Rescoring of *N*-Best Hypotheses Using Top-Down Selective Attention for Automatic Speech Recognition

Ho-Gyeong Kim, Hwaran Lee, Geonmin Kim, Sang-Hoon Oh, and Soo-Young Lee

***Abstract*—In this letter, we propose an *N*-best rescoring system that integrates attentional information for locally confusing words extracted from alternative hypotheses to a conventional speech recognition system. The attentional information is derived by adapting a test input feature for the word of interest, which is motivated by the top-down selective attention mechanism of the brain. To rescore the competing hypotheses, we define a new confidence measure that contains both the conventional posterior probability and the attentional information for the confusing words. In addition, a neural network is designed to provide different weights within the confidence measure for each utterance. The network is then optimized to minimize the word error rates. Tests on the Wall Street Journal and Aurora4 speech recognition tasks were conducted, and our best results achieve a word error rate of 3.83% and 11.09%, yielding a relative reduction of 5.20% and 2.55% over baselines, respectively.**

***Index Terms*—Continuous speech recognition, *N*-best rescoring, parameter optimization, top-down selective attention.**

I. Introduction

THE standard criterion for speech recognition hypotheses aims to maximize a posterior probability (MAP) of the hypothesis over an utterance for a given acoustic and language model. Recent advances in the field of deep learning [1]–[4] have resulted in a number of changes in the design of acoustic and language parts in automatic speech recognition (ASR) systems. In particular, for acoustic modeling, both deep convolutional neural networks (CNNs) [5]–[8] and recurrent neural networks (RNNs) [9]–[12] have been successfully used for continuous speech recognition tasks instead of Gaussian mixture models (GMMs). Moreover, in an attempt to reduce the gap between training and test criteria, task loss optimization methods have been introduced [10], [13] to directly minimize word error rate (WER) in end-to-end learning frameworks.

Rescoring approaches to integrate speech information into the system have been explored in the literature. Most existing speech rescoring studies have focused on rescoring an N-best list or lattice with a large language model [16], [17] or with additional knowledge. Several researchers have attempted to employ knowledge sources (e.g., word posteriors [19], prosody [20], articulatory phonology [21]–[23], and morphology [24]) into ASR systems. The scores corresponding to the knowledge sources were generated from neural network based classifiers [22], [23] or task-specific designed probabilistic models [20]. However, the knowledge-based information is unmanageable when the additional knowledge (e.g., phone boundaries for the articulation attributes [21], [22]) is expensive to compute. In addition, the weights for different knowledge scores in a rescoring formula also have been introduced. These weights are optimized by minimizing the empirical sentence error [14], word error [18] or empirical risk [25] using a grid or gradient search. While most studies define the same weights for all utterances, optimal weights may differ for each utterance.

In cognitive science, a top-down selective attention (TDSA) mechanism of humans has been studied for decades [26]–[29] and is known to be controlled by "objects" in our mind via feedback processes. This cognitive process enhances the perceptual saliency of a response to the object of interest and filters out irrelevant responses. The engineering models using TDSA have been proposed for out-of-vocabulary rejection [30], and isolated word recognition [31]. In this work, we apply the TDSA mechanism to the N-best *rescoring* framework to provide *attentional* information of confusing words within competing hypotheses. The TDSA mechanism is applied to adapt a test input feature for several confusing words. The attentional information required to rescore the hypotheses is then derived as the probability of the adapted features and the amount of feature deformation.

Recently, numerous neural network models with attention have been developed and successfully applied to diverse tasks. The sequence to sequence learning framework [32] with attention has become especially popular for sequence labeling tasks such as neural machine translation [34], image caption generation [33], and speech recognition [35]. While predicting a soft-window over input sequences corresponding to output targets in previous attention works, our attention approach adapts a test input feature "directly" using a gradient to maximize the probability of the feature given target words. Therefore, our system provides the most probable feature of the target words without the need to train extra attention networks.

Manuscript received June 13, 2017; revised October 23, 2017; accepted November 5, 2017. Date of publication November 13, 2017; date of current version December 19, 2017. This work was supported in part by Institute for Information & Communications Technology Promotion Grant funded by the South Korea Government (MSIT) (R0126-15-1117), and in part by the Industrial Strategic Technology Development Program (10076757, Free-Running) funded by the Ministry of Trade Industry and Energy (MI, South Korea). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. James E. Fowler. *(Corresponding author: Ho-Gyeong Kim.)*

H.-G. Kim, H. Lee, G. Kim, and S.-Y. Lee are with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 305-701, South Korea (e-mail: hogyeong@kaist.ac.kr; hwaran.lee@kaist.ac.kr; gmkim90@kaist.ac.kr; sylee@kaist.ac.kr).

