CHARACTER-AWARE ATTENTION-BASED END-TO-END SPEECH RECOGNITION

Zhong Meng, Yashesh Gaur, Jinyu Li, Yifan Gong

Microsoft Corporation, Redmond, WA, USA

ABSTRACT

Predicting words and subword units (WSUs) as the output has shown to be effective for the attention-based encoder-decoder (AED) model in end-to-end speech recognition. However, as one input to the decoder recurrent neural network (RNN), each WSU embedding is learned independently through context and acoustic information in a purely data-driven fashion. Little effort has been made to explicitly model the morphological relationships among WSUs. In this work, we propose a novel character-aware (CA) AED model in which each WSU embedding is computed by summarizing the embeddings of its constituent characters using a CA-RNN. This WSU-independent CA-RNN is jointly trained with the encoder, the decoder and the attention network of a conventional AED to predict WSUs. With CA-AED, the embeddings of morphologically similar WSUs are naturally and directly correlated through the CA-RNN in addition to the semantic and acoustic relations modeled by a traditional AED. Moreover, CA-AED significantly reduces the model parameters in a traditional AED by replacing the large pool of WSU embeddings with a much smaller set of character embeddings. On a 3400 hours Microsoft Cortana dataset, CA-AED achieves up to 11.9% relative WER improvement over a strong AED baseline with 27.1% fewer model parameters.

***Index Terms*—** character-aware, end-to-end, attention, encoder-decoder, speech recognition

1. INTRODUCTION

Traditional hybrid automatic speech recognition (ASR) system [1, 2, 3, 4] consists of an acoustic model, a pronunciation model and a language model. Different components are optimized separately towards different objectives. With the advance of deep learning, end-to-end (E2E) speech recognition has shown promising ASR performance by incorporating the three components into a single deep neural network (DNN) and directly mapping a sequence of input speech signal to a sequence of output labels as the transcription. Connectionist temporal classification (CTC) [5, 6], recurrent neural network transducer [7] and attention-based encoder-decoder (AED) [8, 9, 10] are three dominant approaches that enable E2E speech recognition. With the advantage of no conditional independence assumption over CTC, AED was first introduced to the speech area in [10] for phoneme recognition. In AED model, an encoder maps the input speech frames into high-level representations and a decoder predicts the current output symbol given the acoustic context vector and the embeddings of previously predicted symbols. An attention mechanism [9] aligns each decoder output with the encoded representations and computes the acoustic context vector. In [11, 12], AED is successfully applied to large vocabulary speech recognition and is recently reported to achieve superior performance to the conventional hybrid systems [13].

Initially, characters (graphemes) are commonly used as the output units for AED in E2E ASR [11, 12, 14]. Later on, people began to use words and subword units (WSUs) as the output since the perplexity of a word LM is lower than that of a character LM and the WSUs enable a stronger LM to be learned in the decoder of AED [15]. Modeling WSUs instead of characters enables the E2E system more directly targeting on the ASR output – word hypotheses. One popular type of WSUs is the word pieces model generated by iteratively combining two units out of the current inventory that increase the likelihood the most on the training data [13, 16]. Another kind of WSUs is the mixed-units [17] which include all the frequent words in the vocabulary as the major part and decompose each infrequent word into frequent words and left-over multi-character units. Mixed units were first introduced to address the issue of out-of-vocabulary (OOV) words [17] in a CTC-based E2E system. Recently, for AED-based ASR, mixed units outperform the characters and words as the output units [18]. With around 30k WSUs commonly used for US English, the WSU set is about 1000 times larger than the character set (about 30). Therefore, the WSU-based AED necessitates a much larger output layer with much more parameters but requires fewer decoding steps to generate the ASR results.

The WSU-based AED model learns a distinct embedding vector for each WSU from the text history and the speech signal by conditioning the decoder on previous WSU embeddings to predict the current WSU posteriors. Although good performance is achieved, the relationships among the WSUs are not explicitly modeled or well exploited. In many languages, the semantic relations of WSUs are not only determined by their relative positions and functionality in the sentences, but also are directly reflected in the similarity among

their spellings, i.e., the shared characters that form the WSUs.

