# CHARACTER-LEVEL INCREMENTAL SPEECH RECOGNITION WITH RECURRENT NEURAL NETWORKS

*Kyuyeon Hwang and Wonyong Sung*

Department of Electrical and Computer Engineering
Seoul National University
1, Gwanak-ro, Gwanak-gu, Seoul, 08826 Korea
kyuyeon.hwang@gmail.com; wysung@snu.ac.kr

## ABSTRACT

In real-time speech recognition applications, the latency is an important issue. We have developed a character-level incremental speech recognition (ISR) system that responds quickly even during the speech, where the hypotheses are gradually improved while the speaking proceeds. The algorithm employs a speech-to-character unidirectional recurrent neural network (RNN), which is end-to-end trained with connectionist temporal classification (CTC), and an RNN-based character-level language model (LM). The output values of the CTC-trained RNN are character-level probabilities, which are processed by beam search decoding. The RNN LM augments the decoding by providing long-term dependency information. We propose tree-based online beam search with additional depth-pruning, which enables the system to process infinitely long input speech with low latency. This system not only responds quickly on speech but also can dictate out-of-vocabulary (OOV) words according to pronunciation. The proposed model achieves the word error rate (WER) of 8.90% on the Wall Street Journal (WSJ) Nov'92 20K evaluation set when trained on the WSJ SI-284 training set.

***Index Terms***— Incremental speech recognition, character-level, recurrent neural networks, connectionist temporal classification, beam search

## 1. INTRODUCTION

Incremental speech recognition (ISR) allows a speech-based interaction system to react quickly while the utterance is being spoken. Unlike offline sentence-wise automatic speech recognition (ASR), where the decoding result is available after a user finishes speaking, ISR returns $N$-best decoding results with small latency during speech. These $N$-best results, or hypotheses, gradually improve as the system receives more speech data. Since ISR is usually employed for immediate reaction to speech, word stability [1, 2] and incremental lattice generation [3] have been important topics.

In this paper, we introduce an end-to-end character-level ISR system with two unidirectional recurrent neural networks (RNNs). An acoustic RNN roughly dictates the input speech and an RNN-based language model is employed to augment the dictation result through decoding. Compared to a conventional word-level backend for speech recognition system, the character-level ASR is capable of dictating out of vocabulary (OOV) words based on the pronunciation. Also, our model is trained directly from speech and text corpus and does not require external word dictionary or senone modeling.

There have been efforts to deal with OOV words in conventional HMM based ASR systems. In [4], graphemes are employed as basic units instead of phonemes. Also, a sub-lexical language model is proposed in [5] for detecting previously unseen words.

RNN-based character-level end-to-end ASR systems were studied in [6, 7, 8, 9, 10]. However, they lack the capability of dictating OOV words since the decoding is performed with word-level LMs. Recently, a lexicon-free end-to-end ASR system is introduced in [11], where a character-level RNN LM is employed. We further improve this approach by employing prefix tree based online beam search with additional depth-pruning for ISR.

The character-level ISR system proposed in this paper is composed of an acoustic RNN and an RNN LM. The acoustic RNN is end-to-end trained with connectionist temporal classification (CTC) [12] using Wall Street Journal (WSJ) speech corpus [13]. The output of the acoustic RNN is the probability of characters, which are decoded with character-level beam search to generate $N$-best hypotheses. To improve the performance, a character-level RNN LM is employed to augment the beam search performance. Also, we propose depth-pruning for efficient tree-based beam search. The RNN LM is separately trained with large text corpus that is also included in WSJ corpus. Unlike for word-level language modeling, conventional statistical LMs such as $n$-gram back-off models cannot be used because much longer history window is required for character-level prediction. Both acoustic RNN and RNN LM have deep unidirectional long short-term memory (LSTM) network structures [14, 15]. For continuous ISR on infinitely long input speech, they are trained with virtually infinite training data streams that are generated by randomly concatenating training sequences.

The proposed model is evaluated on a single test sequence that is generated by concatenating all test utterances in WSJ `eval92` (Nov'92 20k evaluation set) without any external reset of RNN states at the utterance boundaries. The ISR performance is examined by varying the beam width and depth. Generally, wider beam increases the accuracy. Under the same beam width, there is a trade-off between the accuracy and stability (or latency), where the balance between them can be adjusted by the beam depth.

## 2. MODELS

### 2.1. Acoustic model

The acoustic model is a deep RNN trained with CTC [12]. The network consists of two LSTM layers with 768 cells each, where the

This work was supported in part by the Brain Korea 21 Plus Project and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2015R1A2A1A10056051).

```
THREE ISSUES ADVANCED MICRO OF AMERICA THE ONLY
WAY TO DIVERSIFY INTO TREATING MODERN ARMIES

LOOKING AHEAD TO MR. LEYSEN WITH AN INTOLERABLE
POP CUT WHEN AN ALL POWERFUL STUDENT SEEKS ITS
CORE DRIVING UPJOHN STOVES

AMERICAN EXPRESS HASN'T YET SWORED PARTICULARLY
WITH THE RESTRUCTURING IS A COMMITMENT TO BUY
POTENTIAL BUYERS IN THE OPEN MARKET
```

**Fig. 1**. Example of character-level random text generation with the RNN LM.

network has total 12.2 M trainable parameters. The model is similar to the one in the previous work about end-to-end speech recognition with RNNs [6] except a few major differences. In our case, the RNN is trained by online CTC [16] with very long training sequences that are generated by randomly concatenating several utterances. There is no need to reset the RNN states at the utterance boundary. This is necessary for ISR systems that runs continuously with an infinite input audio stream. Also, our model has a unidirectional structure since bidirectional networks that are usually employed for end-to-end speech recognition are not suitable for low-latency speech recognition. This is because the backward layers in the bidirectional networks cannot be computed before the input utterance is finished.

The input of the network is a 40-dimensional log mel-frequency filterbank feature vector with energy and their delta and double-delta values, resulting in an 123-dimensional vector. The feature vectors are extracted every 10 ms with 25 ms Hamming window. The input vectors are element-wisely standardized based on the statistics obtained from the training set. The output is a 31-dimensional vector that consists of the probabilities of 26 upper case alphabets, 3 special characters, the end-of-sentence (EOS) symbol, and the CTC blank label.

The networks are trained with stochastic gradient descent (SGD) with 8 parallel input streams on a GPU [17]. The networks are unrolled 2048 times and weight updates are performed every 1024 forward steps. The network performances are evaluated at every 10 M training frames. The evaluation is performed on total 2 M frames from the development set. The learning rate starts from $1 \times 10^{-5}$ and is reduced by the factor of 10 whenever the WER on the development set is not improved for 6 consecutive evaluations. The training ends when the learning rate drops below $1 \times 10^{-7}$.

We trained the networks on two training sets. The first one is the standard WSJ `SI-284` set and the second one, `SI-ALL`, is the set of all speaker independent training utterances in the WSJ corpus. Note that the utterances with verbalized punctuations are removed from both training sets. Also, odd transcriptions are filtered out, which makes the final `SI-284` and `SI-ALL` sets contain roughly 71 and 167 hours of speech, respectively. WSJ `dev93` (Nov'93 20k development set) and `eval92` (Nov'92 20k evaluation set) sets are used as the development set and the evaluation set, respectively.

### 2.2. Language model

An RNN language model (LM) [18] is employed for the proposed ISR system since conventional statistical LMs such as $n$-gram backoff models are not suitable for character-level prediction since they cannot make use of very long history windows. Specifically, the

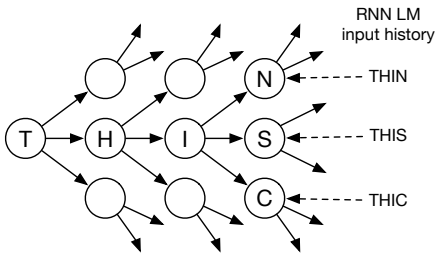


**Fig. 2**. Beam search tree consisting of label nodes. The CTC blank label is not included.

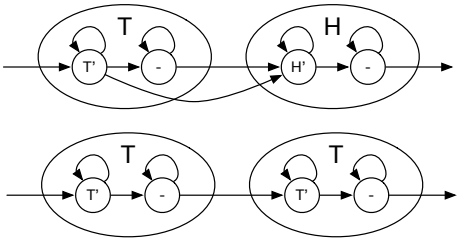


**Fig. 3**. CTC state transition between two label nodes. If the two nodes have the same label, then a transition between the same CTC state is not allowed.

RNN LM has a deep LSTM network structure with two LSTM layers where each of them has 512 memory cells, resulting in total 3.2 M parameters.

The input of the RNN LM is a 30-dimensional vector, where the current label (character) is one-hot encoded. The output is also a 30-dimensional vector which represents the probabilities of next labels. Although the RNN LM is trained to predict the next characters with only given the current character, the past character histories are internally stored inside the RNN and used for the prediction. It is well known that RNN LM can remember contexts for very long time steps.

As for the acoustic RNN, the RNN LM is trained on a very long text stream that is generated by attaching randomly picked sentences and inserting EOS labels between sentences. The RNN LM is trained with AdaDelta [19] based SGD method for accelerated training and better annealing. The WSJ LM training text with non-verbalized punctuation, which contains about 215 M characters, is used for training the RNN LM. Randomly selected 1% of the corpus is reserved for evaluation, on which the final bits-per-character (BPC) of the RNN LM is 1.167 (character-level perplexity of 2.245).

Random sentences can be generated following the method described in [20]. Briefly, the next label is randomly picked following the probabilities of the current output of the RNN LM and fed back to the RNN in the next step. By iterating these steps, texts can be sequentially generated as shown in Figure 1. From the example, it is clear that the RNN LM learned the linguistic structures as well as spellings of words that frequently appear.

## 3. CHARACTER-LEVEL BEAM SEARCH

### 3.1. Tree-based CTC beam search

Let $L$ be the set of labels without the CTC blank label. The label sequence $\mathbf{z}$ is a sequence of labels in $L$. The length of the label

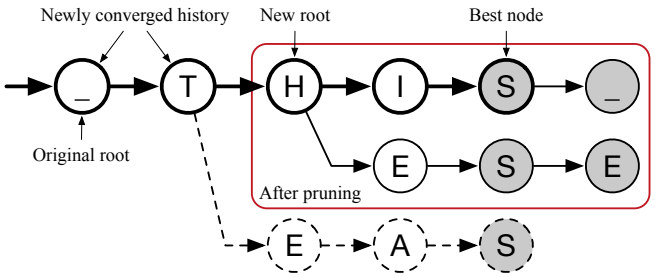


**Fig. 4**. Example of depth-pruning with the beam depth of 2. The pruning is performed by selecting a new root node so that the new depth of the best hypothesis node becomes the beam depth. The shaded nodes indicate the original active nodes. Also, the path of the best hypothesis is drawn with thick strokes.

sequence $\mathbf{z}$ is less than or equal to the number of input frames. The objective of the beam search decoding is to find the label sequence that has the maximum posterior probability given the input features from time 1 to $t$ generated by the acoustic RNNs, that is,

$$\mathbf{z}_{\max} = \arg\max_{\mathbf{z}} P(\mathbf{z}|x_{1:t}), \tag{1}$$

where $x_{1:t}$ is the input features from time 1 to $t$.

However, the CTC-trained RNN output has one more blank label. Let $L'$ be the set of labels (or CTC states) with the additional CTC blank label, and the path $\pi_t^{(i)}$ be a sequence of labels in $L'$ from time 1 to $t$. The length of the path $\pi_t^{(i)}$ is the same as $t$. By the definition of CTC, every $\pi$ can be reduced into the corresponding $\mathbf{z}$. For example, $\pi$ with "aab-c–a" corresponds to $\mathbf{z}$ with "abca", where "-" is the blank label.

There can be many paths, $\pi_t^{(i)}$, that can be reduced into the same $\mathbf{z}$. Let $\mathcal{F}(\cdot)$ be a function that maps a path to the corresponding label sequence, that is, $\mathcal{F}(\pi_t^{(i)}) = \mathbf{z}$, then the posterior probability in (1) becomes,

$$P(\mathbf{z}|x_{1:t}) = \sum_{\{\forall i|\mathcal{F}(\pi_t^{(i)})=\mathbf{z}\}} P(\pi_t^{(i)}|x_{1:t}). \tag{2}$$

Therefore, if the two different paths $\pi_t^{(j)}$ and $\pi_t^{(k)}$ in the decoding network are mapped to the same $\mathbf{z}$, then they can be merged by summing their probabilities.

