Assignment 3: Dependency Parsing

1. Machine Learning & Neural Networks

(a) Adam Optimizer

i. First, Adam uses a trick called *momentum* by keeping track of **m**, a rolling average of the gradients:

$$m \leftarrow \beta_1 m + (1-\beta_1)\nabla_\theta J_{minibatch}(\theta)$$
$$\theta = \theta - \alpha m$$

where β_1 is a hyperparameter between 0 and 1 (often set to 0.9). Briefly explain (you don't need to prove mathematically, just give an intuition) **how using m stops the updates from varying as much and why this low variance may be helpful to learning, overall.**

Answer:

首先m的更新是一种类似移动平均的计算方法，而移动平均是会使得曲线更加光滑（也就是variance更小）。然后更小的方差表现出梯度下降的方向变动幅度比较小，也就是整个loss可以更加稳定地沿着梯度方向下降，而且因为我们是在batch上训练，使用上面的方法也可以将其他batch的梯度信息利用上，而不是单单只用一个batch的梯度信息（一个batch还是容易产生bias）。

ii. Adam also uses *adaptive learning rates* by keeping track of **v**, a rolling average of the magnitudes of the gradients:

$$m \leftarrow \beta_1 m + (1-\beta_1)\nabla_\theta J_{minibatch}(\theta)$$
$$v \leftarrow \beta_2 v + (1-\beta_2)(\nabla_\theta J_{minibatch}(\theta) \odot \nabla_\theta J_{minibatch}(\theta))$$
$$\theta = \theta - \alpha \frac{m}{\sqrt{v}}$$

where $odot$ and / denote elementwise multiplication and division (so $z \odot z$ is elementwise squaring) and β_2 is a hyperparameter between 0 and 1 (often set to 0.99). **Since Adam divides the update by $\sqrt{v}$, which of the model parameters will get larger updates? Why might this help with learning?**

Answer:

因为Adam算法除以了$\sqrt{v}$，所以那些在过去更新中梯度较小的参数会得到更大的更新，这样能够起作用的原因应该是：过去总更新较多的参数应该已经趋于收敛，而那些更新较少的参数可能还处于没怎么优化的状态，所以以一个较大的更新值去更新这些参数能够更快地收敛。

(b) Dropout is a regularization technique. During training, dropout randomly sets units in the hidden layer **h** to zero with probability p_{drop} (dropping different units each minibatch), and then multiplies **h** by a constant γ. We can write this as

$$h_{drop} = \gamma d \circ h$$

Where $d \in \{0,1\}^{D_h}$ (D_h is the size of **h**) is a mask vector where each entry is 0 with probability p_{prop} and 1 with probability $(1-p_{prop})$. γ is chosen such that the expected value of h_{drop} is h:

$$E_{p_{drop}}[h_{drop}]_i = h_i, \ for\ all\ i \in \{1, \dots, D\}$$

i. What must γ equal in terms of p_{drop}? Briefly justify your answer.

Answer:

$$\begin{aligned} E_{p_{drop}}[h_{drop}]_i &= \gamma E_{p_{drop}}[d]_i h_i \\ &= \gamma(1 - p_{drop})h_i \\ &= h_i \\ \implies \gamma &= \frac{1}{1 - p_{drop}} \end{aligned}$$

ii. Why should we apply dropout during training but not during evaluation?

Answer:

在每一个batch中每一次不同的dropout等于是在训练一个子模型，当我们在做测试（评估）时，如果仍然只是用一个子模型进行预测，那么这个预测结果是十分受随机数影响的，相反，如果我们在测试时不使用dropout，我们就可以将在训练集中训练的大量子模型做一个平均输出结果，那样的结果是更加稳定且鲁棒的。

2. Neural Transition-Based Dependency Parsing

(a) (6 points) Go through the sequence of transitions needed for parsing the sentence "*I parsed this sentence correctly*". The dependency tree for the sentence is shown below. At each step, give the configuration of the stack and buffer, as well as what transition was applied this step and what new dependency was added (if any). The first three steps are provided below as an example.

Stack	Buffer	New Dependency	Transition
[ROOT]	[I, parsed, this, sentence, correctly]		Initial Configuration
[ROOT, I]	[parsed, this, sentence, correctly]		SHIFT
[ROOT, I, parsed]	[this, sentence, correctly]		SHIFT
[ROOT, parsed]	[this, sentence, correctly]	parsed -> I	LEFT-ARC
[ROOT, parsed, this]	[sentence, correctly]		SHIFT
[ROOT, parsed, this, sentence]	[correctly]		SHIFT
[ROOT, parsed, sentence]	[correctly]	sentence -> this	LEFT-ARC
[ROOT, parsed]	[correctly]	parsed - > sentence	RIGHT-ARC
[ROOT, parsed, correctly]			SHIFT
[ROOT, parsed]		parsed -> correctly	RIGHT-ARC
[ROOT]		ROOT -> parsed	RIGHT-ARC

(b) (2 points) A sentence containing n words will be parsed in how many steps (in terms of n)? Briefly explain why.

需要$2n$步来完成，因为每一个词都需要经过一个SHIFT和一个LEFT-ARC或RIGHT-ARC，所以需要$2n$

(f)

i.

- **Error type:** Verb Phrase Attachment Error
- **Incorrect dependency:** $wedding \rightarrow fearing$
- **Correct dependency:** $heading \rightarrow fearing$

ii.

- **Error type:** Coordination Attachment Error
- **Incorrect dependency:** $makes \rightarrow rescue$
- **Correct dependency:** $rush \rightarrow rescue$

iii.

- **Error type:** Prepositional Phrase Attachment Error
- **Incorrect dependency:** $named \rightarrow Midland$
- **Correct dependency:** $guy \rightarrow Midland$

iv.

- **Error type:** Modifier Attachment Error
- **Incorrect dependency:** $elements \rightarrow most$
- **Correct dependency:** $crucial \rightarrow most$