INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Detection and Recovery of OOVs for Improved English Broadcast News Captioning

Samuel Thomas, Kartik Audhkhasi, Zoltán Tüske, Yinghui Huang, Michael Picheny

IBM Research AI, Yorktown Heights, USA

{sthomas,kaudhkha,zoltan.tuske,huangyi,picheny}@us.ibm.com

Abstract

In this paper we present a study on building various deep neural network-based speech recognition systems for automatic caption generation that can deal with out-of-vocabulary (OOV) words. We develop several kinds of systems using various acoustic (hybrid, CTC, attention-based neural networks) and language modeling (n-gram and RNN-based neural networks) techniques on broadcast news. We discuss various limitations that the proposed systems have and introduce methods to effectively use them to detect OOVs. For automatic OOV recovery, we compare the use of different kinds of phonetic and graphemic sub-word units, that can be synthesized into word outputs. On an experimental three hour broadcast news test set with a 4% OOV rate, the proposed CTC and attention-based systems are capable of reliably detecting OOVs much better (0.52 F-score) than a traditional hybrid baseline system (0.21 F-score). These improved detection gains translate further to better WER performance. With reference to a non-OOV oracle baseline, the proposed systems at just 12% relative (1.4% absolute) loss in word error rate (WER), perform significantly better than the traditional hybrid system (with close to 50% relative loss), by recovering OOVs using their sub-word outputs.

Index Terms: speech recognition, out-of-vocabulary word detection and recovery, end-to-end systems

1. Introduction

Development of automatic captioning for broadcast news has been an active speech research area for several decades. This technology continues to be of prime importance given Federal Communications Commission (FCC) mandates that require captions for live and near-live clips be available within 12-hours of the original broadcast [1]. In addition to availability requisites, automatic closed captions also have stringent accuracy requirements because of real-time readability issues. Given the large impact deep learning has had on speech recognition, we present a study on building deep neural network-based automatic speech recognition (ASR) systems that can deal with out-of-vocabulary (OOV) words which adversely affect the accuracy and readability of automatically produced captions. The following is an example fragment from an automatic broadcast news caption with two processing strategies to compensate for the errors -

ASR Transcript: *... a new SMITH SONY AN exhibit opened ...*

Detect: *... a new <unintelligible> exhibit opened ...*
$\Rightarrow$ *... a new exhibit opened ...*
Recover: *... a new <_S M IT H S ON I AN_> exhibit opened ...*
$\Rightarrow$ *... a new Smithsonian exhibit opened ...*

In the above example, the occurrence of an OOV word causes the recognizer to use several similar sounding in-vocabulary words, which in turn impacts both the accuracy and readability of the caption. In the first of two possible strategies that can be employed to rectify the error, if the OOV region can be accurately detected, the readability of the caption can probably be improved by masking the error with an appropriate captioning tag. In a second, more complex strategy, the OOV word could possibly be automatically recovered by the system using an open-vocabulary speech recognition system.

Both OOV detection and recovery are very well researched topics in literature. Most of the techniques proposed for detecting OOV region can roughly be placed under one of two broad themes. For methods under the first theme, a hybrid model which explicitly models OOV words with sub-word units [2, 3] is typically used. Under the second broad theme, various confidence scores produced by ASR systems [4] are used to categorize recognition outputs into OOV/non-OOV regions. There have also been several efforts at the intersection of both these themes. In [5] for example, a hybrid model is combined with word level acoustic model confidence scores; [6] uses joint word/phone lattice alignments to classify miss-alignment regions as OOVs; and in [7] contextual information is modeled using conditional random fields with features such as existence of a sub-words, to detect OOVs.

For OOV recovery, one popular approach based on open-vocabulary speech recognition, is building a word/subword recognizer with a hybrid language model [8, 9]. With these techniques, gains have been reported not only in detecting OOV but also improving phone/word error rates for ASR performance [10]. In [3, 11, 12] phonemes have been used for OOV recovery, in conjunction with a grapheme-to-phoneme conversion model [13]. [14] further improves detection and recovery by learning sub-word units optimized for a given task instead of using a predefined unit.