To directly capture the additional morphological relationships among WSUs, we propose a character-aware (CA) AED in which only the character embeddings are learned through the E2E training and each WSU representation is generated by summarizing the embeddings of its constituent characters using a CA-recurrent neural network (RNN). With CA-AED, the embeddings of different WSUs that share the same character sequence are naturally bridged through the WSU-independent CA-RNN. A rare WSU representation can be better estimated through "assembling" the well-trained character embeddings. With the same output layer predicting WSU posteriors, CA-AED inherits the strong WSU discriminability in a large vocabulary and further improves AED through more sophisticated character-aware modeling of WSU embeddings.

Moreover, CA-AED significantly reduces the number of model parameters by replacing a large pool of WSU embeddings with a much smaller set of character embeddings. Therefore, CA-AED is expected to outperform conventional AED models with remarkably reduced model size and computational cost. A similar CA architecture based on convolutional neural network was proposed to improve the perplexity in neural language model [19] and has outperformed the word/morpheme-level long short-term memory network language model with fewer parameters.

Evaluated on 3400 hours Microsoft Cortana dataset (US English) with models of different sizes, the proposed CA-AED achieves up to a 11.9% relative word error rate (WER) improvement over a strong AED baseline with 27.1% fewer model parameters for word-piece output, and up to 8.5% relative WER gain with 29.3% fewer parameters for mixed-unit output.

2. ATTENTION-BASED ENCODER-DECODER (AED) MODEL FOR E2E ASR

In this work, we focus on improving the AED-based E2E speech recognition [10, 11, 12] with WSUs as the output units. AED models the conditional probability distribution $P(\mathbf{Y}|\mathbf{X})$ over sequences of output WSU labels $\mathbf{Y} = \{y_1, \ldots, y_T\}$ given a sequence of input speech frames $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_I\}$, where $y_t \in \mathbb{R}, t = 1, \ldots, T, \mathbf{x}_i \in \mathbb{R}^{d_x}, i = 1, \ldots, I$. To achieve E2E ASR, AED directly maps $\mathbf{X}$ to $\mathbf{Y}$ via an encoder, a decoder, an attention network and a WSU-embedding dictionary as shown in Fig. 1.

The encoder is an RNN which encodes the sequence of input speech frames $\mathbf{X}$ into a sequence of high-level features $\mathbf{H} = \{\mathbf{h}_1, \ldots, \mathbf{h}_I\}$ as follows and it resembles the role of an acoustic model in a traditional ASR system.

$$\mathbf{h}_i = \text{RNN}^{\text{enc}}(\mathbf{h}_{i-1}, \mathbf{x}_i) \quad (1)$$

where $\mathbf{h}_i \in \mathbb{R}^{d_h}$ represents the hidden state of the encoder RNN at current time i. With the encoder, $P(\mathbf{Y}|\mathbf{X})$ is equivalent to the probability over the output WSU sequences conditioned on the encoded high-level features $\mathbf{H}$, i.e., $P(\mathbf{Y}|\mathbf{H})$, as follows.

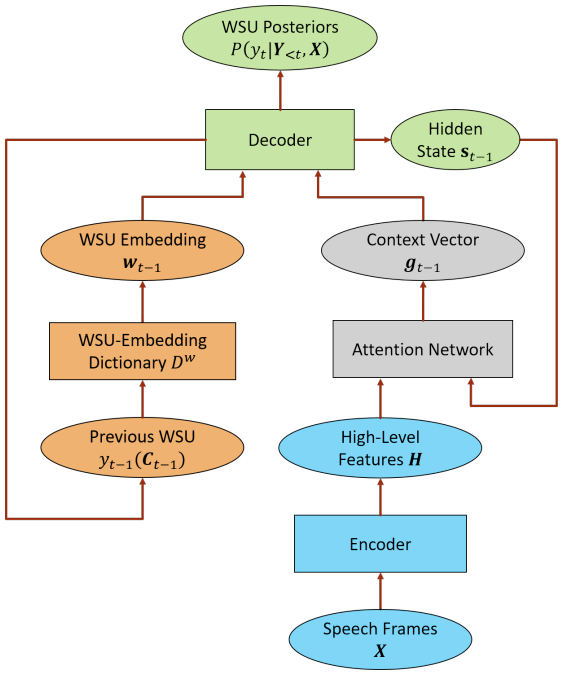

Fig. 1. The architecture of AED model for E2E ASR. The convolution network generating vector $\mathbf{f}_{t,i}$ is omitted for brevity.

