Fall 2020 EC504 Project Final Report
Topic modeling: Twitter Keyword Search

Junyou Chi
Boston University
chi12@bu.edu

Hao Zuo
Boston University
hzuo@bu.edu

Introduction

Our task is to create a keyword search engine that accepts a text file containing a number of tweets as input. And the search engine will provide a simple user interface for querying the list of tweets against keywords.

Literature Review

- Inverted Index
  Inverted index is the most standard free-text search method in information retrieval. An inverted index is a data structure that maps a word, or atomic search item, to the set of documents, or set of indexed units, that contain that word's postings. An individual posting may be a binary indication of the presence of that word in a document or may contain additional information, such as its frequency in the document and a set for each occurrence. Since access to an inverted index is based on a single key, efficient access typically implies that the index is either sorted or organized as a hash table.

- Hashmap
  Hash tables are fundamental components of several network processing algorithms and applications, including route lookup, packet classification, per-flow state management and network monitoring. These applications, which typically occur in the data-path of high-speed routers, must process and forward packets with little or no buffer, making it important to maintain wire-speed throughout. A poorly designed hash table can critically affect the worst-case throughput of an application, since the number of memory accesses

required for each lookup can vary. Hence, high throughput applications require hash tables with more predictable worst-case lookup performance. This allows designers to achieve higher lookup performance for a given memory bandwidth, without requiring large amounts of buffering in front of the lookup engine.

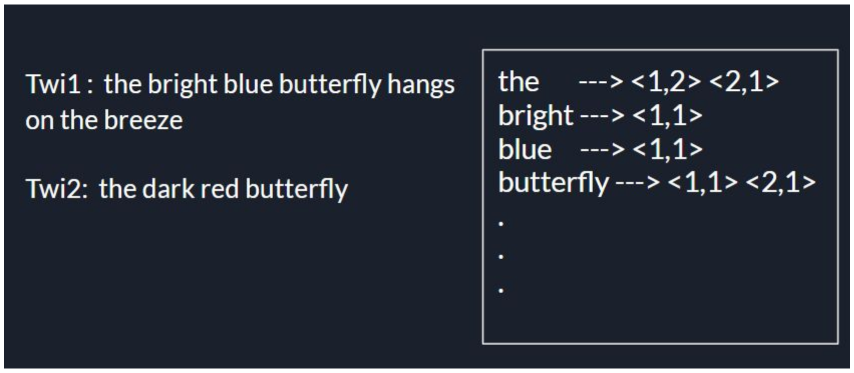
Literature Review

- Data Preprocessing

  Data preprocessing part is removing punctuation, capitalization and other abnormal characters. Also in data preprocessing, we split twitter to words.

- Data Storing
  - Inverted Index

    We use Inverted Index to store the twitter feeds. In this data structure only record the appearance of words without recording the sequence of the appearance. Because we need to count the number of words in twitter feed, we store pairs of int under key words. First int means the index number of twitter feed, while the second means the number of words. For example, there are <1,2> and <2,1> under 'the' so there is 2 'the' in twitter1 and 2 'the' in twitter2.


Twi1 : the bright blue butterfly hangs on the breeze
Twi2: the dark red butterfly
the ---> <1,2> <2,1>
bright ---> <1,1>
blue ---> <1,1>
butterfly ---> <1,1> <2,1>
.
.
.

  - Hashmap

    We use Hashmap to link the keyword and array of int pairs. Because when storing keyword data and searching for keywords, we need to load data for keywords, Hashmap is a fast way to locate the address of an array of int pairs which can finish the job in average O(1) time complexity.

    - Time Complexity

      When storing the total time complexity is O(m) when m is the total words over all twitter feeds.

- Keyword Searching
    - Finding keywords

      When searching for keywords, use keywords as key to hashmap to locate the array of int pairs. Use a new hashmap <int,int> to store candidates of twitter where the first int is the index of twitter feed and the second int is the total number of keywords in this twitter feed. After going through all keywords, we have a hashmap with all candidates twitter and their number of keywords.
    - Sorting

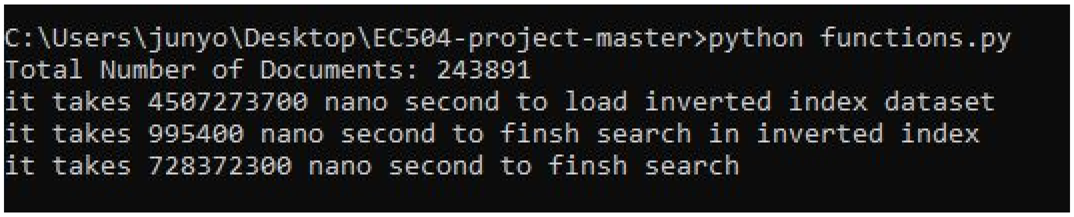
      We only need find top 10 related twitter feeds, so we can go through all candidates once and store top 10 in a sorted array. When checking a new candidate, we compare it through 9th to 1st to find the location of it in the array or if it has less keywords than 10th in the array, we do not record it.
    - Time complexity

      Time to find keywords and build a hashmap for candidates is O(n) where n is the total number of keywords appearing in all twitter feeds.
      Time to sort is O(m) where m is the number of candidates.