More recently OOV detection and resolution has also been investigated in the context of acoustic-to-word neural network models [15, 16]. In both these studies, a primary whole word based recognition system is used along with a separate character based model for OOV recovery. On the other hand, for open vocabulary end-to-end speech recognition, [17, 18] have demonstrated the effectiveness of byte-pair encoding (BPE) [19] over character based systems with just a single deep neural network system. In this paper we present a comprehensive study on both these approaches. We show that these techniques compare well again each other while clearly outperforming traditional hybrid approaches. We further discuss the various challenges they pose and strategies to use them for both OOV detection and recovery. In Section 2 we describe the train/evaluation data and the various models we train. Each of the trained systems is then used to both detect and recover OOVs. These experiments along with their results are presented in Section 3. The paper concludes with a discussion of findings in Section 4.

http://dx.doi.org/10.21437/Interspeech.2019-2793

2. Building ASR Systems

2.1. Train and Test Data Sets

During several past DARPA programs, a significant amount of broadcast news (BN) data was collected and processed for training various ASR systems. In this paper, we use processed versions of these data sources [20] to build deep neural network-based acoustic and language models. We construct a corpus of about 1300 hours (**BN-1300**), which includes 144 hours of carefully transcribed audio along with data with semi-supervised transcripts created from biased LM decodes of matching audio. We use the BN-1300 corpus to train various acoustic models used in this paper. We further design a second data set which comprises of all the processed BN data [20]. This data set (**BN-6000**) includes the BN-1300 data set and an additional 4700 more hours of BN data. The BN-6000 corpus has a vocabulary size of 67K words and a running word count of 68 million. We use the BN-6000 corpus to train various language models in this paper. The efficacy of various acoustic and language models developed in this paper is tested against a locally collected 3 hour experimental broadcast new corpus set. With roughly 33K running words, this test set has a vocabulary of close to 5K words. We further select 520 words that overlap both with the BN-6000 and the test set vocabularies, as an OOV set. To ensure that this set is truly OOV, these words are removed from the BN-6000 training transcripts and vocabulary (**BN-6000-OOV**) before various models developed in this paper are created. The test set has an OOV rate of 4.2% when measured against this corpus. Nearly 60 words are *true* OOVs (unique to the test set) and do not appear in BN-6000, the rest are *in-training* words.

2.2. Hybrid baseline systems

To build our hybrid baseline models we begin by training traditional HMM-GMM-based systems on the BN corpus [20]. Similar to the architecture in [20], we train an LSTM acoustic model with 6 bidirectional layers having 1024 cells per layer (512 per direction), one linear bottleneck layer with 256 units and an output layer with 32K units corresponding to the context-dependent HMM states we derived in the HMM-GMM system build. The model is trained using non-overlapping sub-sequences formed using 21 frames of 40-dimensional speaker independent log-mel features with global cepstral mean subtraction and their deltas and double-deltas. For the cross-entropy based training on the BN-1300, sub-sequences from different utterances are grouped into mini-batches of size 128 for processing speed and reliable gradient estimates.

2.3. CTC-based phone/word piece systems

Similar to the hybrid baseline model, we use the CTC criteria [21] to train 6-layer BLSTMs with 512 hidden neurons in each direction [22]. The output 1024-dimensional vector from the final BLSTM layer is passed through a linear layer and softmax activation function to obtain a posterior probability vector over the output units and the blank symbol. We use log-mel, delta and double-delta features with frame stacking and skipping at a rate of 2 to yield a 240-dimensional feature vector. Models are trained using stochastic gradient descent with a batch size of 16, learning rate of 0.01, and Nesterov momentum of 0.9 similar to other CTC models developed in literature [23, 24, 25, 26, 27]. To train these models, we explore two sets of output units - phone-pieces and word-pieces, using byte-pair encoding (BPE) [19, 17, 18]. We use the SentencePiece (SP) toolkit [28] for this purpose and design an inventory of word-pieces and phone-pieces, each with a vocabulary size of 100. While the phone-piece inventory essentially consists of 44 commonly used phone units and other multi-phone pieces based on the basic phones; the word-piece inventory is made of 26 English characters and other multi-character pieces based on the character set.

