Supervised Attention in Sequence-to-Sequence Models for Speech Recognition

Gene-Ping Yang, Hao Tang

School of Informatics, The University of Edinburgh

geneping.yang@ed.ac.uk, hao.tang@ed.ac.uk

Abstract

Sequence-to-sequence models have been extensively used in automatic speech recognition, and the attention mechanism plays a central role in localizing information from the input representation to generate output tokens. Even though the attention mechanism was invented with the intention of aligning parts of the input with output tokens, discovering alignments appears challenging during end-to-end training. In addition, several studies have shown that interpreting attention as alignments can be problematic. In this paper, we attempt to address these problems and propose to train sequence-to-sequence models with supervised attention, introducing a loss function that minimizes the differences between the attention and the alignments. Experiments on the Wall Street Journal data set show significantly improved performance in phonetic recognition.

Index Terms: speech recognition, attention mechanism, sequence-to-sequence models

1. Introduction

Sequence-to-sequence models are commonly used in speech recognition [1, 2, 3], text to speech [4], machine translation [5, 6], text summarization [7], and image captioning [8], transforming a sequence of input tokens to a sequence of output tokens. At a high level, a sequence-to-sequence model consists of three components, an encoder, the attention mechanism, and a decoder. The encoder encodes the input to a sequence of hidden vectors, each of which is given a weight by the attention mechanism. The decoder takes the weighted hidden vectors and its own hidden vector to produce the output token.

In speech recognition, the input is a sequence of acoustic features or frames, and the output is a sequence of discrete tokens, such as phones, word pieces, or words. Whether it is a phone, a word piece, or a word, there is a natural correspondence between an output token and a contiguous chunk of frames, commonly known as an alignment. The attention mechanism was designed to model alignments [5], and attention weights are intended to be interpreted as how much a frame belongs to the token to be produced. How well the attention weights correspond to alignments has been widely used as a qualitative evaluation for sequence-to-sequence models [3, 9, 10]. This belief spawns variants of attention mechanism [11, 12, 13] and training heuristics [10, 14], and has been further exploited for unsupervised word segmentation [15, 16].

However, attention weights do not always correspond well with alignments. Otherwise, the variants of attention mechanism and training heuristics would not have been necessary. The mismatch between attention weights and alignments is a natural consequence of end-to-end training—we only require that the weights sum to one; nothing requires the attention weights to correspond well with the actual alignments. In fact, there is an ongoing debate as to whether attention weights should be interpreted as alignments [17, 18, 19].

In this paper, we propose to train sequence-to-sequence models with supervised attention [20, 21, 22], guiding the attention weights with the actual alignments by minimizing the differences between the two. As a first step, we simply convert alignments into attention weights by putting uniform weights on the input hidden vectors for each corresponding output token and use Euclidean distance to measure the differences. By training with supervised attention, how well attention weights correspond to alignments becomes a learning problem and we can quantify the generalization of the correspondence. If the correspondence generalizes well, we can safely interpret attention weights as alignments. In addition, if the correspondence correlates well with the token error rates, we would expect the model to perform better in speech recognition.

2. Sequence-to-Sequence Models

Given an input sequence $x_1, x_2, .., .x_T$ of length T and target sequence $y_1, y_2, ..., y_K$ of length K, the conditional probability of the output sequence given the input sequence can be factorized (by the definition of conditional probability) as

$$P(y_{1:K}|x_{1:T}) = \prod_{k=1}^{K} P(y_k|x_{1:T}, y_{1:k-1}), \tag{1}$$

where $x_{1:t}$ is a shorthand for $x_1, \ldots, x_t$. A sequence-to-sequence model defines the probability distribution $P(y_k|x_{1:T}, y_{1:k-1})$, i.e., the probability of the next output token given the input and the past output tokens.

This probability distribution is computed as follows. First, the encoder transforms the input sequence into a sequence of hidden vectors $h_1, h_2, ..., h_{T'}$ of length T'.

$$h_{1:T'} = \text{Enc}(x_{1:T}) \tag{2}$$

Note that T' and T do not necessarily need to be the same. It is common to subsample the hidden vectors, for example, by a factor of four, i.e., $T' = \lfloor T/4 \rfloor$ [3, 23, 24]. For the k-th output token, the decoder first encodes the past output tokens into a vector

$$d_k = \text{Dec}(y_{1:k-1}). \tag{3}$$

The attention weights represented as a vector α_k are defined as

$$\alpha_{k,t} = \frac{\exp(d_k^\top h_t)}{\sum_{j=1}^{T'} \exp(d_k^\top h_j)}. \tag{4}$$

