Speech Recognition

Abstract

Current state-of-the-art speech recognition systems build on recurrent neural networks for acoustic and/or language modeling, and rely on feature extraction pipelines to extract mel-filterbanks or cepstral coefficients. In this paper we present an alternative approach based solely on convolutional neural networks, leveraging recent advances in acoustic models from the raw waveform and language modeling. This fully convolutional approach is trained end-to-end to predict characters from the raw waveform, removing the feature extraction step altogether. An external convolutional language model is used to decode words. On Wall Street Journal, our model matches the current state-of-the-art. On Librispeech, we report state-of-the-art performance among end-to-end models, including Deep Speech 2, that was trained with 12 times more acoustic data and significantly more linguistic data.

Index Terms: Speech recognition, end-to-end, convolutional, language model, waveform

1. Introduction

Recent work on convolutional neural network architectures shows they are competitive with recurrent architectures even on tasks where modeling long-range dependencies is critical, such as language modeling, machine translation and speech synthesis. In end-to-end speech recognition however, recurrent neural networks are still prevalent for acoustic and/or language modelling. There is a history of using convolutional networks in speech recognition, but only as part of an otherwise more traditional pipeline. They were first introduced as TDNNs to predict phoneme classes , and later to generate HMM posterior- grams. They have recently been used in end-to-end systems, but only in combination with recurrent layers, or n-gram language models , or for phone recognition. Convolutional architectures are prevalent when learning from the raw waveform , because they naturally model the computation of standard features such as mel-filterbanks. Given the evidence that convolutional networks are also suitable on long-range dependency tasks, we expect them to be competitive at all levels of the speech recognition pipeline. In this paper, we present a fully convolutional approach to end-to-end speech recognition. Building on recent advances in convolutional learnable front-ends for speech, convolutional acoustic models [12], and convolutional language models, our model is a deep convolutional network that takes the raw waveform as input and is trained end-to-end to predict letters. Sentences are then predicted using beam-search decoding with a convolutional language model.

In addition to presenting the first application of convolutional language models to speech recognition, the main contribution of the paper is to show that fully convolutional architectures achieve state-of-the-art performance among end-to-end systems. Thus, our results challenge the prevalence of recurrent architectures for speech recognition, and they parallel the prior results on other application domains that convolutional architectures are on par with recurrent ones.
More precisely, we perform experiments on the large vocabulary task of the Wall Street Journal dataset (WSJ) and on the 1000h Librispeech. Our overall pipeline improves the state-of-the-art of end-to-end systems on both datasets. In particular,we decrease by 2% (absolute) the Word Error Rate on the noisytest set of Librispeech compared to DeepSpeech 2 and the best sequence-to-sequence model. On clean speech, the improvement is about 0.5% on Librispeech compared to the best end-to-end systems; on WSJ, our results are competitive with

the current state-of-the-art, a DNN-HMM system. In particular, the detailed results show that the convolutional language model yields systematic and consistent improvement over a 4-gram language model for its better perplexity and larger receptive field. In addition, we complement the promising results of [18] regarding the performance of learning the front-end of speech recognition systems: first, we show that learning the front-end yields substantial improvements on noisy speech compared to a mel-filterbanks front-end. Second, we show additional improvements on both WSJ and Librispeech by varying the number of filters in the learnable front-end, leading to a 1.5% absolute decrease in WER on the noisy test set of Librispeech. Our results are the first in which an end-to-end system trained on the raw waveform achieves state-of-the-art performance (among all end-to-end systems) on two datasets.

2. Model

Our approach, described in this section, is illustrated in Fig. 1.2.1. Convolutional Front-end Several proposals to learn the front-end of speech recognition systems have been made Following the comparison, we consider their best architecture, called ”scattering based” (hereafter referred to as learnable front-end). The learnable front-end contains first a convolution of width 2 that emulates the pre-emphasis step used in mel-filterbanks. It is followed by a complex convolution of width 25ms and k filters. After taking the squared absolute value, a low-pass filter of width 25ms and stride 10ms performs decimation. The front end finally applies a log-compression and a per-channel mean-variance normalization (equivalent to an instance normalization layer [20]). Following [18], the ”pre-emphasis” convolution is initialized to [−0.97; 1], and then trained with the rest of the

