Neurocomputing 000 (2018) 1–6

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Fine-grained attention mechanism for neural machine translation

Heeyoul Choi [a,*], Kyunghyun Cho [b], Yoshua Bengio [c]

[a] Handong Global University, Pohang, Republic of Korea
[b] New York University, NY, USA
[c] University of Montreal, QC, Canada

ARTICLE INFO

Article history:
Received 4 May 2017
Revised 13 November 2017
Accepted 4 January 2018
Available online xxx

Communicated by Tie-Yan Liu

Keywords:
Neural machine translation
Attention mechanism
Fine-grained attention

ABSTRACT

Neural machine translation (NMT) has been a new paradigm in machine translation, and the attention mechanism has become the dominant approach with the state-of-the-art records in many language pairs. While there are variants of the attention mechanism, all of them use only temporal attention where one scalar value is assigned to one context vector corresponding to a source word. In this paper, we propose a fine-grained (or 2D) attention mechanism where each dimension of a context vector will receive a separate attention score. In experiments with the task of En-De and En-Fi translation, the fine-grained attention method improves the translation quality in terms of BLEU score. In addition, our alignment analysis reveals how the fine-grained attention mechanism exploits the internal structure of context vectors.

1. Introduction

Neural machine translation (NMT), which is an end-to-end approach to machine translation [1–3], has widely become adopted in machine translation research, as evidenced by its success in a recent WMT'16 translation task [4,5]. The attention-based approach, proposed by [3], has become the dominant approach among others, which has resulted in state-of-the-art translation qualities on, for instance, En-Fr [6], En-De [4,7], En-Zh [8], En-Ru [9] and En-Cz [9,10]. These recent successes are largely due to better handling a large target vocabulary [6,9–11], incorporating a target-side monolingual corpus [12,13] and advancing the attention mechanism [14–16].

We notice that all the variants of the attention mechanism, including the original one by [3], are *temporal* in that it assigns a scalar attention score for each context vector, which corresponds to a source symbol. In other words, all the dimensions of a context vector are treated equally. This is true not only for machine translation, but also for other tasks on which the attention-based task was evaluated. For instance, the attention-based neural caption generation by [17] assigns a scalar attention score for each context vector, which corresponds to a spatial location in an input image, treating all the dimensions of the context vector equally. See [18] for more of such examples.

On the other hand, in [19], it was shown that word embedding vectors have more than one notions of similarities by analyzing the local chart of the manifold that word embedding vectors reside. Also, by contextualization of word embedding, each dimension of the word embedding vectors could play different role according to the context, which, in turn, led to better translation qualities in terms of the BLEU scores.

Inspired by the contextualization of word embedding, in this paper, we propose to extend the attention mechanism so that each dimension of a context vector will receive a separate attention score. This enables finer-grained attention, meaning that the attention mechanism may choose to focus on one of many possible interpretations of a single word encoded in the high-dimensional context vector [19,20]. This is done by letting the attention mechanism output as many scores as there are dimensions in a context vectors, contrary to the existing variants of attention mechanism which returns a single scalar per context vector.

We evaluate and compare the proposed fine-grained attention mechanism on the tasks of En-De and En-Fi translation. The experiments reveal that the fine-grained attention mechanism improves the translation quality up to +1.4 BLEU. Our qualitative analysis found that the fine-grained attention mechanism indeed exploits the internal structure of each context vector.

2. Background: attention-based neural machine translation

The attention-based neural machine translation (NMT) from [3] computes a conditional distribution over translations given a source sentence $X = (w_1^x, w_2^x, \ldots, w_T^x)$:

$$p(Y = (w_1^y, w_2^y, \ldots, w_{T'}^y)|X). \quad (1)$$

* Corresponding author.
E-mail address: hchoi@handong.edu (H. Choi).

https://doi.org/10.1016/j.neucom.2018.01.007
0925-2312/

This is done by a neural network that consists of an encoder, a decoder and the attention mechanism.

The encoder is often implemented as a bidirectional recurrent neural network (RNN) that reads the source sentence word-by-word. Before being read by the encoder, each source word w_t^x is projected onto a continuous vector space:

$$\boldsymbol{x}_t = \boldsymbol{E}^x[\cdot, w_t^x], \tag{2}$$

where $\boldsymbol{E}^x[\cdot, w_t^x]$ is w_t^xth column vector of $\boldsymbol{E}_x \in \mathbb{R}^{E\times|V|}$, a source word embedding matrix, where E and $|V|$ are the word embedding dimension and the vocabulary size, respectively.

The resulting sequence of word embedding vectors is then read by the bidirectional encoder recurrent network which consists of forward and reverse recurrent networks. The forward recurrent network reads the sequence in the left-to-right order while the reverse network reads it right-to-left:

$$\overrightarrow{\boldsymbol{h}}_t = \overrightarrow{\phi}(\overrightarrow{\boldsymbol{h}}_{t-1}, \boldsymbol{x}_t),$$
$$\overleftarrow{\boldsymbol{h}}_t = \overleftarrow{\phi}(\overleftarrow{\boldsymbol{h}}_{t+1}, \boldsymbol{x}_t),$$

where the initial hidden states $\overrightarrow{\boldsymbol{h}}_0$ and $\overleftarrow{\boldsymbol{h}}_{T+1}$ are initialized as all-zero vectors or trained as parameters. The hidden states from the forward and reverse recurrent networks are concatenated at each time step t to form an annotation vector $\boldsymbol{h}$:

$$\boldsymbol{h}_t = \left[\overrightarrow{\boldsymbol{h}}_t; \overleftarrow{\boldsymbol{h}}_t\right].$$

This concatenation results in a context C that is a tuple of annotation vectors:

$$C = \{\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_T\}.$$

The recurrent activation functions $\overrightarrow{\phi}$ and $\overleftarrow{\phi}$ are in most cases either long short-term memory units (LSTM, [21]) or gated recurrent units (GRU, [22]).