$$P(\mathbf{Y}|\mathbf{X}) = P(\mathbf{Y}|\mathbf{H}) = \prod_{t=1}^{T} P(y_t|y_0, \ldots, y_{t-1}, \mathbf{H}) \quad (2)$$

We use a decoder to model $P(\mathbf{Y}|\mathbf{H})$. In $P(y_t|y_0, \ldots, y_{t-1}, \mathbf{H})$, the conditional dependence of y_t on $\mathbf{H}$ is captured through an acoustic context vector $\mathbf{g}_t \in \mathbb{R}^{d_h}$ obtained by a linear combination of all the encoded features $\mathbf{H}$ weighted by an attention probability vector $\mathbf{a}_t \in \mathbb{R}^I$ against $\mathbf{H}$. To estimate $\mathbf{a}_t$, a location-aware attention mechanism [10] is applied to determine which encoded features in $\mathbf{H}$ should the decoder attend to predict the output label y_t. Specifically, $\mathbf{a}_t$ is computed by normalizing the similarity scores, $z_{t,i}, i = 1, \ldots, I$, among the current hidden state $\mathbf{s}_t \in \mathbb{R}^{d_s}$ of the decoder RNN, each encoded feature $\mathbf{h}_i$ and the convoluted attention vector $\mathbf{f}_{t,i}$ as follows.

$$z_{t,i} = \mathbf{v}^\top \text{ReLU}(W_h \mathbf{h}_i + W_s \mathbf{s}_t + W_f \mathbf{f}_{t,i} + \mathbf{b}_z) \quad (3)$$

$$\mathbf{a}_t = \text{softmax}(\mathbf{z}_t) \quad (4)$$

$$\mathbf{g}_t = \sum_{i=1}^{I} a_{t,i} \mathbf{h}_i. \quad (5)$$

where the column vector $\mathbf{v} \in \mathbb{R}^k$, bias $\mathbf{b}_z \in \mathbb{R}^k$, the projection matrices $W_h \in \mathbb{R}^{k \times d_h}$, $W_s \in \mathbb{R}^{k \times d_s}$, $W_f \in \mathbb{R}^{k \times d_f}$

are all learnable parameters. $\mathbf{f}_{t,i} \in \mathbb{R}^{d_f}$ is generated by convolving the previous attention probability vector $\mathbf{a}_{t-1}$ with a matrix $F \in \mathbb{R}^{d_f \times r}$.

The conditional dependence of y_t on $y_0, \ldots, y_{t-1}$ is modeled by an RNN with a feedback connection from the decoder output of the previous time step to the input of the current step. Similar to an RNN language model [20], we maintain a large dictionary $\mathcal{D}^w$ which maps each WSU to an embedding vector and feed the previous WSU embedding instead of the label to the current input of the decoder. The WSU embeddings are learned jointly with the other parts of the AED in the training process. We denote the WSU-embedding sequence of $\mathbf{Y}$ as $\mathbf{W} = \{\mathbf{w}_1, \ldots, \mathbf{w}_T\}$. Therefore, at each time step t, the decoder RNN takes the sum of the previous WSU embedding $\mathbf{w}_{t-1}$ and the acoustic context vector $\mathbf{g}_{t-1}$ as the input to predict the conditional probability of each WSU, i.e., $P(u|y_0, \ldots, y_{t-1}, \mathbf{H}), u \in \mathbb{U}$, at the current time t as follows, where $\mathbb{U}$ is the set of all the WSUs:

$$\mathbf{s}_t = \text{RNN}^{\text{dec}}(\mathbf{s}_{t-1}, \mathbf{w}_{t-1} + \mathbf{g}_{t-1}) \quad (6)$$

$$[P(u|y_0, \ldots, y_{t-1}, \mathbf{H})]_{u \in \mathbb{U}} = \text{softmax}\left[W_y(\mathbf{s}_t + \mathbf{g}_t) + \mathbf{b}_y\right] \quad (7)$$

where bias $\mathbf{b}_y \in \mathbb{R}^k$ and the matrix $W_y \in \mathbb{R}^{d_y \times d_s}$ are learnable parameters. Note that d_y is the number of WSUs in the vocabulary and $d_h = d_s$ in our AED model.