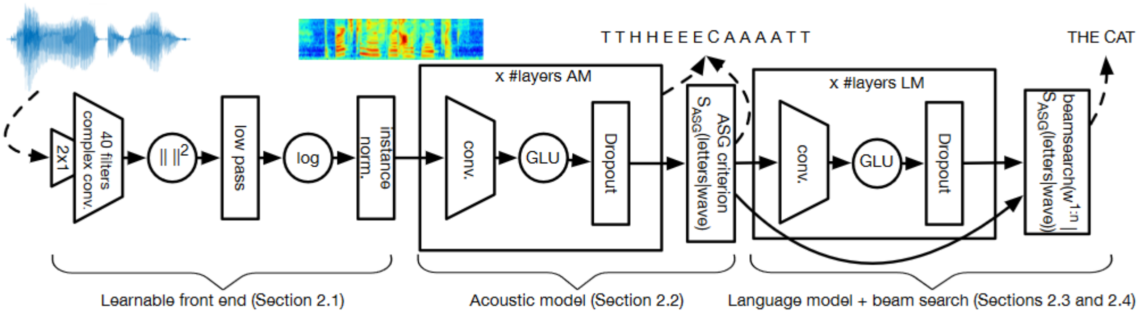

TTHHEEECAAAATT
THE CAT
2x1
40 filters complex conv.
$|| ||^2$
low pass
log
instance norm.
x #layers AM
conv.
GLU
Dropout
ASG criterion S_{ASG}(letters|wave)
x #layers LM
conv.
GLU
Dropout
beamsearch($w^{1:n}$ | S_{ASG}(letters|wave))
Learnable front end (Section 2.1)
Acoustic model (Section 2.2)
Language model + beam search (Sections 2.3 and 2.4)

Figure 1: *Overview of the fully convolutional architecture.*

network. The low-pass filter is kept constant to a squared Hanning window, and the complex convolutional layer is initialized randomly. In addition to the k = 40 filters used by [18], we experiment with k = 80 filters. Notice that since the stride is the same as for mel-filterbanks, acoustic models on top of the learnable front-ends can also be applied to mel-filterbanks (simply modifying the number of input channels if k 6 = 40).

2.1 Methodology

Speech recognition, a critical component of natural language processing, involves the development of systems capable of converting spoken language into written text. This paper provides an overview of the key methodologies employed in the field. The process begins with the collection of diverse audio datasets, representing a spectrum of linguistic and environmental variables. Preprocessing techniques, including signal normalization and feature extraction using methods like Mel Frequency Cepstral Coefficients (MFCCs), prepare the raw audio for analysis.

Acoustic modeling, utilizing approaches such as Hidden Markov Models (HMMs) or deep neural networks (DNNs), maps acoustic features to phonemes or sub-word units. Concurrently, language modeling incorporates contextual dependencies through n-gram models or recurrent neural networks (RNNs). The decoding phase employs algorithms like Viterbi decoding to determine the most probable word sequences based on the combined acoustic and language models.

Training and optimization iteratively refine models using labeled data, employing techniques such as gradient descent and backpropagation for deep learning models. The evaluation phase assesses system performance through metrics, ensuring accuracy and efficiency in transcribing spoken language. This holistic methodology forms the foundation for advancing robust and effective speech recognition systems, facilitating applications across diverse domains.

2.2 Neural networks

Neural networks play a pivotal role in advancing the state-of-the-art in speech recognition, providing a powerful framework to model complex patterns within audio data. One key application involves deep neural networks (DNNs) for acoustic modeling, where layers of interconnected nodes learn hierarchical representations of input features, such as Mel Frequency Cepstral Coefficients (MFCCs). These models excel at capturing intricate relationships in speech signals, enabling more accurate mapping of acoustic features to phonetic units.

Recurrent Neural Networks (RNNs) are particularly adept at modeling sequential dependencies, making them valuable in language modeling for speech recognition. By considering the contextual relationships between words, RNNs enhance the understanding of spoken language nuances, contributing to improved transcription accuracy.

2.3. Convolutional Acoustic Model

The acoustic model is a convolutional neural network with gated linear units [1], which is fed with the output of the learnable front-end. As in [12], the networks uses a growing number of channels, and dropout [21] for regularization. These acoustic models are trained to predict letters directly with the Auto Segmentation Criterion (ASG) [22]. The ASG criterion is similar to CTC [5] except that it adds input-independent transition scores between letters. The depth, number of feature maps per layer, receptive field and amount of dropout of models on each dataset are adjusted individually based on the amount of training data.

2.4. Convolutional Language Model

The convolutional language model (LM) is the GCNN-14B from [1], which achieved competitive results on language modeling benchmarks with similar size of vocabulary and training data to the ones used in our experiments. The network contains 14 convolutional residual blocks [23], and this deep architecture gives us a large enough receptive field. In each residual block, two 1 × 1 1-D convolutional layers are placed at the beginning and the end serving as bottlenecks for computational efficiency. Gated linear units are used as activation functions. We use the language model to score candidate transcriptions in addition to the acoustic model in the beam search decoder described in the next section. Compared to n-gram LMs, convolutional LMs allow the decoder to look at longer context with better perplexity.