Result

- Comparison

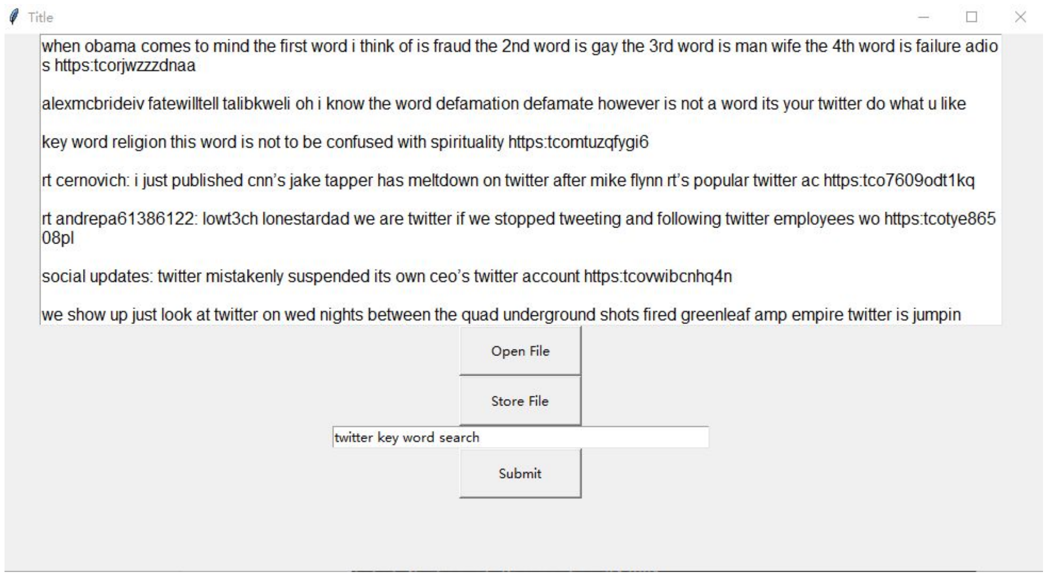
  We compare the inverted index searching with normal searching with O(nm) where n is the number of keywords and m is the number of total words in twitter feeds. We found that the time to load an inverted index database takes longer than a normal search but they are at the same time complex. However, searching in an inverted index database takes much shorter than a normal search. As a conclusion, if we only search once in a database, normal searching can be a little bit better than inverted index while if we are going to use a large database to search multiple times, inverted index can be much more efficient.



```
C:\Users\junyo\Desktop\EC504-project-master>python functions.py
Total Number of Documents: 243891
it takes 4507273700 nano second to load inverted index dataset
it takes 995400 nano second to finsh search in inverted index
it takes 728372300 nano second to finsh search
```

- User interface

  Our user interface can take a path to the input file and the keyword input and gives an output shown in the window with top10 twitter feeds. For now it only takes CSV files with the columns named 'content' as twitter input.



when obama comes to mind the first word i think of is fraud the 2nd word is gay the 3rd word is man wife the 4th word is failure adio s https:tcorjwzzzdnaa

alexmcbrideiv fatewilltell talibkweli oh i know the word defamation defamate however is not a word its your twitter do what u like

key word religion this word is not to be confused with spirituality https:tcomtuzqfygi6

rt cernovich: i just published cnn's jake tapper has meltdown on twitter after mike flynn rt's popular twitter ac https:tco7609odt1kq

rt andrepa61386122: lowt3ch lonestardad we are twitter if we stopped tweeting and following twitter employees wo https:tcotye865 08pl

social updates: twitter mistakenly suspended its own ceo's twitter account https:tcovwibcnhq4n

we show up just look at twitter on wed nights between the quad underground shots fired greenleaf amp empire twitter is jumpin

- CPP version

  We have an incomplete CPP version which can run the test case in the terminal. For now, the CPP version does a similar thing as the Python version. However, the CPP version needs a fixed format of input and the CPP version does not have a user interface. It can be tested with input twitter twi.txt and input keyword key.txt. In the terminal, after 'make' command, run '/project' and type '1' in the terminal to load data and type '2' in the terminal to search. Input of twitter and keywords can be changed by changing content in twi.txt and key.txt.

Work Breakdown

- Junyou Chi
  - Literature review
  - CPP version code
  - Python version code about dataset, searching, sorting.
- Hao Zuo
  - Literature review
  - Python version code about user interface, data preprocessing, file reading and loading.
  - Github uploading

Reference

- Cutting D., Pederson J., Optimizations for Dynamic Inverted Index Maintenance, 1989

  Retrieved from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.41.4893&rep=rep1&type=pdf

  Accessed 13 December, 2020

- Song H., Turner J., Dharmapurikar S., Lockwood J., Fast Hash Table Lookup Using Extended Bloom Filter: An Aid to Network Processing, 2005

  Retrieved from: https://www.arl.wustl.edu/~jst/pubs/2005/sigcomm05.pdf

  Accessed 13 December, 2020