2.4. Attention-based systems

We build two separate attention-based encoder-decoder models on BN-1300 for our experiments - a full-word model and a character-based model, on speaker independent log-mel features. The encoder for both these models have a similar architecture and consists of 6 bidirectional LSTM layers, with 640 nodes per direction. Pyramidal frame rate reduction by max pooling [29] is used at the input of both these models - 16 for the full-word system, and 4 for the character-based system. The final layer of the encoder is a linear bottleneck with 256 nodes. The decoder contains a 256-dimensional embedding layer, one unidirectional LSTM layer with 256 nodes, a single attention mechanism, and an output layer. The decoder network uses location aware attention mechanism of [30], and rectified-linear-unit feed-forward neural network to perform the energy function calculation [31]. The size of the output layer is 31 for the character and 27K for the full-word model. The models were trained by optimizing cross-entropy using Nesterov momentum and teacher forcing with 80% probability [32, 33].

In order to detect OOV words with the direct acoustic-to-word model, we limit the model's vocabulary to 27K words and map all other words to a special unknown symbol - <unk>. The additional neural network-based LM we train has an embedding layer of the NNLM with 512 nodes, followed by 2 LSTM layers, each with 2048 nodes. Before the softmax-based estimation of a 27K-dimensional word posterior vector, the feature space is reduced to 128 by a linear bottleneck layer. During recognition time, the trained NNLM is used in shallow-fusion [34] with our attention model. More details on the training these models can be found in [35].

3. Experiments and Results

3.1. Baseline hybrid system

In our first set of experiments we develop baseline systems using a hybrid phone CD LSTM acoustic model trained on the BN-1300 corpus with various n-gram language models. The first experiment on this set is an oracle experiment using the hybrid LSTM phone CD acoustic model and a word ngram on BN6000 with the full vocabulary of 67K words available for LM training. The WER performance of this oracle experiment at 12.8%, is indicative of the system's performance limits on our test set.

Since the set of words selected as OOVs are absent in both the test vocabulary and LM training data, in order to recover OOVs, a word+fragment LM using whole words and sub-word units needs to be formed from the training corpus. To create such language models with phone-pieces/word-pieces, we first restrict the full words in the LM vocabulary to the top 10K words based on unigram probabilities **(10K-OOV word)**. The remaining 57K words are then represented using phone-pieces/word-pieces. As described earlier, we use two kinds of sub-words units - a set of 100 phone-pieces (PP) and a set of 100 word-pieces (WP). Although there are several alternatives to create fragments using these sub-word units, to have comparable results and similar vocabulary/language models across our various experiments, we do not create any additional frag-

ments other than the original 100 phone-pieces or word pieces **(10K-OOV word + 100 PP/10K-OOV word + 100 WP)**. This results in an increase of just 100 additional entries to the 10K word vocabulary. While we use this common strategy to build various word+fragment language models, we use the same CD LSTM acoustic model for all baseline experiments.

3.1.1. Detection

In our experiments for OOV detection, we create a 4-gram LM using 10K-OOV vocabulary on the [BN-6000-OOV] corpus. We employ ASR word confidences derived from consensus networks as a simple indicator of the presence of OOV regions in the output. We hypothesize that in OOV regions, words will be decoded with lower confidences than in other regions where the correct hypothesis has been predicted. In our first set of detection experiments, we compute the precision/recall/F1-scores (P/R/F1 scores) at various word confidence thresholds for the 10K-OOV word system. The results are summarized in Table 1. It can be seen that at high confidence thresholds, OOV words are correctly detected with high recall rates. Unfortunately, the system produces a high number of false positives and hence has low precision scores. These low scores are probably the result of two kinds of errors discussed in the next section.

