INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

Deep Learning-based Telephony Speech Recognition in the Wild

Kyu J. Han, Seongjun Hahm, Byung-Hak Kim, Jungsuk Kim, Ian Lane

Capio Inc., Belmont, CA, USA

{kyu,seongjun,byunghak,jungsuk,ian}@capio.ai

Abstract

In this paper, we explore the effectiveness of a variety of Deep Learning-based acoustic models for conversational telephony speech, specifically TDNN, bLSTM and CNN-bLSTM models. We evaluated these models on both research testsets, such as Switchboard and CallHome, as well as recordings from a real-world call-center application. Our best single system, consisting of a single CNN-bLSTM acoustic model, obtained a WER of 5.7% on the Switchboard testset, and in combination with other models a WER of 5.3% was obtained. On the CallHome testset a WER of 10.1% was achieved with model combination. On the test data collected from real-world call-centers, even with model adaptation using application specific data, the WER was significantly higher at 15.0%. We performed an error analysis on the real-world data and highlight the areas where speech recognition still has challenges.

Index Terms: Telephony Speech Recognition, Neural Networks, Acoustic Modeling, Recurrent Neural Network Language Models

1. Introduction

There has been significant improvement in the performance of conversational speech recognition in the past two decades. The application of deep learning [1, 2] to acoustic and language modeling has resulted in dramatic improvements. The most recent advancements based on deep learning have been reported by IBM [3, 4] and Microsoft [5, 6]. On the industry benchmark testset for telephone conversations, Switchboard, Microsoft claimed in [6] that they had exceeded human parity of 5.9% Word Error Rate (WER) by achieving 5.8%. Just a few months later, IBM announced their updated system performance of 5.5% [4], suggesting 5.1% as a new human parity and there is still a room to improve to match human ability of understanding spoken dialogues. Considering that 95% accuracy would be the first hurdle that needs to be overcome for mass adoption of speech recognition [7], it seems that we are on the verge of an era where speech recognition will be used by anyone, everywhere, and all the time.

In this paper, we show how Capio's telephony speech recognition system performing a **5.3**% WER has inched closer to human parity performance. We used three different neural network architectures, i.e., Time Delay Neural Network (TDNN), bi-directional Long Short-Term Memory (bLSTM), and Convolutional Neural Network bLSTM (CNN-bLSTM), while exploiting three unique English phonesets for each neural network acoustic model, i.e., CMU[1], MSU[2] and PronLex[3]. For language modeling and rescoring, we trained both N-gram models and Recurrent Neural Networks (RNNs). In order to combine the total 9 systems, we first applied frame-level acoustic model fusion within the same phoneset and then lattice combination across different phonesets. More details follow in Section 2.

In addition, this paper shows that a performance discrepancy still exists between research and production systems for conversational speech recognition. This is due to a few outstanding challenges in real-world applications for conversational speech recognition, which require dependable preprocessing stages, such as speech activity detection, utterance segmentation and diarization for multiple talkers in single-channel audio data. In Switchboard, for example, every test input is a single-speaker audio, prepared from stereo telephony recordings between two people, and manually pre-segmented to approximately 4 second segments on average. We discuss real-world challenges in production speech recognition systems in more details in Section 4.

The rest of the paper is organized as follows. In Section 2, we present the details of Capio's telephony speech recognition system in a categorized fashion of Acoustic Models, Language Models and System Combination. In Section 3, we analyze system performance in various perspectives as well as discuss how each component of the system contributes to the achieved WER of 5.3%. In Section 4, we discuss challenges for telephony speech recognition in practice. The discussion is based on our experience from the conversational speech recognition challenge hosted by one industry partner (undisclosed in the paper). We present how our best single system, i.e., CNN-bLSTM with the PronLex phoneset, performed on a task to recognize conversations between leasing agents and potential tenants, where multiple talkers may exist in the same channel, and show how even state-of-the-art systems for speech recognition would suffer in real-world applications. We summarize the contributions made by this paper and discuss future directions in Section 5.