To train the AED model, we maximize the conditional probability of the reference label sequences $\mathcal{Y} = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_N\}$ given their corresponding input speech sequences $\mathcal{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_N\}$ on the training corpus, which is equivalent to minimizing the total cross-entropy loss $\mathcal{L}_{\text{CE}}$ between the output of the decoder and the references at all the time steps below:

$$\begin{aligned}\mathcal{L}_{\text{CE}} &= -\sum_{n=1}^{N} \log P(\mathbf{Y}_n|\mathbf{X}_n) \\ &= -\sum_{n=1}^{N}\sum_{t=1}^{T_n} \log P(y_t^{(n)}|y_0^{(n)}, \ldots, y_{t-1}^{(n)}, \mathbf{H}_n) \end{aligned} \quad (8)$$

3. CHARACTER-AWARE (CA) AED MODEL FOR E2E ASR

As discussed in Section 2, a dictionary of WSU embeddings are learned through the E2E training of the AED model. The WSU embeddings exhibit the property that semantically and phonetically close words are likewise close in the induced vector space since the encoder and decoder RNNs are able to well capture the acoustic and the contextual relationships at the WSU-level. However, there is another level of connections that exist more apparently among different WSUs which the traditional AED models with WSUs output fail to capture - the morphological relationships. For example, in addition to the semantic and phonetic similarity, the words *note, noted, noting, notification, notify, notified, notifying, notifiable, noticeable, unnoticeable, unnoticeably* include the same sequence of characters "not-", and thus should have structurally correlated embeddings.

In a traditional WSU-based AED, the embeddings of the morphologically related WSUs are initialized and learned independently only through contextual WSUs and speech in a purely data-driven way. The robust estimation of so many WSU embeddings (e.g., around 30k) requires a huge amount of training data. The embeddings are poorly estimated for the WSUs that rarely occur in the training data. This is especially problematic for morphologically rich languages, e.g., in Finnish, a noun has 15 different cases; in French and Spanish, most verbs have more than 40 inflected forms.

To address this problem, we propose a CA-AED which directly makes use of the rich morphological relations among WSUs. As shown in Fig. 2, based on the existing components of AED, CA-AED introduces an additional *character-aware (CA) RNN* and replaces the WSU embeddings in $\mathcal{D}^w$ with WSU representations dynamically generated by this WSU-independent CA-RNN from *character embeddings*.

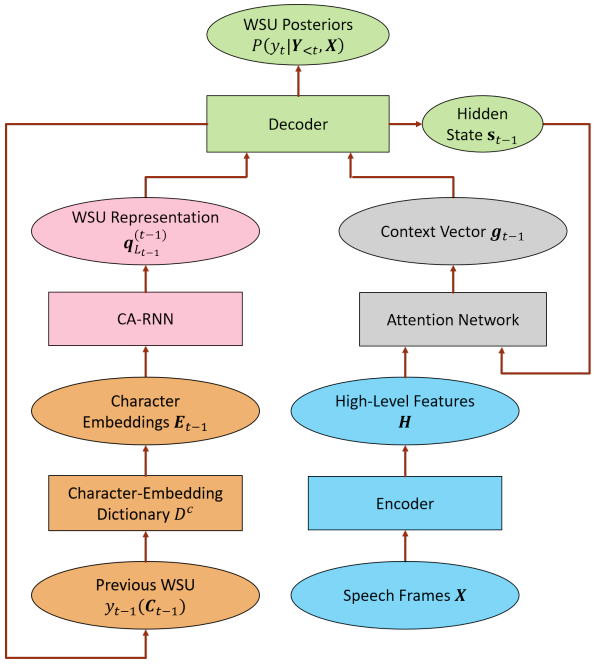

Fig. 2. The architecture of CA-AED model for E2E ASR. The convolution network generating vector $\mathbf{f}_{t,i}$ is omitted for brevity.