The decoder consists of a recurrent network and the attention mechanism. The recurrent network is a unidirectional language model to compute the conditional distribution over the next target word given all the previous target words and the source sentence:

$$p(w_{t'}^y | w_{<t'}^y, X).$$

By multiplying this conditional probability for all the words in the target, we recover the distribution over the full target translation in Eq. (1).

The decoder recurrent network maintains an internal hidden state $\boldsymbol{z}_{t'}$. At each time step t', it first uses the attention mechanism to select, or weight, the annotation vectors in the context tuple C. The attention mechanism, which is a feedforward neural network, takes as input both the previous decoder hidden state, and one of the annotation vectors, and returns a relevant score $e_{t',t}$:

$$e_{t',t} = f_{\text{Att}}(\boldsymbol{z}_{t'-1}, \boldsymbol{h}_t), \tag{3}$$

which is referred to as *score function* [9,14]. The function f_{Att} can be implemented by fully connected neural networks with a single hidden layer where $tanh()$ can be applied as activation function. These relevance scores are normalized to be positive and sum to 1,

$$\alpha_{t',t} = \frac{\exp(e_{t',t})}{\sum_{k=1}^{T}\exp(e_{t',k})}. \tag{4}$$

We use the normalized scores to compute the weighted sum of the annotation vectors

$$\boldsymbol{c}_{t'} = \sum_{t=1}^{T}\alpha_{t',t}\boldsymbol{h}_t, \tag{5}$$

which will be used by the decoder recurrent network to update its own hidden state by

$$\boldsymbol{z}_{t'} = \phi_z(\boldsymbol{z}_{t'-1}, \boldsymbol{y}_{t'-1}, \boldsymbol{c}_{t'}).$$

Similarly to the encoder, ϕ_z is implemented as either an LSTM or GRU. $\boldsymbol{y}_{t'-1}$ is a target-side word embedding vector obtained by

$$\boldsymbol{y}_{t'-1} = \boldsymbol{E}^y[\cdot, w_{t'-1}^y],$$

similarly to Eq. (2).

The probability of each word i in the target vocabulary V' is computed by

$$p(w_{t'}^y = i | w_{<t'}^y, X) = \phi\left(\boldsymbol{W}_i^y \boldsymbol{z}_{t'} + c_i\right),$$

where $\boldsymbol{W}_i^y$ is the ith row vector of $\boldsymbol{W}^y \in \mathbb{R}^{|V|\times dim(\boldsymbol{z}_{t'})}$ and c_i is the bias.

The NMT model is usually trained to maximize the log-probability of the correct translation given a source sentence using a large training parallel corpus. This is done by stochastic gradient descent, where the gradient of the log-likelihood is efficiently computed by the backpropagation algorithm.

2.1. Variants of attention mechanism

Since the original attention mechanism was proposed as in Eq. (3) [3], there have been several variants [14].

[14] presented a few variants of the attention mechanism on the sequence-to-sequence model [2]. Although their work cannot be directly compared to the attention model in [3], they introduced a few variants for score function of attention model – content based and location based score functions. Their score functions still assign a single value for the context vector $\boldsymbol{h}_t$ as in Eq. (3).

Another variant is to add the target word embedding as input for the score function [6,9] as follows:

$$e_{t',t} = f_{\text{AttY}}(\boldsymbol{z}_{t'-1}, \boldsymbol{h}_t, \boldsymbol{y}_{t'-1}), \tag{6}$$

and the score is normalized as before, which leads to $\alpha_{t',t}$, and f_{AttY} can be a fully connected neural network as Eq. (3) with different input size. This method provides the score function additional information from the previous word. In training, teacher forced true target words can be used, while in test the previously generated word is used. In this variant, still a single score value is given to the context vector $\boldsymbol{h}_t$.

3. Fine-grained attention mechanism

All the existing variants of attention mechanism assign a single scalar score for each context vector $\boldsymbol{h}_t$. We however notice that it is not necessary to assign a single score to the context at a time, and that it may be beneficial to assign a score for each *dimension* of the context vector, as each dimension represents a different perspective into the captured internal structure. In [19], it was shown that each dimension in word embedding could have different meaning and the *context* could enrich the meaning of each dimension in different ways. The insight in this paper is similar to [19], except two points: (1) focusing on the encoded representation rather than word embedding, and (2) using 2 dimensional attention rather than the context of the given sentence.

We therefore propose to extend the score function f_{Att} in Eq. (3) to return a set of scores corresponding to the dimensions of the context vector $\boldsymbol{h}_t$. That is,

$$e_{t',t}^d = f_{\text{AttY2D}}^d(\boldsymbol{z}_{t'-1}, \boldsymbol{h}_t, \boldsymbol{y}_{t'-1}), \tag{7}$$

where $e_{t',t}^d$ is the score assigned to the dth dimension of the tth context vector $\boldsymbol{h}_t$ at time t'. Here, f_{AttY2D} is a fully connected neural

Please cite this article as: H. Choi et al., Fine-grained attention mechanism for neural machine translation, Neurocomputing (2018), https://doi.org/10.1016/j.neucom.2018.01.007