2. System Structure

2.1. Acoustic Models

For a better diversity in system combination, we explored three different Deep Learning-based acoustic models - TDNN (uni-directional), bLSTM and CNN-bLSTM (bi-directional). TDNN [8] learns narrow contexts in shallow layers and, as it goes deeper, it processes wider temporal contexts from the hidden activations. Since each layer in TDNN can operate at a different temporal resolution, the overall resolution covered by the TDNN architecture could be modularized [9], increasing as we go to the deeper layers of the network. We built the 7-layer TDNN model with the left context of 17 frames and the right context of 12 frames to learn wide temporal relationships between adjacent acoustic features. The bLSTM model was built by stacking three standard bi-directional LSTM layers [10, 11, 12]. Each layer contains 1,024 cells and splices

[1] http://svn.code.sf.net/p/cmusphinx/code/trunk/cmudict

[2] http://www.isip.piconepress.com/projects/switchboard/releases/sw-ms98-dict.text

[3] https://catalog.ldc.upenn.edu/LDC97L20

http://dx.doi.org/10.21437/Interspeech.2017-1695

Table 1: *Comparion of TDNN, bLSTM and CNN-bLSTM in terms of the total number of parameters in the neural networks (in millions) and the total training time when using 16 NVidia K40 GPUs on the 2,000hr SWBD+Fisher training data (in hours).*

	TDNN	**bLSTM**	**CNN-bLSTM**
# of Params.	22M	46M	93M
Training Time	93hrs	255hrs	350hrs

50 context frames overall in both directions for longer temporal coverage. Each LSTM unit has peephole connections [13, 14] as well as recurrent and non-recurrent projection layers before the output layer [15]. For both TDNN and bLSTM, we used 40-dimensional MFCCs being appended with 100-dimensional i-vectors [17] for speaker-specific feature normalization [18]. The CNN-bLSTM model exploits the same LSTM architecture, and is composed of a total of ten layers of hidden units (first 3-layers: CNN, rest: bLSTM). Log-mel cepstra were fed into the three convolutional layers and a 3×3 kernel was applied with the filter size of 32 throughout the layers. The filtered signals were then passed to the 7-layer bLSTM after being appended with 100-dimensional i-vectors. Each neural network layer is followed by non-linear ReLU (**Re**ctified **L**inear **U**nit) activation. All of the three neural network models contain one fully connected layer before the softmax output layer. Table 1 shows a comparison of the three acoustic models in terms of the total parameters and the training time.

The Lattice-Free Maximum Mutual Information (LF-MMI) objective function was used for training with a sub-sampling rate of 3 frames. To avoid over-fitting during training, we also applied the cross-entropy objective function as an extra regularization as well as leaky HMM [16]. The total number of epochs for training was 4 as we observed that the losses were settled down after as much epochs.

We used three unique phonesets in CMU, MSU, and PronLex for acoustic modeling. The CMU phoneset consists of 39 phonemes with three lexical stress markers. The MSU has 36 phonemes with no stress distinctions. The PronLex phoneset, designed with the purpose of simple and internally consistent allophonic representation of standard American dialects, has 42 phonemes with 3 stress markers. Different trees were formed for the three phonesets during the acoustic model training stage, which generates diversity across phonesets in combination to benefit system combination later. For hesitation modeling, we used 11 distinct hesitation phones to better distinguish some hesitation utterances, such as 'uh-huh' and 'um-hum'.

We used the 2,000hr LDC telephony collection from Fisher English (Part 1 & 2) and Switchboard-1 (Release 2) for acoustic model training. For each phoneset, we prepared a 70K lexicon to cover the entire words contained in the training data (OOV rate $\approx$ 0.4% for the Switchboard/CallHome testset). Prior to neural network acoustic modeling, we first trained Gaussian Mixture Models (GMMs) within the framework of 3-state Hidden Markov Models (HMMs). The conventional 39-dimensional MFCC features were spliced over 9 frames and LDA was applied to project the spliced features onto a 40-dimensional sub-space. Further projection was conducted through MLLT for better orthogonality. Speaker Adaptive Training (SAT) was applied with feature-space MLLR (fMLLR) to further refine mixture parameters in GMMs [19].

Table 2: *CUED-RNNLM configurations.*

Model Structure	
Number of Layers	4
Number of Nodes	
Input layer:	65,244 nodes
1st hidden layer:	1,000 nodes
2nd hidden layer:	1,000 nodes
Output layer:	65,244 nodes
Node type	ReLU
Training	
Criterion	Variance Regularization (VR)
VR Penalty	0.3
BPTT	5
BPTT delay	1
Minibatch	64
Learning rate	0.015625

2.2. Language Models

The 4-gram language model (LM) was trained with the open-source library of OpenGrm [20] on a combination of data, including Fisher, Switchboard, CallHome, Broadcast News, Tedlium, and transcripts from 3,500hrs of 2-party telephone conversations from a variety of call-centers (approximately 3M sentences and 60M word tokens). We used this LM for the 2nd-pass LM rescoring. For the 1st pass decoding, we pruned the trained 4-gram LM with the pruning thresholds of 1.0e-8, 1.0e-7, and 1.0e-6 for bigrams, trigrams, and 4-grams, respectively. The RNN language model built with the CUED-RNNLM tookit [21] was trained on a subset of the aforementioned text data, consisting of only Fisher and Switchboard with 2M sentences and 24M word tokens. We used variance regularization [22] as the optimization criterion of the objective function for the RNN LM with 1,000 nodes in each of two hidden layers. More detailed configurations for the trained RNN LM is shown in Table 2.