S.-H. Oh is with the Division of Information and Communication Convergence Engineering, Mokwon University, Daejeon 302-318, South Korea (e-mail: shoh@mokwon.ac.kr).

Color versions of one or more of the figures in this letter are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2017.2772828

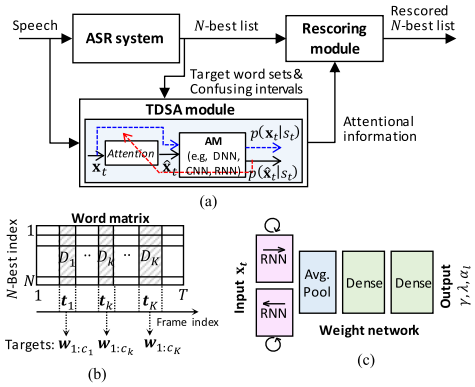

Fig. 1. (a) Proposed system consists of three main blocks: A conventional ASR system, the TDSA module, and the rescoring module. (b) Diagram of target sets and confusing frame intervals using a word matrix. (c) Architecture of the weight network for rescoring weights γ, λ, and α_l.

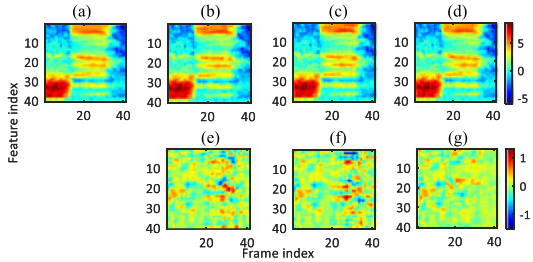

Fig. 2. Example of TDSA. The input feature (a) of a test utterance with a true transcription "SEE" is updated via TDSA processes with respect to "SUE," "SUIT," and "SEE" [from (b) to (d)]. The difference between $\mathbf{x}$ and $\hat{\mathbf{x}}$ from (e) to (g) indicate that there is a minimal feature deformation for the true target.

Finally, we emphasize that the proposed rescoring system has three main contributions: 1) Additional information from existing acoustic models is derived without any other knowledge source, which is motivated from the TDSA mechanism of the brain. 2) "Data-dependent" rescoring weights are used. 3) Moreover, our system helps reduce the mismatch between a word-based evaluation criterion, WER, and the standard sentence-based decision rule, MAP, by attempting to distinguish locally confusing words from the ASR hypotheses.

II. System Overview

A. TDSA Module

1) Confusing Words From an N-Best List: We initially determine a word matrix given the N-best list and their forced word alignments from a lattice, as shown in Fig. 1(b). We then extract the confusing frame intervals $\boldsymbol{t}$ when multiple words appear along the N-best axis. The target words $\boldsymbol{w}$ within $\boldsymbol{t}$ are determined as sets of locally confusing words, namely the target set D, by extracting a word or words along the N-best axis. In addition, we set the confusing intervals to the maximum length for all the confusing words in the same D set.

2) Feature Adaptation Using TDSA: The adaptation of input feature $\mathbf{x}$ is done by maximizing the log-likelihood $\ln p(\mathbf{x}|s)$. The expected input feature $\hat{\mathbf{x}}_t^j$ for the target $\boldsymbol{w}_j$ and time t is

$$\hat{\mathbf{x}}_t^j = \underset{\mathbf{x}_t}{\operatorname{argmax}} \ln p\left(\mathbf{x}_t | s_t^j\right) , \tag{1}$$

where s_t^j is one of output classes of the acoustic model for target $\boldsymbol{w}_j$. Because it is impossible to analytically solve (1), we use a gradient ascent method using $\hat{\mathbf{x}} = \mathbf{x} + \eta \left(\partial \ln p(\mathbf{x}|s)/\partial \mathbf{x}\right)$ based on an error back-propagation procedure in a top-down manner. In this process, the acoustic model forces the input feature to be adjusted to the target words.