The WSU $\mathbf{y}_t$ is comprised of a character sequence $\mathbf{C}_t = \{c_1^{(t)}, \ldots, c_{L_t}^{(t)}\}$, where L_t is length of $\mathbf{y}_t$ in terms of characters. We construct a character-embedding dictionary $\mathcal{D}^c$ that maps each character into an embedding vector. By looking up $\mathcal{D}^c$, we encode $\mathbf{C}_t$ into a sequence of character embeddings $\mathbf{E}_t = \{\mathbf{e}_1^{(t)}, \ldots, \mathbf{e}_{L_t}^{(t)}\}$. In CA-AED, the CA-RNN takes the character-embedding sequence $\mathbf{E}_t$ of the WSU $\mathbf{y}_t$ as the input

and generate a representation for $\mathbf{y}_t$ using its last hidden state $\mathbf{q}_{L_t}^{(t)}$ as follows.

$$\mathbf{q}_l^{(t)} = \text{RNN}^{\text{char}}(\mathbf{q}_{l-1}^{(t)}, \mathbf{e}_l^{(t)}), \quad l = 1, \ldots, L_t \tag{9}$$

$\mathbf{q}_{L_t}^{(t)}$ is then used in place of the WSU embedding $\mathbf{w}_t$ as the input to the decoder RNN below, which further predicts the conditional probabilities of all possible WSUs via Eq. (7).

$$\mathbf{s}_t = \text{RNN}^{\text{dec}}(\mathbf{s}_{t-1}, \mathbf{q}_{L_{t-1}}^{(t-1)} + \mathbf{g}_{t-1}) \tag{10}$$

Fig. 3 shows an example of how CA-RNN works. The encoder and the attention network of CA-AED are exactly the same as the ones in AED. The character embeddings in $\mathcal{D}^c$ along with the CA-RNN are jointly trained with the other parts of CA-AED to minimize cross-entropy loss $\mathcal{L}_{\text{CE}}$ in Eq. (8).

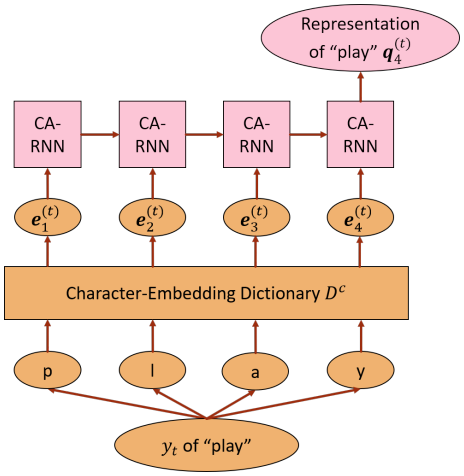

Fig. 3. An example of CA-RNN for generating the representation of WSU "play" with label y_t from the embeddings of its constituent characters.

With CA-AED, the WSUs sharing the same character substrings are naturally and explicitly correlated through the CA-RNN so that the embeddings of rare WSUs can be robustly estimated through assembling their constituent characters whose embeddings are accurately learned from abundant training samples. In addition, CA-AED inherits the strong discriminativity power among WSUs by predicting the same set of WSU units at the decoder output layer.

More importantly, CA-AED entails a much smaller number of character embeddings (e.g., about 30 in English) and a light-weight CA-RNN to be learned together with the encoder, decoder and attention network as opposed to a huge number of WSU embeddings (e.g., about 30k) with 1000 times more parameters in a conventional AED. Benefiting from modeling the additional morphological relations, the CA-AED is expected to generate better WSU embeddings for the decoder and improve the AED-based E2E ASR with significantly reduced number of parameters. The compression ratio becomes higher for a CA-AED model of smaller size since the character embeddings plus CA-RNN save a fixed number of parameters from WSU embeddings. Therefore, CA-AED has even higher potential for improving low-footprint AED models on mobile devices.

Before testing, all the WSU embeddings are pre-computed for once to form the WSU dictionary $\mathcal{D}^w$ by feed the constituent character embeddings of each WSU to the well-trained CA-RNN. Just as a conventional AED model described in Section 2, the pre-computed WSU dictionary is then looked up at each decoding step to provide the WSU embedding that the decoder is currently conditioned on to predict the next WSU output. Therefore, CA-AED does not increase the computational cost over the conventional AED model during evaluation.

Note that, during the WSU embedding computation for both training and testing, the CA-RNN resets its memory every time the first character embedding of a WSU is fed as the input. The CA-RNN thus only models the morphology of each WSU, i.e., the statistical relationships among internal characters, without performing any WSU-level language modeling.

