Worth: 15% **Due:** By 6:45pm on Oct. 10rth

Name: Yuzheng Wu
Student number: N14898213
Netid: yw3375
GitHub repo: https://github.com/awuyz0214/DS1011

Bag of N-Gram Document Classification

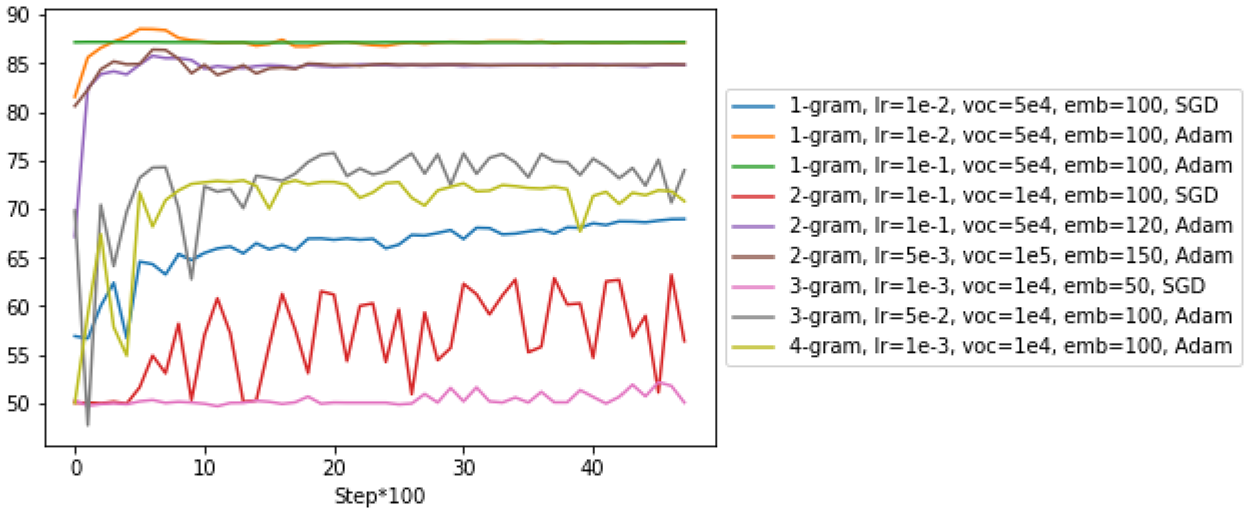
Parameters considered
n-gram = $1,2,3,4$
vocabulary size = $1e4, 5e4, 1e5$
embedding size = $50,100,150$
optimizer = SGD, Adam
learning rate = $0.05, 0.01, 0.005$, linear annealing of learning rate
max sentence length = 200, 500

Results

n-gram	vocabulary	embedding	optimizer	learning rate	max sentnce len.	validation accuracy
1	50,000	100	Adam	0.01	200	68.9
1	50,000	100	SGD	0.01	-	87.1
1	50,000	100	SGD	0.1	-	87.1
2	50,000	100	Adam	0.01	-	84.9
2	100,000	150	Adam	0.05	-	84.8
2	10,000	100	Adam	0.1	-	82.7
3	10,000	50	SGD	0.001	-	50.6
3	50,000	120	SGD	0.01	-	52.0
3	10,000	120	Adam	Decay from 0.5	-	73.9
4	10,000	100	Adam	0.01	-	70.1
4	50,000	100	SGD	0.01	500	70.2

Comments
1.
Adam learns the model much faster than SGD does. When I tried to learn the model for 1-gram, 50,000 vocabulary, 100 embedding and a fixed learning rate of 0.01, Adam performed much better than SGD, where Adam has $85+\%$ and SGD only has 68%.
However, Adam has a longer running time than SGD. It takes around twice time for Adam to finish compared to SGD.

2.
As the number of n-grams goes up, the running time becomes significantly longer. Also, increasing the vocabulary size reduces the running speed. These are obvious since a larger language pool should have longer time to run.



(a) Results

3.

Finally, 2-gram with 5e4 vocabulary and 100 embedding size implemented by Adam with learning rate=0.01 worked the best. It turned out short-grams work pretty well!