For the beam search, we first represent the lattice with a tree-based structure so that each node has one of labels in $L$ as depicted in Figure 2. Then, backtracking from any node generates a unique label sequence $\mathbf{z}$. To deal with CTC state transitions, we need a state-based network that is represented with CTC states, $L'$. As shown in Figure 3, this can be easily done by expanding each tree node, of which label is in $L$, into two CTC states, one with the corresponding label in $L'$ followed by the blank CTC label. Since the label-level ($L$) search network is based on a tree structure, two different state-level ($L'$) paths with different label sequences never meet each other. This simplifies the problem since there is no interaction between two different sequence labelings (hypotheses) and (2) is the only equation that we should concern.

As proposed in [8, 11], external language models can be integrated by modifying the posterior probability term in (1) into:

$$\log(P(\mathbf{z}|x_{1:t})) = \log(P_{\text{CTC}}(\mathbf{z}|x_{1:t})) + \alpha\log(P_{\text{LM}}(\mathbf{z})) + \beta|\mathbf{z}|, \tag{3}$$

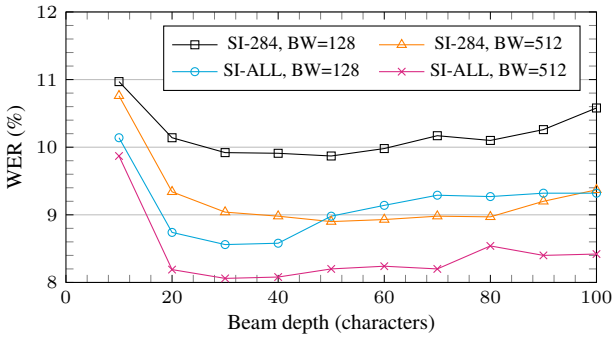


**Fig. 5**. WER of the proposed online decoding on the evaluation set with respect to the beam depth. Experiments are conducted with two acoustic RNNs trained on `SI-284` and `SI-ALL` and beam search is performed with the beam width (BW) of 128 and 512.

where $\alpha$ is the LM weight and $\beta$ is the insertion bonus. This modification can be applied by adding the additional terms with $\alpha$ and $\beta$ to the log probability of the destination state when a state transition between two different label nodes occurs.

The probability of the next label is computed using the RNN LM when a new active label node is added to the beam search tree. For this, the RNN LM context (hidden activations) is copied from the parent node to the child node and the RNN LM processes the new label of the child node with the copied context. Therefore, each active node has its own RNN LM context.

### 3.2. Pruning

Pruning of the search tree is performed by the standard beam search approach. That is, at each frame, only the active nodes with the top $N$ hypotheses and their ancestor nodes remain alive after the pruning with the beam width of $N$. However, this standard pruning, or *width-pruning*, cannot prevent the tree from growing indefinitely especially when the input speech is very long. This gradually degrades the efficiency of beam search on recent nodes since more and more hypotheses would be wasted to maintain the old part of the lattice that is already out of the context range of the RNN LMs.

To remedy this issue, we propose an additional pruning method called *depth-pruning*. The procedure is as follows. First, find the $M$-th ancestor of the node with the best hypothesis, where $M$ is the beam depth. Then, the ancestor node becomes a new root node. The pruning is performed by removing the nodes that are not descendants of the new root node. In this way, a beam can be better utilized for recent hypotheses rather than older ones. Figure 4 shows an example of depth-pruning with the beam depth of 2. Note that the depth of some nodes can be larger than the beam depth. In the following experiments, depth-pruning is performed every 20 frames.

## 4. EXPERIMENTS

The proposed ISR system is evaluated on a single 42-minute speech stream that is formed by concatenating all 333 utterances in the evaluation set, `eval92` (WSJ Nov'92 20k evaluation set). We use $\alpha = 2.0$ and $\beta = 1.5$ for the system trained with `SI-284`, and $\alpha = 1.5$ and $\beta = 2.0$ for the other one trained with `SI-ALL`.

The effects of beam depth and width to the final WER are examined in Figure 5. The gap between the beam width of 128 and 512

100: HE'S_THE_
150: HE'S_THE_ONLY_GU
200: HE'S_THE_ONLY_GUY_WHO_COULD_S
250: HE'S_THE_ONLY_GUY_WHO_COULD_SHOW_UP_IN_THE_
300: ...IN_THE_PLAZA_I
350: ...IN_THE_PLAZA_IN_ROCK_R
400: ...IN_THE_PLAZA_IN_**DRAW_RATE**_OF_SEVE
450: ...IN_THE_PLAZA_IN_DRAW_RATE_OF_SEVENTY_FIVE_THO
500: ...IN_THE_PLAZA_**AND**_DRAW_**CROWD**_OF_SEVENTY_FIVE_THOUSAND_**PEO**
550: ...IN_THE_PLAZA_AND_DRAW_CROWD_OF_SEVENTY_FIVE_THOUSAND_PEOPLE_S
600: ...IN_THE_PLAZA_AND_DRAW_CROWD_OF_SEVENTY_FIVE_THOUSAND_PEOPLE_SAYS_ONE_LA
650: ...IN_THE_PLAZA_AND_DRAW_CROWD_OF_SEVENTY_FIVE_THOUSAND_PEOPLE_SAYS_ONE_LATIN_DIPLOM
700: ...IN_THE_PLAZA_AND_DRAW_CROWD_OF_SEVENTY_FIVE_THOUSAND_PEOPLE_SAYS_ONE_LATIN_DIPLOMAT

Ground truth: HE'S_THE_ONLY_GUY_WHO_COULD_SHOW_UP_IN_THE_PLAZA_AND_DRAW_
**A**_CROWD_OF_SEVENTY_FIVE_THOUSAND_PEOPLE_SAYS_ONE_LATIN_DIPLOMAT

**Fig. 6**. Example of ISR partial results. The best hypothesis is shown at every 50 frames (500 ms). The word "ROCK" is corrected to "DRAW" after hearing "RATE" and "IN DRAW RATE" to "AND DRAW CROWD" while hearing "PEOPLE".

**Table 1**. CER / WER in percent on the evaluation set with online depth-pruning and offline sentence-wise decoding. The error rates are reported with two acoustic RNNs trained on SI-284 (71 hrs) and SI-ALL (167 hrs).

| Method | Beam width | SI-284 | SI-ALL |
|---|---|---|---|
| Online (no LM) | 512 | 10.96 / 38.37 | 9.66 / 35.44 |
| Online | 128 | 4.25 / 9.87 | 3.56 / 8.56 |
| Online | 512 | 3.80 / **8.90** | 3.39 / **8.06** |
| Sentence-wise | 128 | 4.46 / 10.30 | 3.63 / 8.84 |
| Sentence-wise | 512 | 4.04 / 9.45 | 3.38 / 8.28 |

**Table 2**. Comparison of WERs with other end-to-end speech recognizers in the literature. For reference, WERs of phoneme based GMM/DNN-HMM systems are also reported. All systems are trained with SI-284 and evaluated on eval92.

| System | Model | WER |
|---|---|---|
| Proposed ISR | Uni. CTC + Char. RNN LM | 8.90% |
| Graves and Jaitly [6] | CTC + Trigram (extended) | 8.7% |
| Miao *et al*. [9] | CTC + Trigram (extended) | 7.34% |
| Miao *et al*. [9] | CTC + Trigram | 9.07% |
| Hannun *et al*. [8] | CTC + Bigram | 14.1% |
| Bahdanau *et al*. [10] | Encoder-decoder + Trigram | 11.3% |
| Woodland *et al*. [21] | GMM-HMM + Trigram | 9.46% |
| Miao *et al*. [9] | DNN-HMM + Trigram | 7.14% |

is roughly 0.5% to 1% WER. However, there was little difference when the beam width increases from 512 to 2048 in our preliminary experiments. The best performing beam depths are 50 and 30 for the SI-284 and SI-ALL systems, respectively. This means the SI-ALL system can recognize speech more immediately than the SI-284 system. We consider this is because the acoustic model of the SI-ALL system can embed stronger language model due to increased training data, and can make decision more precisely without relying on the external language model much. The character error rate (CER) and WER are reported in Table 1 with the optimal beam depths. For comparison, we also report sentence-wise offline decoding results without depth-pruning.

The proposed ISR system is compared with other end-to-end word-level speech recognition systems in Table 2. The other systems perform sentence-wise offline decoding with bidirectional RNNs. The best result was achieved by Miao *et al*. [9] with a CTC-trained deep bidirectional LSTM network and a retrained trigram LM with extended vocabulary. The systems with the original trigram model provided with the WSJ corpus perform worse than our ISR system with character-level RNN LM. On the other hand, our system is beaten by the other ones with extended trigram models. However, more precise comparison of the decoding stages should be done by employing the same CTC model.

Figure 6 shows the incremental speech recognition result with the proposed ISR system. The best hypothesis is reported every 50 frames (500 ms). It is shown that the past best result can be corrected by making use of the additional speech input. For example, the word "ROCK" is changed to "DRAW" in the frame 450 by listening the word "RATE". Moreover, the correction of "IN DRAW RATE" to "AND DRAW CROWD" during hearing the word "PEOPLE" in the frame 500 is a good evidence that long term context can also be considered.

## 5. CONCLUDING REMARKS

A character-level incremental speech recognizer is proposed and analyzed throughout the paper. The proposed system combines a CTC-trained RNN with a character-level RNN LM through tree-based beam search decoding. For online decoding with very long input speech, depth-pruning is proposed to prevent indefinite growth of the search tree. When the proposed model is trained with WSJ SI-284, 8.90% WER can be achieved on the very long speech that is formed by concatenating all utterances in the WSJ eval92 evaluation set. The incremental recognition result shows the evidence that character-level RNN LM can learn dependencies between two words even when they are five words apart, which are hard to be caught using conventional $n$-gram back-off language models.

Note that the proposed system only requires speech and text corpus for training. External lexicon or senone modeling is not needed for training, which is a huge advantage. Moreover, it is expected that OOV words or infrequent words such as names of places or people can be dictated as they are pronounced.

## 6. REFERENCES


[1] Ethan O Selfridge, Iker Arizmendi, Peter A Heeman, and Jason D Williams, "Stability and accuracy in incremental speech recognition," in *Proceedings of the SIGDIAL 2011 Conference*. Association for Computational Linguistics, 2011, pp. 110–119.

[2] Ian McGraw and Alexander Gruenstein, "Estimating word-stability during incremental speech recognition," *Training*, vol. 17, no. 27,327, pp. 6–4, 2011.

[3] Gerhard Sagerer, Heike Rautenstrauch, Gernot A Fink, Bernd Hildebrandt, A Jusek, and Franz Kummert, "Incremental generation of word graphs.," in *ICSLP*. Citeseer, 1996.

[4] Mirjam Killer, Sebastian Stüker, and Tanja Schultz, "Grapheme based speech recognition.," in *INTERSPEECH*, 2003.

[5] Maximilian Bisani and Hermann Ney, "Open vocabulary speech recognition with flat hybrid models.," in *INTERSPEECH*, 2005, pp. 725–728.

[6] Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1764–1772.

[7] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al., "DeepSpeech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[8] Awni Y Hannun, Andrew L Maas, Daniel Jurafsky, and Andrew Y Ng, "First-pass large vocabulary continuous speech recognition using bi-directional recurrent DNNs," *arXiv preprint arXiv:1408.2873*, 2014.

[9] Yajie Miao, Mohammad Gowayyed, and Florian Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," *arXiv preprint arXiv:1507.08240*, 2015.

[10] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio, "End-to-end attention-based large vocabulary speech recognition," *arXiv preprint arXiv:1508.04395*, 2015.

[11] Andrew L Maas, Ziang Xie, Dan Jurafsky, and Andrew Y Ng, "Lexicon-free conversational speech recognition with neural networks," in *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, 2015, pp. 345–354.

[12] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.

[13] Douglas B Paul and Janet M Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 357–362.

[14] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 273–278.

[16] Kyuyeon Hwang and Wonyong Sung, "Online sequence training of recurrent neural networks with connectionist temporal classification," *arXiv preprint arXiv:1511.06841*, 2015.

[17] Kyuyeon Hwang and Wonyong Sung, "Single stream parallelization of generalized LSTM-like RNNs on a GPU," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1047–1051.

[18] Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Honza Černockỳ, and Sanjeev Khudanpur, "Extensions of recurrent neural network language model," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5528–5531.

[19] Matthew D Zeiler, "ADADELTA: An adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[20] Ilya Sutskever, James Martens, and Geoffrey E Hinton, "Generating text with recurrent neural networks," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 1017–1024.

[21] Phillip C Woodland, Julian J Odell, Valtcho Valtchev, and Steve J Young, "Large vocabulary continuous speech recognition using HTK," in *Acoustics, Speech, and Signal Processing, 1994. ICASSP-94., 1994 IEEE International Conference on*. IEEE, 1994, vol. 2, pp. II–125.