2.5. Beam-search decoder

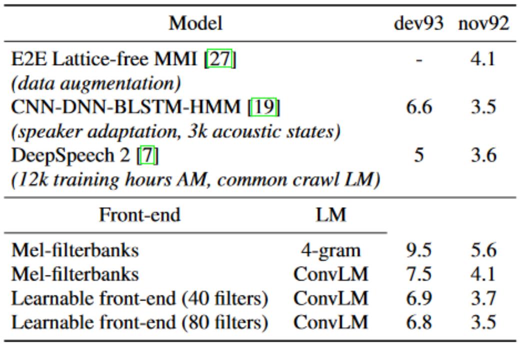
We use the beam-search decoder presented in [12] to generate word sequences given the output from our acoustic model. Given input X to the acoustic model, the decoder finds the word transcription W which maximizes:

$$AM(W|X)+\alpha\log Plm(W)+\beta|W|-\gamma|\{i|\pi i=\langle sil\rangle\}|, (1)$$

where π is a path representing a valid sequence of letters for W and πi is the i-th letter in this sequence. The score of the acoustic model is computed based on the score of paths of letters (including silences) that are compatible with the output se-quence. Denoting by Gasg the corresponding graph, the score of a sequence of words W given by the accoustic model.

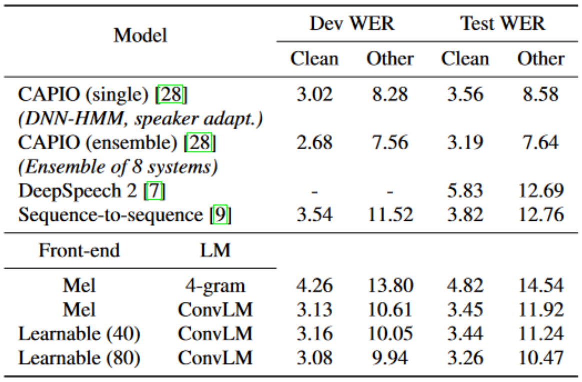
3. Experiments

We evaluate our approach on the large vocabulary task of the Wall Street Journal (WSJ) dataset, which contains 80 hours of clean read speech, and Librispeech, which contains 1000 hours with separate train/dev/test splits for clean and noisy speech. Each dataset comes with official textual data to train language models, which contain 37 million tokens for WSJ, 800 million tokens for Librispeech. Our language models are trained separately for each dataset on the official text data only. These datasets were chosen to study the impact of the different components of our system at different scales of training data and in different recording conditions.The models are evaluated in Word Error Rate (WER). Our experiments use the open source codes of wav2letter1 for the acoustic model, and fairseq2 for the language model. More details on the experimental setup are given below.Baseline Our baseline for each dataset follows. It uses the same convolutional acoustic model as our approach but a mel-filterbanks front-end and a 4-gram language model.



Model		dev93	nov92
E2E Lattice-free MMI [27] *(data augmentation)*		-	4.1
CNN-DNN-BLSTM-HMM [19] *(speaker adaptation, 3k acoustic states)*		6.6	3.5
DeepSpeech 2 [7] *(12k training hours AM, common crawl LM)*		5	3.6
Front-end	LM		
Mel-filterbanks	4-gram	9.5	5.6
Mel-filterbanks	ConvLM	7.5	4.1
Learnable front-end (40 filters)	ConvLM	6.9	3.7
Learnable front-end (80 filters)	ConvLM	6.8	3.5

Table 1: *WER (%) on the open vocabulary task of WSJ.*



Model		Dev WER		Test WER	
		Clean	Other	Clean	Other
CAPIO (single) [28] *(DNN-HMM, speaker adapt.)*		3.02	8.28	3.56	8.58
CAPIO (ensemble) [28] *(Ensemble of 8 systems)*		2.68	7.56	3.19	7.64
DeepSpeech 2 [7]		-	-	5.83	12.69
Sequence-to-sequence [9]		3.54	11.52	3.82	12.76
Front-end	LM				
Mel	4-gram	4.26	13.80	4.82	14.54
Mel	ConvLM	3.13	10.61	3.45	11.92
Learnable (40)	ConvLM	3.16	10.05	3.44	11.24
Learnable (80)	ConvLM	3.08	9.94	3.26	10.47

Table 2: *WER (%) on Librispeech.*

4. Results

4.1. Word Error Rate results

Table 1 shows Word Error Rates (WER) on WSJ for the current state-of-the-art and our models. The current best model trained on this dataset is an HMM-based system which uses a combination of convolutional, recurrent and fully connected layers, as well as speaker adaptation, and reaches 3.5% WER on nov92. DeepSpeech 2 shows a WER of 3.6% but uses 150 times more training data for the acoustic model and huge text datasets for LM training. Finally, the state-of-the-art among end-to-end systems trained only on WSJ, and hence the most comparable to our system, uses lattice-free MMI on augmented data (with speed perturbation) and gets 4.1% WER. Our baseline system, trained on mel-filterbanks, and decoded with a n-gram language model has a 5.6% WER. Replacing the n-gram LM by a convolutional one reduces the WER to 4.1%, and puts our model on par with the current best end-to-end system. Replacing the speech features by a learnable front-end finally reduces the WER to 3.7% and then to 3.5% when doubling the number of learnable filters, improving over DeepSpeech 2 and matching the performance of the best HMM-DNN system.