Table 1: *Detection performance of hybrid CD phone models*

Confidence threshold	**P/R/F1**
Confidence 0.80	0.14/0.45/0.22
Confidence 0.90	0.13/0.56/0.21
Confidence 0.95	0.11/0.63/0.20
Confidence 0.99	0.10/0.72/0.17

3.1.2. Recovery

We now evaluate the performance of the baseline system to recover OOVs. For these experiments we use the [10K-OOV word + 100 PP] LM which has phone-pieces in addition to whole words in its vocabulary and language model. Phone pieces when they appear in decoded outputs are converted back to words using a simple lookup dictionary. Table 2 shows how the baseline system performs in this new setting designed to allow for phone pieces to be included to the ASR pipeline. Our first observation is the WER increase from 12.8% to 19.8% as a result of the ASR vocabulary being restricted to 10K words from 67K words (System 2). By introducing phone pieces to the language model [10K-OOV word + 100 PP], we now allow the system to recover from 2 possible kinds of errors - (a) the original errors due to true OOVs in the test set [Type 1 errors], and (b) a set of secondary errors due to words which now appear only as phone pieces in the language model and vocabulary [Type 2 errors]. Although phone piece segments appear in the output, the system cannot sufficiently recover both these errors as seen. The WER of System (3) improves only slightly from 19.8% to 19.5% as shown in Table 2. As a consequence of its training, the system prefers words or phone piece segments where several triphones appear in context. Since the size of phone piece vocabulary has been restricted to just the basic 100 units, very few of these units are segments with triphone contexts. The model hence in general, picks in-vocabulary words instead of the phone pieces, resulting in higher WERs and inability to recover words using PPs. We also observe that many recognized phone piece segments cannot be converted to words because they do not exactly match canonical pronunciations for their corresponding words. Often times, a single phone is probably missing or substituted by a confusable phone. This points to another limitation of the proposed system - the [10K-OOV word + 100 PP] system needs to be run with a *noisy* phone-to-grapheme converter which can account for these kind of errors. In the next section, we address both these issues - we model larger acoustic units (word/phone pieces) than triphones using the CTC criterion and also investigate the use of word pieces instead of phone pieces, to circumvent the need for accurate phone-to-grapheme converters.

Table 2: *Performance of various hybrid CD phone models*

System	**WER**
1. Non-OOV oracle hybrid model	12.8
2. Hybrid model with [10K-OOV word] vocab	19.8
3. Hybrid model with [10K-OOV word + 100 PP] vocab + recovery with lookup dictionary	19.5

3.2. CTC phone/word piece system

Similar to prior experiments, we train two separate LSTM acoustic model using the CTC criterion on the BN-1300 corpus using 100 phone/word pieces. The first experiment on this set is an oracle experiment using the CTC LSTM phone piece acoustic model and a word ngram on the BN6000 text data available with the full vocabulary of 67K words available for LM training. With *in-training* words selected as OOV still in vocabulary for these oracle experiments, the PP model and WP models perform at 12.7% and 11.7% respectively.

3.2.1. Detection

For our initial CTC-based detection experiments we use the same word confidence-based criteria that we use with the baseline system. At a word confidence score threshold of 0.80, with the phone piece based CTC acoustic model we obtain P/R/F1 scores of 0.03/0.16/0.05. At higher thresholds, we obtain still lower F-scores. A similar behavior is also observed with the word-piece based system as well. These low scores are a result of the inherently very peaky posterior outputs of CTC-based acoustic models. In general, word confidences derived from CTC-based consensus networks are always very high, resulting in a very high number of false positive and negative detections. Given that the word confidence based metric is not suitable for our CTC-based systems, we investigate the use of a different metric for these systems.