4. EXPERIMENTS

We perform E2E ASR using AED and CA-AED with WSUs as the output units on a Microsoft Windows phone short message dictation (SMD) task.

4.1. Data Preparation

The training data consists of 3400 hours of Microsoft internal live US English Cortana utterances collected through a number of deployed speech services including voice search and SMD. The test data includes about 5600 utterances (6 hours). We explore both the word pieces and mixed units as the WSUs. We extract 80-dimensional log Mel filter bank (LFB) features from the speech signal in both the training and test set every 10 ms over a 25 ms window. We stack 3 consecutive frames and stride the stacked frame by 30 ms, to form a sequence of 240-dimensional input speech frames. We first generate 29190 word pieces as in [21] and 33755 mixed units as in [17] based on the training transcription and then produce both word-piece and mixed-unit label sequences serving as the training targets. We insert a special token `<space>` in between every two adjacent words to indicate word boundaries and add tokens `<sos>`, `<eos>` to the beginning and the end of each label sequence, respectively, to represent sentence boundaries.

4.2. AED Baseline System

We train a WSU-based AED model for E2E ASR. The encoder is a bi-directional gated recurrent units (GRU)-RNN [8, 22] with 4 or 6 hidden layers, each with 512 hidden units. Layer normalization [23] is applied for each encoder hidden layer. Units at the last hidden layer are used as the encoded high-level features. Each WSU is represented by a 512-dimensional embedding vector in $\mathcal{D}^w$. The decoder is a uni-directional GRU-RNN with 2 hidden layers, each with 512 hidden units. The decoders predicting word pieces and mixed units have 29190 and 33755 output units, respectively. During training, scheduled sampling [24] is applied to the decoder with a sampling probability starting at 0.0 and gradually increasing to 0.4 [13]. Dropout [25] with a probability of 0.1 is used in both encoder and decoder. We use 1-D convolution with a filter size of 15 and 512 output channels to generate $\mathbf{f}_{t,i}$ and fix W_h, W_s, W_f as identity matrices to compute the similarity scores $z_{t,i}$ in Eq. (5). A label-smoothed cross-entropy [26] loss is minimized during training. Greedy decoding is performed to generate the ASR transcription. We use PyTorch [27] for all the experiments.

As shown in Table 1, AED achieves 9.52% and 7.75% WERs with 4-layer and 6-layer encoders, respectively, by predicting word pieces at the output. By predicting mixed-unit output, the WERs decrease to 9.31% and 7.58% with 4-layer and 6-layer encoders, respectively. AED achieves better ASR performance with mixed-unit output.

WSU	System	N_e	WER	WERR	N_p	PRR
Word Piece	AED	4	9.52	-	44.9	-
		6	7.75	-	52.2	-
	CA-AED	4	8.39	11.9	32.7	27.1
		6	7.36	5.0	39.0	23.8
Mixed Unit	AED	4	9.31	-	49.5	-
		6	7.58	-	55.8	-
	CA-AED	4	8.52	8.5	35.0	29.3
		6	7.35	3.0	41.3	26.0

Table 1. The WER (%) performance of AED and CA-AED with different WSU output units for E2E ASR on a 3400 hours Microsoft Cortana dataset. N_e is the number of hidden layers in a encoder GRU and N_p (in million) is the total number of model parameters. WERR (%) and PRR (%) are the relative WER improvement and the parameter reduction rate of a CA-AED with respect to the AED with the same N_e.

4.3. Character-Aware (CA) AED System

We further train a CA-AED for E2E ASR with the same training data. The encoder, decoder and attention network in CA-AED have exactly the same architectures as the ones in AED. We map each of the 30 characters into a 256-dimensional embedding vector. CA-RNN is a GRU with 2 hidden layers and 512 hidden units for each layer. The last state of the top hidden layer of CA-RNN is used as the 512-dimensional WSU representation.

