B10615023 楊傑安

B10615023 楊傑安 hw4

使用參數
- alpha = 0.3
- beta = 0.3
- TOPICS = 8

開發環境
Lab 內的 Server：
- CPU: i7-8700
- RAM: 64G
- OS: ubuntu 18.04.5
- 語言: Python 3.6.9 w/numba.jit

使用工具
- tqdm
- numba
- scipy.sparse
- numpy

心得
這次的作業在讀寫檔也都可以沿用以寫過的 function，需要實作的部分也都幾乎只是數學算式，寫起來很簡單，但真的太~慢了。

得益於實驗室強大的 Server，即使不對矩陣做壓縮還是能直接硬 train 一發。但由於 Python 效能低落和常常需要 acess 虛擬記憶體，一個 epoch 大約需要花 1.5 ~ 2 個小時才能完成，這個速度不太能接受。

後來有好心人士在 Kaggle 的提示大家可以使用 Sparse Matrix 和 numba.jit 來節省空間和時間，不過使用上的限制還不少。Sparse Matrix 只有在二維陣列使用才有提高效能的意義；numba.jit 使用上的限制更多：第三方的 package 幾乎除了 numpy 以外都不支援，當然也不支援 scipy.sparse，導致我必須在兩者擇一。最後我選擇只把 scipy.sparse 用來儲存 checkpoint，Ram 裏面還是裝了好幾個超大的矩陣。

而且 numba.jit 似乎有 menmory leak，只要我在宣告成 numba.njit 的 function 中 reference 到 global numpy ndarray，無論如何我都無法 free 掉該 ndarray 佔用的 RAM。所以最後我的方法是：一次只 train 一個 epoch，存檔之後就先 exit，然後在重新讀檔再 train。用這個方法的話，一個 epoch 大約只需要 200 秒 上下，雖然 Topic 的數量頂不上去（Ram 佔用超過實體大小導致都在等 swap），但也算能夠勉強接受了。