However, as the above attention process continues, $\hat{\mathbf{x}}$ over-fits to the most likely feature of the target words. Therefore, we calculate the stopping log-likelihood to define a stopping criterion for each word in the dictionary. The histograms of the log-likelihoods a for the words are first calculated from a training set. We then utilize the value of the empirical cumulative distribution function $F(a)$ based on the histogram as a stop parameter p. Finally, the log-likelihood value corresponding to a specific value of p is used as the stopping log-likelihood a^* for each word (i.e., $a^* = F^{-1}(p)$). If the target contains several words, we use the average of the stopping log-likelihoods for the words.

B. Rescoring Module

1) Confidence Measure (CM): As a rescoring criterion for the N-best hypotheses using the attentional information from the TDSA module, we propose a CM, including the posterior from the conventional ASR system, posterior given the adapted feature, and the amount of feature deformation caused by TDSA. The TDSA process finds the most probable feature $\hat{\mathbf{x}}$ for the target near the input feature $\mathbf{x}$. If an input feature $\mathbf{x}$ of a test utterance is adapted to the wrong target words, $\hat{\mathbf{x}}$ will become quite different from $\mathbf{x}$. Therefore, the distance between $\mathbf{x}$ and $\hat{\mathbf{x}}$ is also regarded as an important measure to make a decision.

From this point of view, the ith CM is defined as

$$c_i = P(W_i|\mathbf{X})^{1-\gamma} P\left(W_i|\hat{\mathbf{X}}_i\right)^{\gamma} \exp\left(-\lambda\left(\sum\nolimits_l \alpha_l d\left(\hat{\mathbf{h}}_i^{(l)}, \mathbf{h}^{(l)}\right)\right)\right)$$
$$\text{for} \quad 0 \le \gamma \le 1, \quad \lambda \ge 0, \text{and} \sum\nolimits_l \alpha_l = 1. \tag{2}$$

Here, i is the hypothesis index and $P(W_i|\mathbf{X})$ is the original posterior probability of the ith word sequence given the input feature (including acoustic and language scores). In addition, $P(W_i|\hat{\mathbf{X}}_i)$ is the ith posterior given the whole adapted feature $\hat{\mathbf{X}}$, which is determined by changing $\mathbf{x}$ to $\hat{\mathbf{x}}$ for all D. Parameter γ controls the relative importance between the two and λ is the weight on the total difference. Moreover, $\hat{\mathbf{h}}^{(l)}$ and $\mathbf{h}^{(l)}$ are the lth layer outputs, which are propagated from $\hat{\mathbf{X}}$ and $\mathbf{X}$, respectively. Finally, d is the feature difference with weight α_l.

2) Optimization of the Rescoring Weights: As mentioned above, the optimal values of the rescoring weights in the CM may differ for each utterance. Hence, we design a neural network, namely the weight network, which helps provide data-dependent weights γ, λ, and α_l as shown in Fig. 1(c). The proposed network consists of a bidirectional RNN and a multilayer perceptron (MLP). The input of the network consists of the feature $\mathbf{x}_t$ inside the confusing intervals and the output represents the weight parameters. Specifically, to satisfy the conditions of the weight parameters in (2), the following nonlinear functions are used for the network output: sigmoid for γ, softplus [36] for λ, and softmax function for α_l.

The weight networks for γ, λ, and α_l are optimized by choosing one of the competing hypotheses that has the minimum WER of all the hypotheses. The cross-entropy criterion is used as the objective function:

$$J_{\mathrm{CE}} = -\sum\nolimits_u \sum_{i=1}^{N} \left(l_i^{(u)} \ln p_i^{(u)} + \left(1 - l_i^{(u)}\right) \ln\left(1 - p_i^{(u)}\right)\right) \tag{3}$$

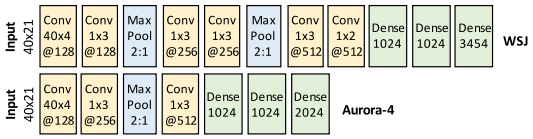

Fig. 3. Configurations of the CNN architectures for WSJ and Aurora4. "Conv," "MaxPool," and "Dense" denote convolutional, max-pooling, and fully-connected layers, respectively.

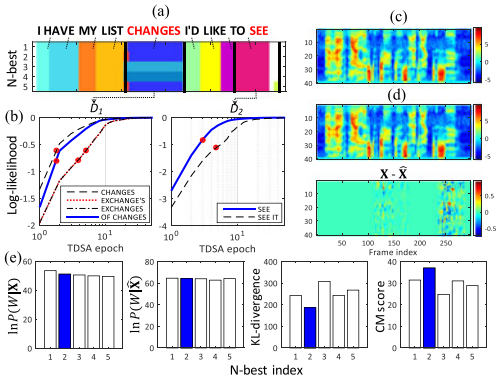

Fig. 4. Example of the rescoring system. (a) Color-coded word matrix and confusing intervals (black bold lines). (b) Log-likelihood curve and stopping points of the targets (red dots). (c) and (d) Initial and adapted features of the true hypothesis. (e) Bar plots of the terms in the CM for each hypothesis, indicating that the minimal feature deformation after the adaptation is also important for similar probabilities before and after adaptation.

where u is an utterance index, p_i is the normalization of a log-scaled c_i with a softmax function (i.e., $p_i = e^{\ln c_i} / \sum_{i'=1}^{N} e^{\ln c_{i'}}$), and l_i is the hard class label, which is coded when the ith word error of that hypothesis has a minimum value over all the competing hypotheses. Then, the error signals of weights γ, λ, and α_l for an utterance are derived as

$$\frac{\partial J_{\mathrm{CE}}}{\partial\{\gamma,\lambda,\alpha_l\}} = -\sum_{i=1}^{N}\left(\frac{l_i - p_i}{1-p_i}\right) \times \left(\frac{\partial \ln c_i}{\partial\{\gamma,\lambda,\alpha_l\}} - \frac{\sum_{i'=1}^{N} e^{\ln c_{i'}} \frac{\partial \ln c_{i'}}{\partial\{\gamma,\lambda,\alpha_l\}}}{\sum_{i'=1}^{N} e^{\ln c_{i'}}}\right)$$

where $\partial c_i/\partial\gamma = \ln P(W_i|\hat{\mathbf{X}}_i) - \ln P(W_i|\mathbf{X})$,

$$\partial c_i/\partial\lambda = -\sum_l \alpha_l d_i^{(l)}, \quad \text{and} \quad \partial c_i/\partial\,\alpha_l = -\lambda d_i^{(l)}. \tag{4}$$

The model parameters of the network are trained through standard error-back propagation using (4), which is regarded as the error at the network output. If several target sets exist (i.e., a number of D) for a test utterance, the outputs of different target sets are averaged for the final network output and the model parameters are shared for different D sets while the network was originally designed for a single D set.

Finally, the system rescores the *N*-best list for a test utterance based on the proposed CM using the outputs γ, λ, and α_l of the trained weight network and provides the top word sequence.

III. Experiments

We report a series of experiments using the Wall Street Journal (WSJ) [37] and Aurora4 [38] speech corpora, which are large vocabulary continuous speech recognition tasks. We used the 81-h training dataset (SI-284) of the WSJ corpus. The Aurora4 database is a subset (SI-84) of the WSJ with additive noise and convolutional distortion. The following results were trained on the multiconditioned training dataset.

TABLE I
WERs [%] of the Rescoring Systems on the WSJ

System	Top-1	Top-2	Top-3	Top-4	Oracle
Baseline	4.27	2.94	2.55	2.23	2.04
+ Rescoring (empirical)	4.09	2.78	2.49	2.23	
+ Rescoring (weight net.)	3.96	2.79	2.48	2.21	
+ sMBR training	4.04	2.83	2.34	2.00	1.79
+ Rescoring (empirical)	3.93	2.81	2.26	2.07	
+ Rescoring (weight net.)	3.83	2.74	2.21	1.94	

A. ASR Baseline

The raw speech signal was processed via short-time Fourier transform with a Hamming window of 25 ms and window shifts of 10 ms. We first trained the GMM-HMM system over the feature-space maximum likelihood linear regression (fMLLR) features. The forced alignment of each frame obtained by the GMM-HMM system is the target label of the neural networks for acoustic modeling. We used 40-dimensional log-mel filter bank (LMFB) features without the energy coefficient, and concatenated the frames with a context window size of 21 ($\pm$10 frames) to feed them into the networks as inputs. The CNN-HMM hybrid system is trained as the ASR baseline system. The configurations of the CNN architectures for WSJ and Aurora4 are shown in Fig. 4. Each layer is trained with a momentum of 0.9, an L2-decay term of 0.0005, and minibatch size of 512. We used a 146 K word extended dictionary and a trigram pruned language model.

