A Parsing-Reading-Predict Network

The forward pass of PRPN is described here. Parsing-Reading-Predict Network contains three components.

A.1 Parsing Network

The syntactic distance between a given token represented as word embedding e_i and the previous token e_{i-1} is calculated by convolution kernel over a set of previous tokens e_{i-L}, e_{i-L+1},..., e_i. Mathematically, syntactic distance d_i between e_{i-1} and e_i is computed as:

$$h_i = \text{ReLU}(W_c \begin{bmatrix} e_{i-L} \\ e_{i-L+1} \\ \cdots \\ e_i \end{bmatrix} + b_c) \quad (1)$$

$$d_i = \text{ReLU}\left(W_d h_i + b_d\right) \quad (2)$$

where W_c, b_c are the kernel parameters. W_d and b_d can be seen as another convolutional kernel with window size 1, convolved over h_i's. The kernel window size L, that indicates how far back into the history node e_i can reach while computing its syntactic distance d_i, is called the *look-back range*. For the tokens in the beginning of the sequence, $L-1$ zero vectors are padded to the front of the sequence. This will produce $K-1$ distances for sequence length of K.

To determine the closest word x_j that has larger syntactic relationship than d_j for time step t, α_j^t is defined as:

$$\alpha_j^t = \frac{\text{hardtanh}\left((d_t - d_j)\cdot\tau\right) + 1}{2} \quad (3)$$

where τ is the temperature parameter that controls the sensitivity of α_j^t to the differences between distances.

The soft gate values that will be used for language modeling are then computed as:

$$g_i^t = \mathbf{P}(l_t \leq i) = \prod_{j=i+1}^{t-1} \alpha_j^t \quad (4)$$

A.2 Reading Network

The reading network uses Long Short-Term Memory-Network that maintains two sets of vectors as the memory states: a hidden tape $H_{t-1} = (h_{t-N_m}, ..., h_{t-1})$, and a memory tape $C_{t-1} = (c_{t-L}, ..., c_{t-1})$, where N_m is the upper bound for the memory span. Hidden state m_i is represented by a tuple of two vectors (h_i, c_i).

At each step, the reading network links the current token to all the previous tokens that are syntactically similar:

$$k_t = W_h h_{t-1} + W_x x_t \quad (5)$$

$$\tilde{s}_i^t = \text{softmax}(\frac{h_i k_t^{\mathrm{T}}}{\sqrt{\delta_k}}) \quad (6)$$

where, δ_k is the dimension of the hidden state. The structured intra-attention weight is defined based on the gates in Eq.4:

$$s_i^t = \frac{g_i^t \tilde{s}_i^t}{\sum_i g_i^t} \quad (7)$$

An adaptive summary vector for the previous hidden tape and memory denoted by $\tilde{h}_t$ and $\tilde{c}_t$ are computed as:

$$\begin{bmatrix} \tilde{h}_t \\ \tilde{c}_t \end{bmatrix} = \sum_{i=1}^{t-1} s_i^t \cdot m_i = \sum_{i=1}^{t-1} s_i^t \cdot \begin{bmatrix} h_i \\ c_i \end{bmatrix} \quad (8)$$

The Reading Network then takes x_t, $\tilde{c}_t$ and $\tilde{h}_t$ as input, computes the values of c_t and h_t by the LSTM recurrent update (Hochreiter and Schmidhuber, 1996). Then the *write* operation concatenates h_t and c_t to the end of hidden and memory tape.

A.3 Predict Network

Predict Network models the probability distribution of next word x_{t+1}, based on hidden states $m_0, ..., m_t$, and gates $g_0^{t+1}, ..., g_t^{t+1}$:

$$p(x_{t+1}|x_{t<t+1}) \approx p(x_{t+1}; f(m_{t<t+1}, g_{t<t+1}^{t+1})) \quad (9)$$