2.3. System Combination

We used multi-level combination to combine the total 9 systems across three neural networks and three phonesets. The first-level combination applied a frame-level fusion of acoustic models within the same phoneset[4], resulting in phoneset-specific combined outputs. The combination weights for the three neural network acoustic models per each phoneset were trained using a held-out development set. In the second-level combination, we generated lattices from each combined output from the CMU, MSU, PronLex systems, and applied lattice combination, which conducts a union of lattices from component systems and searches the best path from the extended lattices [23]. We used equal weighting for the lattice combination. We took an advantage of the Kaldi tooklit [24] for implementation.

3. Experimental Results

We evaluated the performance of the three types of neural network acoustic models, across three different phonesets, on the

[4]Note that the acoustic models within the same phoneset share the same HMM states.

Table 3: *Experimental Evaluation of Capio's Telephony Speech Recognition System for three types of neural network acoustic models, across three different phonesets, with and without RNN language model rescoring. Performance is given as WER (%).*

Phoneset	Acoustic Model	SWBD		CallHome		SWBD + CallHome	
		N-gram	RNN	N-gram	RNN	N-gram	RNN
CMU	TDNN	8.4	7.4	15.2	13.7	11.8	10.6
	bLSTM	7.2	6.6	13.2	12.3	10.3	9.4
	CNN-bLSTM	6.7	6.1	12.2	11.5	9.5	8.8
	Frame-Level Fusion	6.3	5.7	11.5	10.7	8.9	8.2
MSU	TDNN	8.2	6.9	15.1	13.6	11.7	10.4
	bLSTM	7.5	6.6	14.0	13.1	10.8	9.9
	CNN-bLSTM	6.6	5.9	12.1	11.3*	9.4	8.7
	Frame-Level Fusion	6.3	5.6	11.3	10.7	8.8	8.2
PronLex	TDNN	8.2	6.9	14.9	13.5	11.6	10.2
	bLSTM	7.6	6.6	13.1	12.2	10.4	9.5
	CNN-bLSTM	6.5	5.7*	12.8	12.0	9.7	9.0
	Frame-Level Fusion	6.1	5.4	11.3	10.8	8.7	8.1
3-Way System Combination		5.9	***5.3***	10.6	***10.1***	8.3	7.7

*: best single system performance on SWBD/CallHome

Switchboard and CallHome testsets. The model performance in terms of WER, before and after RNN language model rescoring, is shown in Table 3.

Among the three neural network acoustic models, CNN-bLSTM outperformed the other two models for every phoneset. The CNN-bLSTM model obtained WERs 0.7%-0.9% lower than bLSTM and 1.2%-2.0% lower than the TDNN model, showing the effectiveness of this model. Adding convolutional layers to the bLSTM model in addition to adding 4 additional bLSTM layers appears to improve the robustness of the acoustic model, leading to a reduction in WER of up to 15% relative (from 13.1% to 11.3%) in the case of the MSU phoneset. We observed that RNN rescoring provides consistent improvement on top of N-gram models in all cases with absolute improvements between 0.6% and 1.5%, and a maximum reduction in WER of up to 10% relative in the case of the TDNN model with the CMU phoneset on CallHome (from 15.2% to 13.7%).

Frame-level fusion of acoustic scores enables efficient decoding across the three neural network acoustic models per phoneset. With weighted frame-level fusion we observed that the performance across the three phonesets is almost identical, demonstrating that the phoneset itself does not have a strong influence in system performance. During decoding weights of 0.25, 0.30 and 0.45 were applied for the TDNN, bLSTM and CNN-bLSTM models respectively, for all phonesets.

Finally, using lattice combination we combined the output across the three phonesets, using lattices generated with frame-level fusion. The combined output obtained a WER of 5.3% for Switchboard and 10.1% for CallHome. To date these are the best reported numbers on these two testsets.