Since the CTC-based systems are inherently trained to recognize phone/word pieces, we hypothesize that in OOV regions phone/word pieces will be selected better over whole words. We hence propose a simple method where a detection is labeled as an OOV, if the underlying hypothesis segment is synthesized from only phone/word pieces. To test this simple detection metric, we first construct two separate CTC PP/WP systems with [10K-OOV] whole words and the remaining words modeled as either phone or word pieces [10K-OOV word + 100 PP]/[10K-OOV word + 100 WP] as described earlier. In both these cases, OOV words are not used either as whole words or to create PP/WP fragments. System (1) of Table 3 shows the first set of detection results. Both systems perform at much higher precision and recall compared to the hybrid CD based system, confirming the usefulness of the proposed metric.

However unlike the word confidence based method, this new detection criteria cannot be adjusted to vary the preci-

Table 3: *Detection performance of CTC-based PP/WP models*

System	PP P/R/F1	WP P/R/F1
1. Interpolation 1.0	0.21/0.68/0.32	0.31/0.63/0.42
2. Interpolation 0.8	0.16/0.69/0.26	0.29/0.71/0.41
3. Interpolation 0.5	0.08/0.78/0.15	0.23/0.74/0.36
4. Interpolation 0.2	0.03/0.94/0.07	0.04/0.94/0.07

sion/recall operating points of the system based on a threshold. To allow for such adjustments, we first create separate 4-gram phone/word piece language models on the BN-6000-OOV corpus entirely using phone or word pieces. These language models are then interpolated with the default word+fragment based LM using different weights. Systems (2-4) in Table 3 show the different performances we obtain while using interpolation weights of 0.8/0/5/0.2 on the default word+fragment based LM. System 4 for example, has much higher interpolation weight on phone/word piece model causing almost all words to be generated using phone/word pieces.

3.2.2. Recovery

We now examine how the [10K-OOV word + 100 PP] and [10K-OOV word + 100 WP] systems can effectively recover from OOVs. From Table 4, we see that both the PP and WP-based models degrade when the whole word vocabulary is reduced to [10K-OOV]. However since these systems model phone/word pieces directly and do not have triphone-like constraints, both systems can more efficiently recover both Type 1 and 2 errors described earlier. Although the phone-piece model behaves like the hybrid CD models and has a higher WER, the word-piece model performs remarkably well, reducing the WER to a difference of just 1.4% absolute with respect to the oracle baseline. Shifting the modeling assumptions and nature of acoustic units to more larger units, has clearly made a difference to OOV recovery. The word piece model has an extra advantage of also not requiring any look up dictionaries - we trivially combine fragments using special begin-of-word and end-of-word symbols attached to word-pieces.

3.3. Attention whole word system

In a final set of experiments, we increase the unit size of the detection system still further from phone/word pieces to modeling whole words using attention-based encoder-decoder systems. Precise OOV detection with attention-based systems, however needs proper time-stamps. This is an issue with these systems since the input features and output label streams of encoder-decoder models are not handled synchronously, and the decoder output does not contain time information. In order to use the encoder-decoder approach for OOV detection, we hence construct time information using the attention mechanism. Let $\alpha_{l,n}$ denote the attention value of encoder frame n in the lth decoding step, where $1 < n < N$. Since attention can be interpreted as a probability density function, the simple expected value of $\bar{n}_l = \sum_n n\alpha_{l,n}$ can easily be transformed back to time-stamps for every l, which could denote the middle of a word. While the expected value based time-stamps unfortunately lead to serious time-offset, the simple use of $\bar{n}_l = \arg\max_n \alpha_{l,n}$ resulted in much better detection accuracy.

3.3.1. Detection

Using the approach described above for producing accurate time markings, we now use the whole word model for OOV detection. We designate regions that are recognized with the special symbol - <unk> as OOV regions. Table 5 shows the results for the proposed system with and without the external LM. By being able to predict full words including OOV words, the system performs significantly better than both the hybrid and CTC-based sub-word based systems especially with an external LM.