Since the model cannot observe x_{t+1} at time step t, a temporary estimation of d_{t+1} is computed using $x_{t-L}, ..., x_t$:

$$d'_{t+1} = \text{ReLU}(W'_d h_t + b'_d) \quad (10)$$

The corresponding $\{\alpha^{t+1}\}$ and $\{g_i^{t+1}\}$ for Eq.9 are computed after that. $f(\cdot)$ function is parameterized as:

$$f(m_t, ..., m_t, g_0^{t+1}, ..., g_t^{t+1}) = \hat{f}([h_{l:t-1}, h_t]) \quad (11)$$

where $h_{l:t-1}$ is an adaptive summary of $h_{l_{t+1}\leq i\leq t-1}$, output by structured attention

controlled by $g_0^{t+1}, ..., g_{t-1}^{t+1}$. $\hat{f}(\cdot)$ could be a simple feed-forward MLP, or more complex architecture, like ResNet, to add more depth to the model.

A.4 Sample Parses from the model with the best F1 score

In Figure A.1, we report a few example parses from the model with the best F1 score (PRPN-UP trained on AllNLI) among our experiments, compared with the ground truth PTB parses.

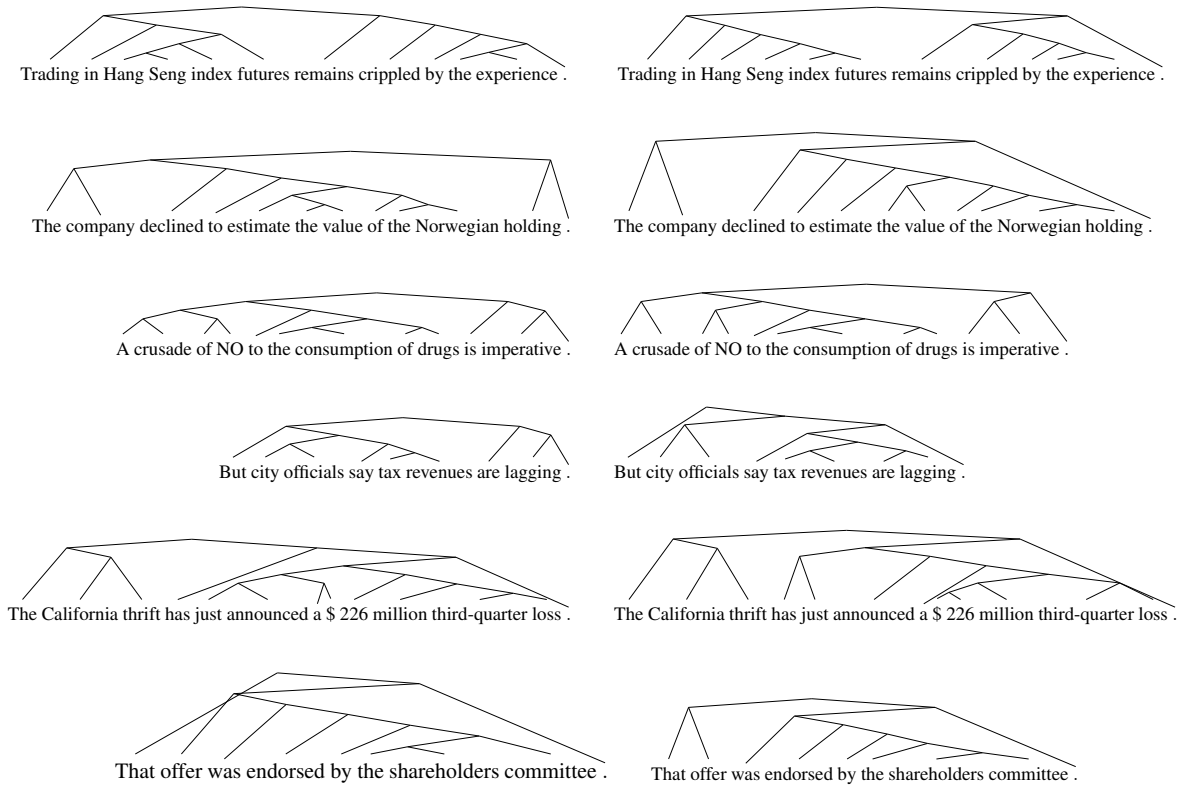

Figure A.1: *Left* Parses from PRPN-LM trained on AllNLI (stopping criterion: language modeling). *Right* Ground truth parses from Penn Treebank.