CS 224n Assignment #3

Yoshihiro Kumazawa

June 3, 2020

1. Machine Learning & Neural Networks

(a) i. When β_1 is large, $\boldsymbol{m}$ relies more on the history of the past gradients rather than the new one. For example, if $\beta_1 = 0.9$, the contribution of the new gradient to weight update is only 10% of that without momentum.

ii. The model parameters with smaller gradients will get larger updates. When the variance of the gradients is high, Adam might help the parameters go in the direction of small gradients, which the vanilla update strategy would ignore due to rounding errors.

(b) i. γ must equal $1/(1 - p_{\text{drop}})$ to make $\mathbb{E}_{p_{\text{drop}}}[\boldsymbol{h}_{\text{drop}}]_i = h_i$.

ii. Dropout during evaluation time would make the network stochastic, which is usually not desirable.

2. Neural Transition-Based Dependency Parsing

(a) See Table 1.

Table 1: Dependency parsing steps

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) A sentence containing n words will be parsed in $2n$ steps since every word is processed in 1 `SHIFT` operation and 1 `LEFT-ARC` or `RIGHT-ARC` operation.

(c) See `parser_transitions.py`.

(d)

(e)

(f)