B. System Setting

For the TDSA module, the stopping thresholds for the 146 K words in the dictionary were determined by observing the histograms of the log-likelihoods from the training speech for each corpus. For the rescoring module, we use the kullback-leibler (KL)-divergence from the adapted to the initial input features is used as the feature difference in the CM. The stop parameter p was first found using the development set for each corpus varying from 0.5 to 0.9 using an empirical search. The weight networks for the rescoring weights were then trained using the development set. The 40-dimensional LMFB features with a context window size of 11 frames were fed into the weight network as inputs. The numbers of neurons of the bidirectional recurrent layers and dense layer were set as 256 and 128, respectively. The activations of forward and backward RNNs were averaged to be used as the input to the dense layer. The number of neurons of the output layer is $2 + L$, where L is the number of layers in the CNN. The ASR baseline and proposed system were developed using the KALDI toolkit [39].

C. Rescoring Results

The performance of the proposed rescoring system is shown in Tables I and II. The WERs of the "Top-n" are presented as the recognition results to demonstrate the entire rescoring performance. The Top-n result of the baseline and proposed system implies the minimum WER among the n highest posterior probabilities and CM scores, respectively. The "Oracle"

TABLE II
WERS [%] OF THE RESCORING SYSTEMS ON THE AURORA4

System		A	B	C	D	Avg.
Baseline	Top-1	3.90	7.14	8.01	17.41	11.38
	Top-2	2.76	5.68	6.48	15.91	9.91
	Top-3	2.02	5.01	5.75	15.10	9.18
	Top-4	1.76	4.46	5.45	14.58	8.68
+ Rescoring (weight net.)	Top-1	3.77	6.73	7.68	17.24	11.09
	Top-2	2.34	5.33	6.03	15.50	9.52
	Top-3	1.91	4.69	5.51	14.83	8.89
	Top-4	1.72	4.37	5.27	14.45	8.56
Oracle		1.61	4.18	5.16	14.22	8.37

is the Top-N performance that is the lower bound WER. In Table I, the top-1 WER for WSJ was 3.96%, yielding a relative reduction of 7.26% when *N* is equal to 5. The overall rescoring results outperformed the results of the ASR baseline for every top-n result. Moreover, the system was achieved a WER of 4.01% and 3.97% when *N* is equal to 3 and 10, respectively. In addition, we evaluated the rescoring system based on the ASR baseline, which is retrained by an sMBR criterion [40]. The final WER of 3.83% was achieved for the rescoring system, yielding a relative reduction of 5.20% over the sMBR baseline, indicating that the acoustic model via sMBR training provides better acoustic information to rescore the hypotheses. Moreover, the results show that the adaptation rule still works when testing adaptation rule and training criterion are not the same. We also demonstrated an empirical weight search method to show the effectiveness of the weight network as presented in Table I. The rescoring system performed well on the Aurora4 as shown in Table II. For the top-2 results, substantial WER improvements of 15.5%, 6.16%, 6.94%, and 2.58% were achieved for A to D sets, respectively. It is remarkable that substantial improvements of the overall performance are achieved in both clean and noisy conditions.

D. Comparison With Previous Works

We compared the rescoring results of previous studies on the WSJ. The system for lattice rescoring achieved a WER of 4.13% using MLPs [22] and 4.0% using DNNs [23]. Although our system did not use any articulator information from the designed phonetic feature detectors, it yielded a comparative word error simply by adapting an input feature for confusing words using the existing acoustic models without any other knowledge source and rescoring an *N*-best list with a smaller size of hypothesis spaces compared to the entire lattice. In the end, we compared with other adaptation techniques on the WSJ. We trained DNNs using fMLLR features with and without *i*-vectors, which has the similar number of parameters with the CNN of the baseline, yielding a WER of 3.99% and 3.83%, respectively. In addition, we obtained a WER of 4.08% for the CNN using fMLLR features. Therefore, the rescoring system delivered competitive performance, where our best WER was 3.83%. It is noteworthy that our system directly adapts the input features using only the acoustic model without any speaker adaptation techniques.

