1. tinyGoole

TinyGoogle will support 2 operations, searching and indexing. We think people who are qualified to insert index data into database are different from people who would search all the time in the database. So we separate index-client from search-client. You can either start an index-client to index new files or start a search-client to search for keywords. When you start a server, you have to specify several parameters, such as IP address, port number, location where you want to store index file, number of helpers that will connect to server initially. Then you need to start corresponding number of helper to connect to server in order for server to go ahead. If you start a index-client, you need to type in server IP address and port number for connection, then simply type in the path of file that you want to index. We design it to support concurrency of searching and indexing. The underlying mechanism is that when an index request comes in, the index master will first make a copy of each sub index file. Then each index helper will update those new copies of index files first. And the master will inform the server that index file names have been updated once all index helpers have done their jobs. The server therefore will change its local information of the paths of the index files. And those search requests that come in when the system is indexing will be performed on the original index files which is at most one version older than the latest version of index files. For the experimentation, we choose to measure response time, we will change number of keywords, size of input file, number of helpers.

Helper size response time

1. 10 567

1 100 655

1 200 670

1 500 740

1 1000 850

1 10000 2050

2 10 662

2 100 754

2 200 752

2 500 785

2 1000 875

2 10000 1675

6 10 2212

6 100 2334

6 200 2192

6 500 1989

6 1000 2504

6 10000 3276

Conclusion, since we need to synchronizing each index helper when they are trying to update main index file, we actually do not benefit from more index helpers. Additionally, index time increases as files sizes increase, but since the synchronizing time is almost the same, the increasing rate of index time is slower than the increasing rate of files sizes.

Search: