016.