Table 4: *Recovery performance of CTC/Attention-based models*

System	WER
1. Non-OOV oracle CTC phone piece model	12.7
2. CTC-PPM with [10K-OOV word] vocab	18.4
3. CTC-PPM with [10K-OOV word + 100 PP] + recovery using lookup dictionary	17.7
4. Non-OOV oracle CTC word piece model	11.7
5. CTC-WPM with [10K-OOV word] vocab	17.2
6. CTC-WPM with [10K-OOV word + 100 WP] + recovery using word piece combination	13.1
7. Non-OOV attention-based word model	14.2
8. [Vocab-OOV] attention-based word model	14.8
9. System 8 + recovery using character model	13.0

Table 5: *Detection performance of the attention-based system*

System	P/R/F1
1. Without external LM	0.44/0.53/0.48
2. With external LM	0.40/0.75/0.52

3.3.2. Recovery

We now use the character-based encoder-decoder system in conjunction with the word-based attention system to recover OOV words. In regions where the word-based system produces the <unk> symbol, we use synthesized word outputs of the character-based system. Systems 7-9 in Table 4 are WER performances of various attention-based systems. Since the word-based system is already operating with a different reduced vocabulary, the initial baseline number at 14.2%, is higher than the non-OOV oracle result of the word piece system. Constraining the vocabulary to be free of the selected OOV words, causes the WER to increase to 14.8%. Using the character-based system for recovery in the next experiment (System 9) allows the system to recover both Type 1 and 2 errors described earlier and reach a WER of 13.0%. This result is very encouraging as it shows the usefulness of the proposed time markings and recovery strategy using both character and word attention models.

4. Conclusions

In this paper we have conducted a comprehensive study on the use of various acoustic and language models for OOV detection and recovery. Our experiments demonstrate the effectiveness of using BPE-based/character/whole-word units for CTC/attention-based ASR systems. We have shown that these models can both efficiently detect and help recover OOVs compared to traditional hybrid systems. We have also addressed two important issues for OOV detection. For CTC-based model we have proposed a simple alternate detection metric to circumvent the use of confidence scores. For attention-based models we have demonstrated how accurate timings can be constructed for OOV detection. Using these methods we show how neural network-based models can be used to produce accurate captions for broadcast news.

5. References

[1] *FCC, Closed captioning of televised video programming*, ser. 47. C. F. R. §79.1, 2016.

[2] T. Schaaf, "Detection of OOV words using generalized word models and a semantic class language model," in *Proc. of Interspeech*, 2001.

[3] M. Bisani and H. Ney, "Open vocabulary speech recognition with flat hybrid models," in *Proc. of the European Conf. on Speech Communication and Technology*, 2005, pp. 725–728.

[4] L. Burget, P. Schwarz, P. Matejka, M. Hannemann, A. Rastrow, C. White, S. Khudanpur, and H. Hermansky, "Combining of strongly and weakly constrained recognizers for reliable detection of oovs," in *Proc. of ICASSP*, 2008, pp. 4081–4084.

[5] T. Hazen and I. Bazzi, "A comparison and combination of methods for OOV word detection and word confidence scoring," in *Proc. of ICASSP*, 2001, pp. 397 – 400 vol.1.

[6] H. Lin, J. A. Bilmes, D. Vergyri, and K. Kirchhoff, "OOV detection by joint word/phone lattice alignment," in *Proc. of ASRU*, 2007, pp. 478–483.

[7] C. Parada, M. Dredze, D. Filimonov, and F. Jelinek, "Contextual information improves OOV detection in speech," in *Proc. of NAACL*, 2010, pp. 216–224.

[8] J. Mamou, B. Ramabhadran, and O. Siohan, "Vocabulary independent spoken term detection," in *Proc. of ACM SIGIR*, 2007, pp. 615–622.

[9] C. Parada, A. Sethy, and B. Ramabhadran, "Query-by-example spoken term detection for OOV terms," in *Proc. of ASRU*, 2009, pp. 404–409.

[10] A. Rastrow, A. Sethy, B. Ramabhadran, and F. Jelinek, "Towards using hybrid word and fragment units for vocabulary independent lvcsr systems," in *Proc. of Interspeech*, 2009.