In other words, each attention weight is computed based on the similarity of a hidden vector h_t and d_k, followed by a Softmax, which constrains the weights to sum to one. The attention weights are then used to compute a context vector

$$c_k = \sum_{j=1}^{T'} \alpha_{k,j} h_j. \tag{5}$$

Finally, the context vector and the decoder hidden vector are used for prediction, i.e.,

$$P(y_{k,i}|x_{1:T}, y_{1:k-1}) = \frac{\exp(\phi_{k,i})}{\sum_{v=1}^{V} \exp(\phi_{k,v})} \tag{6}$$

where

$$\phi_k = W \begin{bmatrix} c_k \\ d_k \end{bmatrix}. \tag{7}$$

To train the model, for each pair $x_{1:T}$ and $y_{1:K}$ in the training set, we optimize the loss function

$$L_{\text{ce}} = -\ln P(y_{1:K}|x_{1:T}), \tag{8}$$

commonly known as teacher forcing [25, 26].

3. Supervised Attention

The attention weights α can be represented as a matrix, where each element $\alpha_{k,i}$ is the similarity score of the hidden vector d_k from the decoder and hidden vector h_i from the encoder.

An alignment is a sequence of 3-tuples $(s_1, t_1, y_1), \ldots, (s_K, t_K, y_K)$, where s_k is the start time and t_k is the end time, indicating the time span where the label y_k is in the input. In this paper, as a first step, we convert an alignment into the attention weights α^* by assigning uniform weights to the vectors between the corresponding start time and end time. Formally,

$$\alpha^*_{k,t} = \frac{1}{e_k - s_k} \mathbb{1}_{s_k \le t < e_k}, \tag{9}$$

where $\mathbb{1}_c$ is 1 if c is true, and 0 otherwise. Once an alignment is converted into attention weights, we use the Frobenius norm to measure the differences between α and α^*, and introduces an additional loss

$$L_{\text{attn}} = \|\alpha^* - \alpha\|_F^2 \tag{10}$$

to the objective for model training. When subsampling is involved, say by a factor of r, we simply sum the corresponding weights together. Specifically, the attention weights after subsampling is

$$\alpha^*_{k,t'} = \sum_{t=r(t'-1)}^{rt'} \frac{1}{e_k - s_k} \mathbb{1}_{s_k \le t < e_k} \tag{11}$$

The overall loss function is

$$L = L_{\text{ce}} + \gamma L_{\text{attn}}, \tag{12}$$

where γ is a hyperparameter.

4. Experiments

To validate the use of supervised attention, we conduct phonetic recognition experiments on the Wall Street Journal dataset. The training set `si284` is split into training and validation set with a ratio of $9:1$. We use the canonical pronunciations in `cmudict` as the gold standard. The label set, with stress markers removed, includes 39 phones and three special tokens for silence (`sil`) and noise (`spn` and `nsn`). Forced alignments produced by a speaker-adaptive HMM-GMM are used for supervised attention. We tune the hyperparameters on the validation set and report the final numbers on `dev93` and `eval92`. We use 40-dimensional Mel-scale spectrograms as input, without the first and second-order derivatives. Global (instead of speaker-dependent) mean and variance normalization is applied to the input features.

Table 1: *Phone error rates (%) on the testing sets* `dev93` *and* `eval92`*. The γ is 0.5 and the dropout rate is 0.4.*

	dev93	eval92
seq2seq	24.81	20.87
seq2seq + attn loss	12.09	8.65
seq2seq + dropout	15.31	11.79
seq2seq + dropout + attn loss	10.39	7.73

Our model setup follows the setting in [10]. The encoder is a 4-layer bidirectional LSTM, with 320 cells in each direction and each layer. Subsampling of factor two is done twice, one between after the 2nd layer and another after the 3rd layer of the encoder, resulting in $1/4$ of the original frame rate. The decoder consists of a phone embedding layer, a 1-layer unidirectional LSTM, and 2 fully connected layers after the LSTM. We use teacher forcing throughout training and greedy decoding during testing. We tune the value of γ and the dropout rate on the validation set.

Preliminary results are shown in Table 1. We see improvements when the models are trained with supervised attention. In addition, the improvement is still present after we regularize the models with dropout.

5. Conclusion and Future Work

Supervised attention is a simple approach to improve the performance of sequence-to-sequence models. One restriction is that it requires access to preprocessed alignments during training. However, accurate forced alignments are not difficult to obtain as long as the labels are sufficiently accurate.