We note that the best individual model was the CNN-bLSTM acoustic model built using the PronLex phoneset. This single system obtained a WER on Switchboard of 5.7%, only a 0.4% drop compared to our joint Frame-Level Fusion + System Combination output. This individual system obtained accuracies that are significantly higher than the individual systems reported in [4] (best single system result is 7.2%) or [6] (best single system result is 6.6%) and is even better than the 15-way combined system result reported in [6] (5.8%). For real-world systems, it is much more common to use a single acoustic model during decoding. For our real-world evaluation in section 4, we use this model as our non-adapted baseline.

Table 4: *WER (%) of Capio's CNN-bLSTM model on real-world recordings from a call-center analytics application. Performance is shown with no model adaptation and when adaptation is performed with 10 and 125hrs of application-specific data*

Adaptation	Dev	Eval
-	16.0	20.2
10hrs	14.3	15.8
125hrs	13.7	15.0

4. Real-World Performance

To evaluate the real-world performance of our system we evaluated our best individual speech recognition system on real-world data from a call-center analytics application. To better deal with the type of conversations encountered in the targeted call-center application, we explored adapting both the acoustic and language models with first 10hrs (83,403 words) of data and then with an additional 115hrs (1,183,868 words). We evaluated the performance with no adaptation, adapting with 10hrs of application-specific speech data and adapting with 125hrs of application-specific data.

The evaluation set consists of 13.6hrs of real-world, 2-party telephone conversations between a call-center agent and a customer. Conversations are on average 10 minutes in length and the evaluation set consists of both single-channel recordings (10.3hrs) where the agent and customer are recorded on separate channels and multi-speaker recordings (3.3hrs), where both speakers are recorded on a single audio channel. A separate 2hr development set (Dev), was used for parameter tuning.

The evaluation was performed using automatic segmentation. An initial decoding pass was performed using the MSU-TDNN system. Silences whose length were longer than 1 second were used as segmentation boundaries. Using these initial segments, segmentation was applied a second time resulting in smaller utterance-like segments for decoding.

Table 4 shows the performance of our speech recognition system before and after adaptation. With no adaptation the WER is 20.2%. This is twice the WER the same model obtained on CallHome testset. Adapting the system on 10hrs of application-specific data provided a significant jump in performance reducing the WER from 20.2% to 15.8%. Further adaptation using an additional 115hrs of data (a total of 125hrs of speech data) obtained a small additional reduction in WER from 15.8% to 15.0%.

Even after adaptation, the difference in performance of the same model on the Switchboard and CallHome testsets compared to real-world telephony data is dramatic, almost a 50% higher error rate. Through analysis of the speech recognition output we determined that the speech recognition errors in the real-world evaluation set were mainly due to *unclear speech*, difficulty transcribing *non-native English speakers* and speakers with *strong non-US dialects*, dealing with *over-lapping speech* in multi-speaker recordings and *automated utterance segmentation*. We describe these challenges below:

4.1. Unclear Speech/Non-nativity/Dialects

For those recordings whose WERs were extremely high, e.g., over 75%, there included a significant amount of non-audible murmurs by the speakers from diverse ethnic groups. We also note that approximately 25% of utterances used for adaptation also included one or more sections that were not-audible, highlighting the difficulty, even for professional transcribers, to accurately transcribe conversational telephony data.

4.2. Multiple Speakers in One Channel

10% of the evaluation set contained more than 2 speakers or multiple speech sources other than humans, like automatic response systems. The average WER of those recordings is 5% higher than in the single-speaker cases. Given that we exploit i-vectors as supplementary speaker-specific features in addition to high resolution MFCCs across the entire acoustic models, the i-vectors extracted from multi-speaker audio recordings would hurt speaker normalization on the features and result in performance degradation. Speaker diarization could be a help.

4.3. Utterance segmentation

Results reported on the Switchboard and CallHome test sets typically use manual segmentation and are thus overly optimistic compared to real-world performance. For real-world applications, speech activity detection and utterance segmentation is required. With our 2-pass segmentation approach, we observed a 7% relative degradation in WER (from 12.7% $\rightarrow$ 13.7%) on the development set when compared to manual segmentation. Errors due to utterance segmentation were most pronounced when there were background speech in single-speaker audio, and when there were rapid changes between speakers in multi-speaker recordings.