4.2. Analysis of the convolutional language model

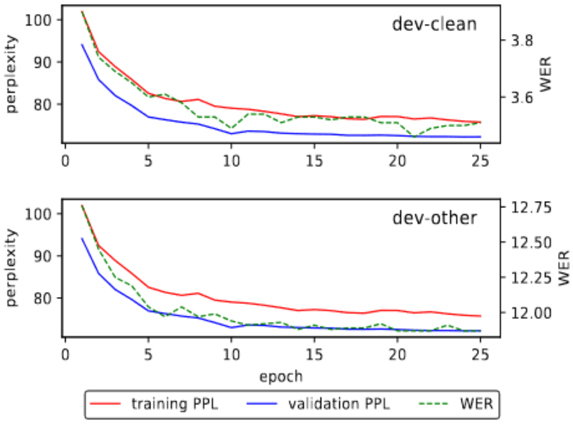
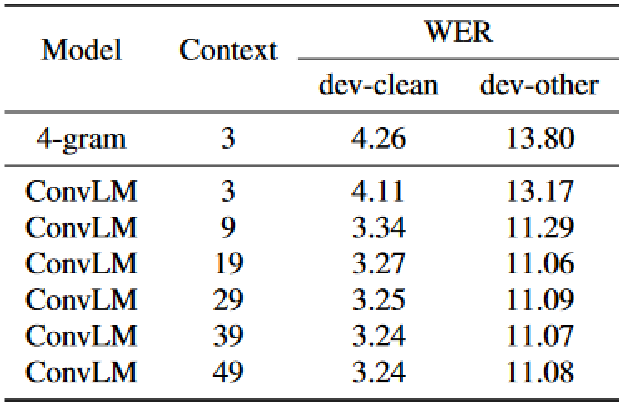


Figure 2: *Evolution of WER (%) on Librispeech with the perplexity of the language model.*



Model	Context	WER	
		dev-clean	dev-other
4-gram	3	4.26	13.80
ConvLM	3	4.11	13.17
ConvLM	9	3.34	11.29
ConvLM	19	3.27	11.06
ConvLM	29	3.25	11.09
ConvLM	39	3.24	11.07
ConvLM	49	3.24	11.08

Table 3: *Evolution of WER (%) on Librispeech with the context size of the language model.*

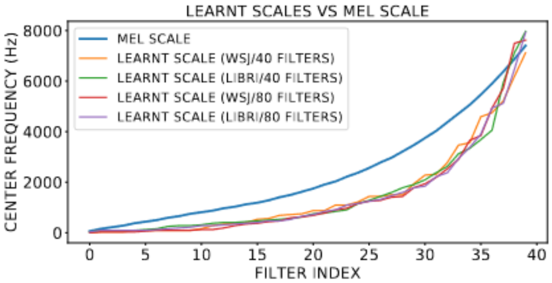


Figure 3: *Center frequency of the front-end filters, for the mel-filterbank baseline and the learnable front-ends.*

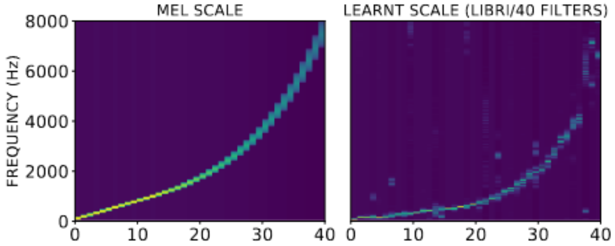


Figure 4: *Power heatmap of the 40 mel-filters (left) and of the frequency response of the 40 convolutional filters learned from the raw waveform on Librispeech (right).*

5. Conclusion

We introduced the first fully convolutional pipeline for speech recognition, that can directly process the raw waveform and shows state-of-the art performance on Wall Street Journal and on Librispeech among end-to-end systems. This first attempt at exploiting convolutional language models in speech recognition improves significantly over a 4-gram language model on both datasets. Replacing mel-filterbanks by a learnable front-end gives additional gains in performance, that appear to be more prevalent on noisy data. This suggests learning the front-end is a promising avenue for speech recognition with challenging recording conditions.