DL4MT-Tutorial: Conditional Gated Recurrent Unit with Attention Mechanism

Orhan Firat Kyunghyun Cho

May 15, 2016

This document describes the *gru_cond_layer* used in Session 2 and Session 3.

Given a source sequence $(x_1, \ldots, x_{T_x})$ of length T_x and a target sequence $(y_1, \ldots, y_{T_y})$, let $\mathbf{h}_i$ be the annotation of the source symbol at position i, obtained by concatenating the forward and backward encoder RNN hidden states, $\mathbf{h}_i = [\overrightarrow{\mathbf{h}}_i; \overleftarrow{\mathbf{h}}_i]$. A conditional GRU with attention mechanism, cGRU_{att}, uses it's previous hidden state $\mathbf{s}_{j-1}$, the whole set of source annotations $\mathrm{C} = \{\mathbf{h}_i, \ldots, \mathbf{h}_{T_x}\}$ and the previously decoded symbol y_{j-1} in order to update it's hidden state $\mathbf{s}_j$, which is further used to decode symbol y_j at position j,

$$\mathbf{s}_j = \text{cGRU}_{\text{att}}\left(\mathbf{s}_{j-1}, y_{j-1}, \mathrm{C}\right). \tag{1}$$

Internals The conditional GRU layer with attention mechanism, cGRU_{att}, consists of three components, two recurrent cells and an attention mechanism ATT in between. First recurrent cell REC_1, combines the previous decoded symbol y_{j-1} and previous hidden state $\mathbf{s}_{j-1}$ in order to generate an intermediate representation $\mathbf{s}'_j$ with the following formulations:

$$\mathbf{s}'_j = \text{REC}_1\left(y_{j-1}, \mathbf{s}_{j-1}\right) = (1 - \mathbf{z}'_j) \odot \underline{\mathbf{s}}'_j + \mathbf{z}'_j \odot \mathbf{s}_{j-1}, \tag{2}$$

$$\underline{\mathbf{s}}'_j = \tanh\left(\mathbf{W}'\mathbf{E}[y_{j-1}] + \mathbf{r}'_j \odot (\mathbf{U}'\mathbf{s}_{j-1})\right), \tag{3}$$

$$\mathbf{r}'_j = \sigma\left(\mathbf{W}'_r\mathbf{E}[y_{j-1}] + \mathbf{U}'_r\mathbf{s}_{j-1}\right), \tag{4}$$

$$\mathbf{z}'_j = \sigma\left(\mathbf{W}'_z\mathbf{E}[y_{j-1}] + \mathbf{U}'_z\mathbf{s}_{j-1}\right), \tag{5}$$

where $\mathbf{E}$ is the target word embedding matrix, $\underline{\mathbf{s}}'_j$ is the proposal intermediate representation, $\mathbf{r}'_j$ and $\mathbf{z}'_j$ being the reset and update gate activations. In this formulation, $\mathbf{W}'$, $\mathbf{U}'$,

$\mathbf{W}'_r$, $\mathbf{U}'_r$, $\mathbf{W}'_z$, $\mathbf{U}'_z$ are trained model parameters[1] tanh and σ are hyperbolic tangent and logistic sigmoid activation functions respectively.

The attention mechanism ATT, inputs the entire context set C along with intermediate hidden state $\mathbf{s}'_j$ in order to compute the context vector $\mathbf{c}_j$ as follows:

$$\mathbf{c}_j = \text{ATT}\left(\text{C}, \mathbf{s}'_j\right) = \sum_{i}^{T_x} \alpha_{ij}\mathbf{h}_i, \tag{6}$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{Tx} \exp(e_{kj})}, \tag{7}$$

$$e_{ij} = \mathbf{v}_a^{\intercal} \tanh\left(\mathbf{U}_a \mathbf{s}_j^{(1)} + \mathbf{W}_a \mathbf{h}_i\right), \tag{8}$$

where α_{ij} is the normalized alignment weight between source symbol at position i and target symbol at position j and $\mathbf{v}_a, \mathbf{U}_a, \mathbf{W}_a$ are the trained model parameters.

Finally, the second recurrent cell REC_2, generates $\mathbf{s}_j$, the hidden state of the cGRU_{att}, by looking at intermediate representation $\mathbf{s}'_j$ and context vector $\mathbf{c}_j$ with the following formulations:

$$\mathbf{s}_j = \text{REC}_2\left(\mathbf{s}'_j, \mathbf{c}_j\right) = (1 - \mathbf{z}_j) \odot \underline{\mathbf{s}}_j + \mathbf{z}_j \odot \mathbf{s}'_j, \tag{9}$$

$$\underline{\mathbf{s}}_j = \tanh\left(\mathbf{W}\mathbf{c}_j + \mathbf{r}_j \odot (\mathbf{U}\mathbf{s}'_j)\right), \tag{10}$$

$$\mathbf{r}_j = \sigma\left(\mathbf{W}_r \mathbf{c}_j + \mathbf{U}_r \mathbf{s}'_j\right), \tag{11}$$

$$\mathbf{z}_j = \sigma\left(\mathbf{W}_z \mathbf{c}_j + \mathbf{U}_z \mathbf{s}'_j\right), \tag{12}$$

similarly, $\underline{\mathbf{s}}_j$ being the proposal hidden state, $\mathbf{r}_j$ and $\mathbf{z}_j$ being the reset and update gate activations with the trained model parameters $\mathbf{W}, \mathbf{U}, \mathbf{W}_r, \mathbf{U}_r, \mathbf{W}_z, \mathbf{U}_z$.

[1] All the biases are omitted for simplicity.