E. Analysis on Optimized Weights

During the weight network training, we plotted the initial weights (red "x" points) and converged weights (yellow "o" points) on the map of the cross-entropy values in (3) and WERs in Fig. 5. Here, α_l are uniformly assigned to all the layers. In the left figure, the random outputs are initially predicted using the weight network, but (γ, λ) points are drawn near the minimum area of the cross-entropy map after the network are optimized. Comparing the left and right figures, although there is a difference between the maps of the cross-entropy of the development set and WER of the test set, the weights, to which the test input features are propagated, are moderately distributed on the map of the WER. In addition, the converged values of γ and λ were located around 0.16 and 0.01, respectively. The system performance degraded when $\gamma = 0$ or $\lambda = 0$, i.e., when not using the posterior probability with adapted features or the feature difference. This result implies that the proposed attentional information is helpful for improving the system performance with appropriate γ and λ values.

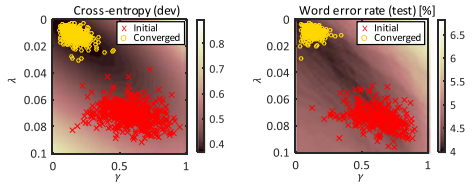

Fig. 5 Example of γ and λ points on the map of cross-entropy (right) and WER (left) for the development and test set of WSJ, respectively.

F. Computational Complexity

The computational complexity for the TDSA module was compared to the standard feedforward process in terms of the number of time frames and TDSA epochs. The average values of the size of the confusing intervals $|\boldsymbol{t}|$ and the number of TDSA epochs M for all targets in the test set of WSJ were 49.2 and 8.0, respectively. The average value of the total number of frames T was 760.3. Therefore, the average complexity of the TDSA process ($=|\boldsymbol{t}| \times M$) is less than half the average complexity of the feedforward process ($=T$) when parallelizing the computations for all targets per test utterance. Our system focuses on the locally confusing words, which results in a reduction in the total complexity.

IV. CONCLUSION

In this letter, we proposed an *N*-best rescoring system with acoustic attentional information via the TDSA process. The TDSA mechanism was used to adapt the input feature by maximizing the log-likelihood of the feature given confusing words. In addition, a stopping criterion was employed to avoid overfitting via iterative attention processes. The attentional information was finally integrated into the conventional ASR system in the form of the CM. Furthermore, we designed a neural network to output data-dependent rescoring weights in the proposed CM and it is optimized by minimizing the WERs. We demonstrated that the WERs were improved over the baseline on WSJ and Aurora4, clearly showing that the reranked performance was meaningful using the proposed CM. Finally, we emphasize that the proposed system can be applied on the ASR systems that are capable of generating competing hypotheses and providing the gradient of the input feature for confusing words. Such application will improve the recognition results including the rescoring performance even if the testing adaptation rule and training criterion are not the same.

REFERENCES

[1] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 8609–8613.

[2] G. E. Hinton, L. Deng, D. Yu, and G. E. Dahl, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[3] L. Deng *et al.*, "Recent advances in deep learning for speech research at Microsoft," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 8604–8608.

[4] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 4580–4584.

[5] O. Abdel-Hamid, A. R. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2012, pp. 4277–4280.

[6] T. N. Sainath, A. R. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 8614–8618.

[7] O. Abdel-Hamid, L. Deng, and D. Yu, "Exploring convolutional neural network structures and optimization techniques for speech recognition," in *Proc. Int. Speech Commun. Assoc.*, 2013, pp. 3366–3370.

[8] H. Lee, G. Kim, H. G. Kim, S. H. Oh, and S. Y. Lee, "Deep CNNs along the time axis with intermap pooling for robustness to spectral variations," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1310–1314, Oct. 2016.

[9] A. Graves, A. R. Mohamed, and G. E. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 6645–6649.

[10] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 1764–1772.

[11] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *INTERSPEECH*, 2014, pp. 338–342.

[12] C. Weng, D. Yu, S. Watanabe, and B. H. F. Juang, "Recurrent deep neural networks for robust speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 5532–5536.

[13] D. Bahdanau, D. Serdyuk, P. Brakel, N. R. Ke, J. Chorowski, A. Courville, and Y. Bengio, "Task loss estimation for sequence prediction," in *Proc. Int. Conf. Learn. Represent.*, 2016. [Online]. Available: https://arxiv.org/abs/1511.06456.

[14] A. Stolcke, Y. Konig, and M. Weintraub, "Explicit word error minimization in n-best list rescoring," in *Proc. Eur. Conf. Speech Commun. Technol.*, 1997, vol. 97, pp. 163–166.