There are many questions that remain open. The choice of representation of the alignments and the choice of loss functions remain to be explored. For example, instead of assigning weights uniformly, we could potentially use a Gaussian-like function to represent the alignments. The Frobenius norm does not respect that attention weights are probability mass functions, so other divergence functions that respect this property, such as KL divergence, might be more suitable.

Since the models are trained end-to-end, we are still in short of an approach to evaluate whether the models perform better when we supply the ground truth alignments. This prohibits us from analyzing where the problem is when the model is not performing well. One potential approach is to decouple the dependency of the attention weights, such as using the method of auxiliary coordinates [27]. It also has the extra benefits of decoupling the training of encoders and decoders.

In our experiments, we assume all utterances come with their corresponding alignments. The requirement can potentially be relaxed, and we might only need to train the models with a small set of alignments. Curriculum training might be useful too in this scenario: we first train the model on a small set of utterances with their alignments and train the model on a larger set of utterances as we normally would.

Though sequence-to-sequence models have been used extensively for speech recognition, there are many questions that remain to be answered. Supervised attention provides the first step towards that direction.

6. References

[1] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio, "End-to-end continuous speech recognition using attention-based recurrent nn: First results," in *NIPS 2014 Workshop on Deep Learning*, 2014.

[2] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *IEEE international conference on acoustics, speech and signal processing*, 2016.

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

[4] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Interspeech*, 2018.

[5] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2015.

[6] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Empirical Methods in Natural Language Processing*, 2015.

[7] R. Nallapati, B. Zhou, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," in *Conference on Computational Natural Language Learning*, 2016.

[8] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[9] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems*, 2015.

[10] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multi-task learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

[11] C. Chiu and C. Raffel, "Monotonic chunkwise attention," in *International Conference of Representation Learning*, 2018.

[12] S. Zhang, E. Loweimi, P. Bell, and S. Renals, "Windowed attention mechanisms for speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

[13] T.-T. Nguyen, X.-P. Nguyen, S. Joty, and X. Li, "Differentiable window for dynamic local attention," in *Annual Meeting of the Association for Computational Linguistics*, 2020.

[14] A. Hannun, A. Lee, Q. Xu, and R. Collobert, "Sequence-to-sequence speech recognition with time-depth separable convolutions," in *Interspeech*, 2019.

[15] P. Godard, M. Z. Boito, L. Ondel, A. Berard, F. Yvon, A. Villavicencio, and L. Besacier, "Unsupervised word segmentation from speech with attention," in *Interspeech*, 2018.

[16] M. Z. Boito, A. Villavicencio, and L. Besacier, "Empirical evaluation of sequence-to-sequence models for word discovery in low-resource settings," in *Interspeech*, 2019.

[17] S. Serrano and N. A. Smith, "Is attention interpretable?" in *Annual Meeting of the Association for Computational Linguistics*, 2019.

[18] S. Wiegreffe and Y. Pinter, "Attention is not not explanation," in *Empirical Methods in Natural Language Processing*, 2019.

[19] S. Jain and B. C. Wallace, "Attention is not explanation," in *North American Chapter of the ACL*, 2019.

[20] L. Liu, M. Utiyama, A. Finch, and E. Sumita, "Neural machine translation with supervised attention," in *International Conference on Computational Linguistics*, 2016.

[21] H. Mi, Z. Wang, and A. Ittycheriah, "Supervised attentions for neural machine translation," in *Empirical Methods in Natural Language Processing*, 2016.

[22] S. Nadig, S. Chakraborty, A. Shah, C. Sharma, V. Ramasubramanian, and S. Rao, "Jointly learning to align and transcribe using attention-based alignment and uncertainty-to-weigh losses," in *International Conference on Signal Processing and Communications*, 2020.

[23] V. Vanhoucke, M. Devin, and G. Heigold, "Multiframe deep neural networks for acoustic modeling," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.

[24] Y. Miao, J. Li, Y. Wang, S.-X. Zhang, and Y. Gong, "Simplifying long short-term memory acoustic models for fast training and decoding," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016.

[25] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural computation*, vol. 1, no. 2, pp. 270–280, 1989.

[26] A. Lamb, A. Goyal, Y. Zhang, S. Zhang, A. Courville, and Y. Bengio, "Professor forcing: a new algorithm for training recurrent networks," in *Advances in Neural Information Processing Systems*, 2016.

[27] M. Carreira-Perpinan and W. Wang, "Distributed optimization of deeply nested systems," in *Artificial Intelligence and Statistics*. PMLR, pp. 10–19.