[11] I. Bazzi and J. Glass, "Learning units for domain-independent out-of-vocabulary word modeling," in *Proc. of Eurospeech*, 2001.

[12] A. Rastrow, A. Sethy, and B. Ramabhadran, "A new method for OOV detection using hybrid word/fragment system," in *Proc. of ICASSP*, 2009.

[13] S. F. Chen, "Conditional and joint models for grapheme-to-phoneme conversion," in *Proc. of Eurospeech*, 2003.

[14] C. Parada, M. Dredze, A. Sethy, and A. Rastrow, "Learning subword units for open vocabulary speech recognition," in *Proc. of ACL*, 2011, pp. 712–721.

[15] J. Li, G. Ye, R. Zhao, J. Droppo, and Y. Gong, "Acoustic-to-word model without OOV," in *Proc. of ASRU*, 2017, pp. 111–117.

[16] H. Inaguma, M. Mimura, S. Sakai, and T. Kawahara, "Improving OOV detection and resolution with external language models in acoustic-to-word ASR," in *Proc. of SLT*, 2018, pp. 212–218.

[17] T. Zenkel, R. Sanabria, F. Metze, and A. Waibel, "Subword and crossword units for CTC acoustic models," in *Proc. of Interspeech*, 2018, pp. 396–400.

[18] Z. Xiao, Z. Ou, W. Chu, and H. Lin, "Hybrid CTC-Attention based end-to-end speech recognition using subword units," in *Proc. of ISCSLP*, 2018.

[19] P. Gage, "A new algorithm for data compression," *The C Users Journal*, vol. 12, no. 2, pp. 23–38, 1994.

[20] S. Thomas, M. Suzuki, Y. Huang, G. Kurata, Z. Tuske, G. Saon, B. Kingsbury, M. Picheny, T. Dibert, A. Kaiser-Schatzlein, and B. Samko, "English broadcast news speech recognition by humans and machines," in *Proc. of ICASSP*, 2019.

[21] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proc. of ICML*, 2006, pp. 369–376.

[22] K. Audhkhasi, B. Kingsbury, B. Ramabhadran, G. Saon, and M. Picheny, "Building competitive direct acoustics-to-word models for English conversational speech recognition," in *Proc. ICASSP*, 2018, pp. 959–963.

[23] G. Zweig, C. Yu, J. Droppo, and A. Stolcke, "Advances in all-neural speech recognition," in *Proc. of ICASSP*, 2017, pp. 4805–4809.

[24] Y. Miao, M. Gowayyed, and F. Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *Proc. of ASRU*, 2015, pp. 167–174.

[25] C. Yu, C. Zhang, C. Weng, J. Cui, and D. Yu, "A multistage training framework for acoustic-to-word model," in *Proc. of Interspeech*, 2018, pp. 786–790.

[26] R. Sanabria and F. Metze, "Hierarchical multi task learning with CTC," in *Proc. of SLT*, 2018.

[27] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," in *Proc. of Interspeech*, 2015.

[28] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proc. of EMNLP*, 2018, pp. 66–71.

[29] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: a neural network for large vocabulary conversational speech recognition," in *Proc. of ICASSP*, 2016, pp. 4960–4964.

[30] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. of NIPS*, 2015, pp. 577–585.

[31] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. of ICML*, 2010, pp. 807–814.

[32] Y. Nesterov, "A method of solving a convex programming problem with convergence rate $O(1/k^2)$," in *Soviet Mathematics Doklady*, vol. 27, no. 2, 1983, pp. 372–376.

[33] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *Proc. of NIPS*, 2015, pp. 1171–1179.

[34] Ç. Gülçehre, O. Firat, K. Xu, K. Cho, L. Barrault, H. Lin, F. Bougares, H. Schwenk, and Y. Bengio, "On using monolingual corpora in neural machine translation," 2015. [Online]. Available: http://arxiv.org/abs/1503.03535

[35] Z. Tüske, K. Audhkhasi, and G. Saon, "Advancing sequence-to-sequence based speech recognition," in *Proc. of Interspeech*, 2019.