To overcome these challenges we believe that multiple approaches are required. We expect that the performance of the system can definitely be improved with additional training data, specifically additional acoustic speech data to provide better coverage of Non-Native and Dialectal speech. New approaches are required to effectively handle unclear speech (those segments that are incoherent to a human transcriber) both during training and decoding. The speech recognition system should have the option to back off when it encounters incoherent regions so errors do not propagate further in time. Finally, utterance segmentation, speaker-tracking and speaker separation (in regions of overlapping speech) all directly affect the performance of the down-stream speech recognition systems. New methods such as joint speaker and utterance tracking may be required to overcome these issues. These are all areas we intend to explore in the future.

5. Conclusions

In this paper, we have presented the two important contributions by Capio's telephony speech recognition system. One is 5.7% WER on Switchboard from the single best system in PronLex CNN-bLSTM. The other is 5.3% on Switchboard from the 9 system combination, which is the closest performance so far to human parity, yet reached by any other system. In addition, we have discussed how the most state-of-the-art system for conversational speech recognition could still suffer from real-world application data in various perspectives, even after domain-specific adaptation of acoustic and language models.

By comparing the performance of the same model on both research tasks such as Switchboard and CallHome as well as test data from real-world applications, it is evident that there are a number of challenges remaining that need to be overcome to push the current technology for conversational speech recognition towards what will be massively adopted by the non-technical public without any setback.

6. Acknowledgements

The authors would like to acknowledge TranscribeMe for their collaboration in this paper, providing real-world data for both training and model evaluation.

7. References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[2] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, November 2012.

[3] G. Saon, T. Sercu, S. Rennie, and H. Kuo, "The IBM 2016 English conversational telephone speech recognition system," in *Proc. Interspeech*, pp. 7-11, 2016.

[4] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L. Lim, B. Roomi, and P. Hall, "English conversational telephone speech recognition by humans and machines," *arXiv:1703.02136*, 2017

[5] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "The Microsoft 2016 conversational speech recognition system," in *Proc. ICASSP*, pp. 5255–5259, 2017.

[6] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving human parity in conversational speech recognition," *arXiv:1610.05256*, 2016.

[7] BI Intelligence, "IBM edges closer to human speech recognition," *Business Insider*. Retrieved from http://www.businessinsider.com

[8] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Trans. Acoustics. Speech Signal Process.*, vol. 37, no. 3, pp. 328-339, March 1989.

[9] A. Waibel, "Modular construction of time-delay neural networks for speech recognition," *Neural Computation*, vol. 1, no. 1, pp. 39-46, 1989.

[10] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.* vol. 45, pp. 2673-2681, 1997.

[11] J. Schmidhuber, "Learning complex, extended sequences using the principle of history compression," *Neural Comp.*, vol. 4, pp. 234–242, 1992.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comp.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] F. Gers and J. Schmidhuber, "LSTM recurrent networks learn simple context free and context sensitive languages," *IEEE Trans. Neural Networks*, vol. 12, no. 6, pp. 1333–1340, 2001.

[14] F. Gers, N. Schraudolph and J. Schmidhuber, "Learning precise timing with LSTM recurrent networks," *J. Machine Learn. Research*, vol. 3, pp. 115–143, 2002.

[15] H. Sak, A. Senior and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," *arXiv:1402.1128*, 2014.

[16] D. Povey, V. Peddinti, D. Galvez, P. Ghahrmani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Proc. Interspeech*, pp. 2751-2755, 2016.

[17] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 19, no. 4, pp. 788–797, 2011.

[18] G. Saon, H. Soltau, D. Namahoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *Proc. ASRU*, pp. 55–59, 2013.

[19] M. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Comp. Speech and Lang.*, vol. 12, pp. 75–98, 1997.

[20] B. Roark, R. Sproat, C. Allauzen, M. Riley, J. Sorensen, and T. Tai, "The OpenGrm open-source finite-state grammar software libraries," in *Proc. ACL*, pp. 61–66, 2012.

[21] X. Chen, X. Liu, Y. Qian, M.J.F. Gales, P.C. Woodland, "CUED-RNNLM - An open-source toolkit for efficient training and evaluation of recurrent neural network language models," in *Proc. ICASSP*, pp. 6000–6004, 2016.

[22] Y. Shi, W. Zhang, M. Cai, J. Liu, "Variance regularization of RNNLM for speech recognition," in *Proc. ICASSP*, 2014.

[23] H. Xu, D. Povey, L. Mangu, and J. Zhu, "Minimum Bayes risk decoding and system combination based on a recursion for edit distance," *Comp. Speech and Lang.*, vol. 4, pp. 802–828, 2011.

[24] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *Proc. ASRU*, 2011.