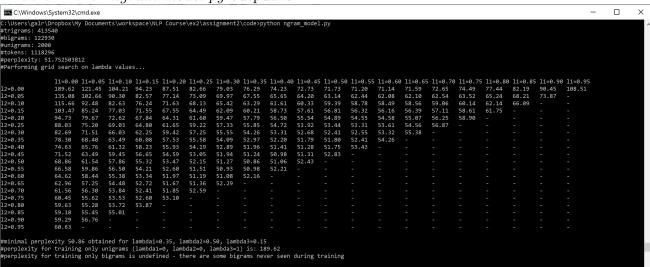
Home Assignment 2: Language Models

Daniel Solomon ID: 206335283 Gal Ron ID: 304856529

April 24, 2017

Problem 1.b. ngram_model.py output is:



Problem 2.a. We want to derive $CE(y, \hat{y}) = -\Sigma_i y_i log(\hat{y}_i)$ with respect to vector θ . First let's derive it for each co-ordinate:

$$\frac{\partial CE}{\partial \theta_k} = -\frac{\partial log(\hat{y_i})}{\partial \theta_k} = -\frac{\partial}{\partial \theta_k} log(\frac{\exp \theta_i}{\Sigma_j \exp \theta_j})$$

Assume k = i, we get:

$$-\frac{\frac{\exp\left(\theta_{k}\right)\Sigma_{j}\exp\left(\theta_{j}\right)-\exp\left(\theta_{k}\right)^{2}}{\left(\Sigma_{j}\exp\left(\theta_{j}\right)\right)^{2}}}{\frac{\exp\left(\theta_{k}\right)}{\Sigma_{j}\exp\left(\theta_{j}\right)}}=-\frac{\Sigma_{j}\exp\left(\theta_{j}\right)-\exp\left(\theta_{k}\right)}{\Sigma_{j}\exp\left(\theta_{j}\right)}=\frac{\exp\left(\theta_{k}\right)-\Sigma_{j}\exp\left(\theta_{j}\right)}{\Sigma_{j}\exp\left(\theta_{j}\right)}=\frac{\exp\theta_{k}}{\Sigma_{j}\exp\theta_{j}}-1=\hat{y_{k}}-1.$$

Assume $k \neq i$, we get:

$$-\frac{\frac{-\exp(\theta_i)\exp(\theta_k)}{(\Sigma_j\exp(\theta_j))^2}}{\frac{\exp(\theta_k)}{\Sigma_j\exp(\theta_j)}} = \frac{\exp\theta_i}{\Sigma_j\exp\theta_j} = \hat{y_i}.$$

Bottom line we can generalize it as:

$$\frac{\partial CE}{\partial \theta} = \hat{y} - y.$$

Problem 2.b. For one's comfort let's denote $z_1 = xW_1 + b_1$ and $z_2 = h(W_2 + b_2)$. Using chain rule we get:

$$\frac{\partial CE}{\partial x} = \frac{\partial CE}{\partial z_2} \frac{\partial z_2}{\partial x} \stackrel{(i)}{=} (\hat{y} - y) \frac{\partial z_2}{\partial h} \frac{\partial h}{\partial x} \stackrel{(ii)}{=} (\hat{y} - y) W_2^T \frac{\partial h}{\partial z_1} \frac{\partial z_1}{\partial x} \stackrel{(iii)}{=} (\hat{y} - y) W_2^T \circ \sigma'(z_1) W_1^T.$$

Explanations:

- (i) Left derivative calculated by 2.a, right derivative chain rule.
- (ii) Left derivative calculated by simple calculus, right derivative chain rule.
- (iii) Left derivative calculated by simple calculus so as right derivative (notice that o sign represents multiplication member by member as we saw in class).

Problem 2.d. *neural_lm.py* output is:

params: 104550

train examples: 1118296 training took 324 seconds

dev perplexity: 112.967665327