清華大學

Tsinghua University

Department of Computer Science and Technology

Natural Language Processing

Assignment 2

Sahand Sabour

2020280401

1 Gradient Calculation

Assuming that we are given a predicted word vector v_c for the center word c in skip-gram, and word prediction is made with the following Softmax function:

$$y_o = p(o|c) = \frac{exp(u_o^T v_c)}{\sum_{w=1}^{W} exp(u_w^T v_c)} \tag{1}$$

where w denotes the w-th word and $u_w = (w = 1, ..., W)$ are the "output" word vectors for all words in the vocabulary. In addition, assuming that the cross entropy loss is used, we would have:

$$\mathcal{L} = -\sum_{w=1}^{W} t_i log(y_i) = -log(p(o|c)) = -log(exp(u_o^T v_c)) + log(\sum_{w=1}^{W} exp(u_w^T v_c)) \tag{2}$$

Where t is the label and is either 1 or 0 since the input is one-hot encoded. Therefore, there would only be one element of the sum that is non-zero (for the expected word o). Further simplifying the loss function gives:

$$\mathcal{L} = -u_o^T v_c + log(\sum_{w=1}^{W} exp(u_w^T v_c)) \tag{3}$$

Accordingly, we would derive the gradient from this loss as follows:

$$\begin{aligned} \frac{\delta \mathcal{L}}{\delta v_c} &= -\frac{\delta u_o^T v_c}{\delta v_c} + \frac{\delta log(\sum_{w=1}^{W} exp(u_w^T v_c))}{\delta v_c} \\ &= -u_o^T + (\frac{1}{\sum_{w=1}^{W} exp(u_w^T v_c)})(exp(u_w^T v_c))(u_w^T) \\ &= \sum_{w=1}^{W} p(o|c) u_w^T - u_o^T \end{aligned} \tag{4}$$

2 Word2vec Implementation

3 Word2vec Improvement