We vary the number of hidden layers in the encoder N_e to investigate the effectiveness of CA-AED for different model sizes with different parameter reduction rates (PRR). As shown in Table 1, for word-piece model, CA-AED achieves 8.39% and 7.36% WERs, respectively, with 4-layer and 6-layer encoders, which are 11.9% and 5.0% relative gains over the AED baseline system with 27.1% and 23.8% less model parameters, respectively. For mixed-unit model, CA-AED achieves 8.52% and 7.35% WERs, respectively, with 4-layer and 6-layer encoders, which are 8.5% and 3.0% relative improved over the AED baseline system with 29.3% and 26.0% reduction in model parameters, respectively.

As expected, PRR grows as the number of encoder layers decreases, indicating increased compression ratio. With a 4-layer encoder, CA-AED performs better for word-piece output, but with a 6-layer encoder, CA-AED achieves similar WERs for mixed-unit and word-piece outputs. With significantly reduced model parameters, CA-AED improves consistently over AED models for both word-piece and mixed-unit outputs. We also observe that the relative WER gain doubles when the encoder downsizes from 6 layers to 4 layers possible because the less accurate acoustic embeddings generated by a weaker encoder of smaller size make more room for the improvement from a more sophisticated WSU representation learned by the CA mechanism. This implies that CA-AED can achieve higher relative improvement upon corresponding AED model with a smaller number of parameters, and thus with a higher PRR. Therefore, CA-AED is even more effective in improving the accuracy of low-footprint AED models on mobile devices.

5. CONCLUSION

In this work, we propose a character-aware AED model for E2E ASR. The CA-AED explicitly models the morphological relations that exist prevalently among WSUs sharing the same sequence of characters. An additional CA-RNN is introduced to generate WSU representations by taking in the embeddings of their constituent characters. CA-AED makes prediction still at WSU level while entails only a few character embeddings be learned instead of a huge set of WSU embeddings.

Evaluated on a 3400 hours Microsoft Cortana dataset, CA-AED improves the WER of a traditional AED by up to 11.9% relatively with 27.1% fewer parameters with no increase of computational cost during testing. The gain is consistent for both word pieces or mixed units as the output units. CA-AED has great potential in improving small-footprint model on mobile devices, as the relative gain is higher over the AED models with fewer parameters.

6. REFERENCES

[1] T. Sainath, B. Kingsbury, B. Ramabhadran *et al.*, "Making deep belief networks effective for large vocabulary continuous speech recognition," in *Proc. ASRU*, 2011, pp. 30–35.

[2] N. Jaitly, P. Nguyen, A. Senior, and V. Vanhoucke, "Application of pretrained deep neural networks to large vocabulary speech recognition," in *Proc. INTERSPEECH*, 2012.

[3] G. Hinton, L. Deng, D. Yu *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[4] L. Deng, J. Li, J.-T. Huang *et al.*, "Recent advances in deep learning for speech research at Microsoft," in *ICASSP*. IEEE, 2013, pp. 8604–8608.

[5] A. Graves, S. Fernández, F. Gomez, and J. 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.

[6] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International Conference on Machine Learning*, 2014, pp. 1764–1772.

[7] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[8] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[9] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[10] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in neural information processing systems*, 2015, pp. 577–585.

[11] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4945–4949.

[12] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[13] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4774–4778.

[14] L. Lu, X. Zhang, and S. Renais, "On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5060–5064.

[15] A. Kannan, Y. Wu, P. Nguyen, T. N. Sainath, Z. Chen, and R. Prabhavalkar, "An analysis of incorporating an external language model into a sequence-to-sequence model," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1–5828.

[16] M. Schuster and K. Nakajima, "Japanese and korean voice search," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 5149–5152.

[17] J. Li, G. Ye, A. Das, R. Zhao, and Y. Gong, "Advancing acoustic-to-word CTC model," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5794–5798.

[18] Y. Gaur, J. Li, Z. Meng, and Y. Gong, "Acoustic-to-phrase end-to-end speech recognition," in *Proc. INTERSPEECH*. IEEE, 2019.

[19] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[20] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, and S. Khudanpur, "Recurrent neural network based language model," in *Eleventh annual conference of the international speech communication association*, 2010.

[21] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1715–1725.

[22] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[23] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[24] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 1171–1179.

[25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[26] J. Chorowski and N. Jaitly, "Towards better decoding and language model integration in sequence to sequence models," *CoRR*, vol. abs/1612.02695, 2016. [Online]. Available: http://arxiv.org/abs/1612.02695

[27] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.