---Command Line Excerpt---

2019-04-30 09:17:08,507 - DEBUG - Epoch=1, iteration=100

2019-04-30 09:18:18,489 - DEBUG - Epoch=1, iteration=200

2019-04-30 09:18:26,180 - INFO - Perplexity on training data after epoch 1: 629.22

2019-04-30 09:19:37,282 - DEBUG - Epoch=2, iteration=100

2019-04-30 09:20:47,405 - DEBUG - Epoch=2, iteration=200

2019-04-30 09:20:55,111 - INFO - Perplexity on training data after epoch 2: 312.15

2019-04-30 09:22:06,239 - DEBUG - Epoch=3, iteration=100

2019-04-30 09:23:16,258 - DEBUG - Epoch=3, iteration=200

2019-04-30 09:23:23,992 - INFO - Perplexity on training data after epoch 3: 145.42

2019-04-30 09:24:35,269 - DEBUG - Epoch=4, iteration=100

2019-04-30 09:25:45,504 - DEBUG - Epoch=4, iteration=200

2019-04-30 09:25:53,216 - INFO - Perplexity on training data after epoch 4: 97.93

2019-04-30 09:27:04,130 - DEBUG - Epoch=5, iteration=100

2019-04-30 09:28:14,340 - DEBUG - Epoch=5, iteration=200

2019-04-30 09:28:22,070 - INFO - Perplexity on training data after epoch 5: 77.06

2019-04-30 09:29:33,063 - DEBUG - Epoch=6, iteration=100

2019-04-30 09:30:43,158 - DEBUG - Epoch=6, iteration=200

2019-04-30 09:30:50,880 - INFO - Perplexity on training data after epoch 6: 65.55

2019-04-30 09:32:02,025 - DEBUG - Epoch=7, iteration=100

2019-04-30 09:33:12,239 - DEBUG - Epoch=7, iteration=200

2019-04-30 09:33:19,962 - INFO - Perplexity on training data after epoch 7: 58.01

2019-04-30 09:34:30,646 - DEBUG - Epoch=8, iteration=100

2019-04-30 09:35:40,688 - DEBUG - Epoch=8, iteration=200

2019-04-30 09:35:48,394 - INFO - Perplexity on training data after epoch 8: 52.52

2019-04-30 09:36:59,376 - DEBUG - Epoch=9, iteration=100

2019-04-30 09:38:09,605 - DEBUG - Epoch=9, iteration=200

2019-04-30 09:38:17,271 - INFO - Perplexity on training data after epoch 9: 48.22

2019-04-30 09:39:28,240 - DEBUG - Epoch=10, iteration=100

2019-04-30 09:40:38,285 - DEBUG - Epoch=10, iteration=2002019-04-30 09:40:45,982 - INFO - Perplexity on training data after epoch 10: 44.70

Perplexity: 57.78 of the dev set

* This perplexity astonished me. After reading up about perplexities, I thought my text would have a lower perplexity since it is about reviewing clothing.
* I chose this text because I thought it was an interesting start to use Kaggle. Also since the texts are mostly about female clothing, I thought it would be interesting to calculate the complexity behind it. As follow up one could grab the male clothing reviews, and compare length, and complexity.
* For the next time, I think I should do better preprocessing. However though, this episode has shown me that many people who comment on clothing have bad grammar/spelling. This is certainly also visible in the samples.
* The best result is when Epochs are halved and Vocab size is doubled.
* Step for first commit : Epoch halved and Vocab size doubled
* When Initially looked at the epochs I thought they would take long to do. I thought it would be a nice idea to change the max vocabulary to see if it would actually make a difference, since people describe their sentiments about clothing. When I looked at the data, I thought initially that the vocabulary would be very focussed on clothing, but as it turns out, women especially mention stories about how they find this dress to be wonderful on the wedding … and other occasions. Also emotions, as we know, are hard to described so most women commenting had something else to say. However “Love” , “hate” etc were prominent, the other describing words were also very much there so to say.

---Command Line Excerpt---

2019-04-30 11:08:08,090 - DEBUG - Epoch=1, iteration=1002019-04-30 11:09:56,666 - DEBUG - Epoch=1, iteration=2002019-04-30 11:10:08,606 - INFO - Perplexity on training data after epoch 1: 813.882019-04-30 11:11:58,177 - DEBUG - Epoch=2, iteration=1002019-04-30 11:13:46,778 - DEBUG - Epoch=2, iteration=2002019-04-30 11:13:58,713 - INFO - Perplexity on training data after epoch 2: 547.652019-04-30 11:15:48,167 - DEBUG - Epoch=3, iteration=1002019-04-30 11:17:36,377 - DEBUG - Epoch=3, iteration=2002019-04-30 11:17:48,277 - INFO - Perplexity on training data after epoch 3: 402.652019-04-30 11:19:37,437 - DEBUG - Epoch=4, iteration=1002019-04-30 11:21:25,958 - DEBUG - Epoch=4, iteration=2002019-04-30 11:21:37,953 - INFO - Perplexity on training data after epoch 4: 217.602019-04-30 11:23:27,728 - DEBUG - Epoch=5, iteration=1002019-04-30 11:25:16,510 - DEBUG - Epoch=5, iteration=2002019-04-30 11:25:28,469 - INFO - Perplexity on training data after epoch 5: 131.95

perplexity: 112.98 -> - 19.15

* Next run -> Epoch = 15 to make a difference, vocab size did not change.
* Change in perplexity : Perplexity higher. One should go with the first : Half of the epochs and double the vocabulary.
* 2019-04-30 12:42:42,947 - INFO - Perplexity on training data after epoch 15: 37.39
* Perplexity: 63.70
* The first fabricated text is clearly more readable.

Ending statement: One should go with the first : Half of the epochs and double the vocabulary. There was a reduced complexity in the end. Perplexity on training data after epoch 5: 131.95 dev perplexity: 112.98 -> difference of - 19.15

In the end, I think it is interesting to read back on the blog post. The author also mentioned that he got luck in the first time, without “knowing” the parameters. Also to be honest, the complexity is lower in the first run but as an absolute value, the second one is lower. The first run with the preset data has the absolute lowest so for me, the presets win.