[15] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus among words: Lattice-based word error minimization," in *Proc. Eur. Conf. Speech Commun. Technol.,* 1999, pp. 495–498.

[16] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, 2003.

[17] T. Mikolov, M. Karafiat, L. Burget, J. Cernocky, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. Int. Speech Commun. Assoc.*, 2010, pp. 1045–1048.

[18] R. Schwartz *et al.*, "New uses for the N-best sentence hypotheses within the BYBLOS speech recognition system," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1992, pp. 1–4.

[19] G. Evermann and P. C. Woodland, "Large vocabulary decoding and confidence estimation using word posterior probabilities," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2000, pp. 1655–1658.

[20] S. Ananthakrishnan and S. Narayanan, "Improved speech recognition using acoustic and lexical correlates of pitch accent in a n-best rescoring framework," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2007, pp. 873–876.

[21] J. Li, Y. Tsao, and C. H. Lee, "A study on knowledge source integration for candidate rescoring in automatic speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2005, pp. 837–840.

[22] S. M. Siniscalchi, and C. H. Lee, "A study on integrating acoustic-phonetic information into lattice rescoring for automatic speech recognition," *Speech Commun.*, vol. 51, no. 11, pp. 1139–1153, 2009.

[23] S. M. Siniscalchi, D. Yu, L. Deng, and C. H. Lee, "Exploiting deep neural networks for detection-based speech recognition," *Neurocomputing*, vol. 106, no. 17, pp. 148–157, 2013.

[24] H. Sak, M. Saraçlar, and T. Güngör, "Discriminative reranking of ASR hypotheses with morpholexical and n-best-list features," in *Proc. 2011 IEEE Workshop Autom. Speech Recognit. Understanding*, IEEE, 2011, pp. 202–207.

[25] V. Goel and W. J. Byrne, "Minimum Bayes-risk automatic speech recognition," *Comput. Speech Lang.*, vol. 14, no. 2, pp. 115–135, 2000.

[26] S. Grossberg, "The attentive brain," *Amer. Sci.*, vol. 83, vol. 5, pp. 438–449, 1995.

[27] J. Kauramäki, I. P. Jääskeläinen, and M. Sams, "Selective attention increases both gain and feature selectivity of the human auditory cortex," *PLoS One*, vol. 2, no. 9, 2007, Art. no. e909.

[28] D. M. Beck and S. Kastner, "Top-down and bottom-up mechanisms in biasing competition in the human brain," *Vis. Res.*, vol. 49, no. 10, pp. 1154–1165, 2009.

[29] A. E. Paltoglou, C. J. Sumner, and D. A. Hall, "Examining the role of frequency specificity in the enhancement and suppression of human cortical activity by auditory selective attention," *Hear. Res.*, vol. 257, no. 1, pp. 106–118, 2009.

[30] K. Y. Park and S. Y. Lee, "Out-of-vocabulary rejection based on selective attention model," *Neural Process. Lett.*, vol. 12, no. 1, pp. 41–48, 2000.

[31] C. H. Lee and S. Y. Lee, "Noise-robust speech recognition using top-down selective attention with an HMM classifier," *IEEE Signal Process. Lett.*, vol. 14, no. 7, pp. 489–491, Jul. 2007.

[32] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.

[33] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. S. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Machine Learning*, 2015, pp. 2048–2057.

[34] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. Int. Conf. Learn. Represent.*, 2014. [Online]. Available: https://arxiv.org/abs/1409.0473

[35] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 4945–4949.

[36] S. Tan and K. C. Sim, "Fine context, low-rank, softplus deep neural networks for mobile speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2016, pp. 5965–5969.

[37] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proc. Workshop Speech Natural Lang. Assoc. Comput. Linguistics*, 1992, pp. 357–362.

[38] D. Pearce, "Aurora working group: DSR front end LVCSR evaluation AU/384/02," Ph.D. dissertation, Mississippi State Univ., Starkville, MS, USA, 2002.

[39] D. Povey *et al.*, "The Kaldi speech recognition toolkit," in *Proc. 2011 IEEE Workshop Autom. Speech Recognit. Understanding*, IEEE, 2011.

[40] D. Povey *et al.*, "The Kaldi speech recognition toolkit," in *Proc. Automatic Speech Recognition Understanding (ASRU), 2011 IEEE Workshop . IEEE*, 2011, pp. 1–4.

[41] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Interspeech*, pp. 2345–2349, 2013.