Submitted By:
Sanidhya Singal
B.Tech. CSE 3rd Yr.
2015085

CSE 508 Information Retrieval Assignment 1

Analysis on how the variation in the number of skips affects the system's performance:

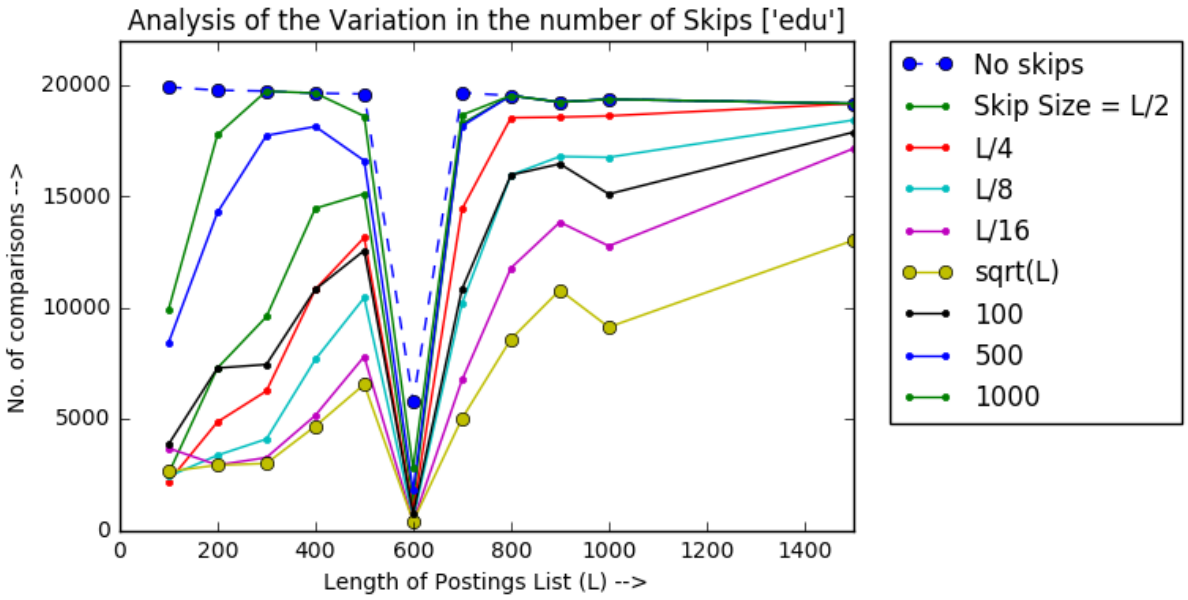
We start by building a new dictionary – 'inv_freq_dic'. The keys are the unique lengths of the postings lists and the values are the words for which the postings list has that length. e.g.: {'608': 'faster', '1994': 'price'}.

We apply some statistics on this. We find the length of the longest postings list and the mean and the median length of the entire postings lists. We find that these correspond to 'edu', 'price' and 'programm' respectively. We use these strings to compute the number of comparisons with postings lists of several different lengths.

We loop through the keys in the inv_freq_dic dictionary. For every key, we have a word of the length stored as the key. E.g. For 608, we have 'faster'. We find the postings list corresponding to 'faster' using word_dic dictionary. We use this postings list to compute AND with 'edu'. We use 9 different values for the skips:

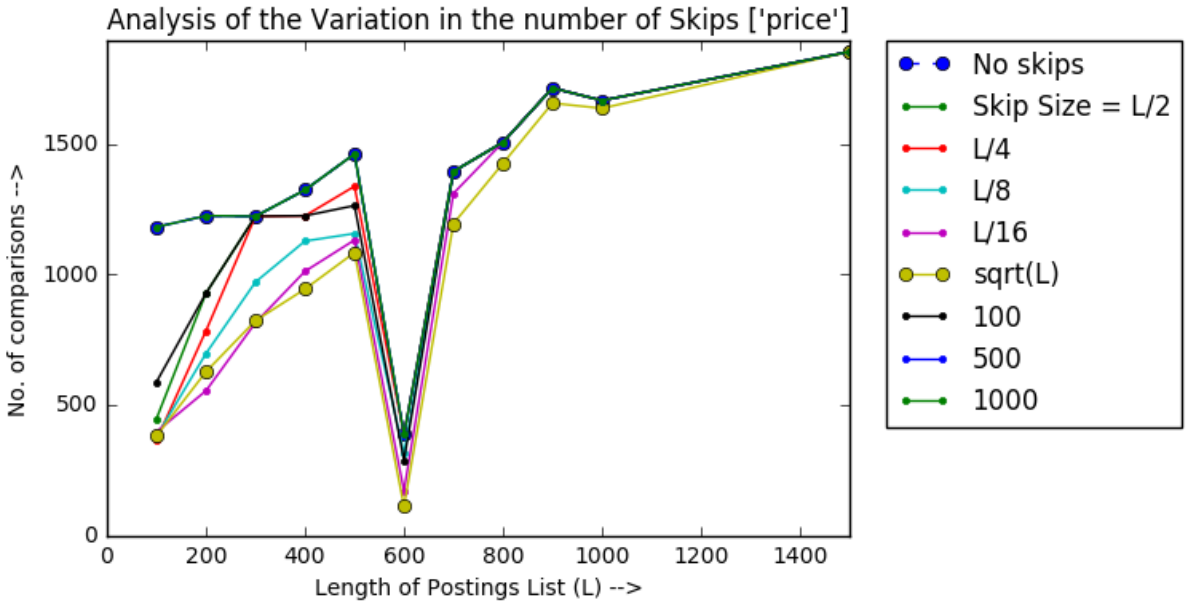
1. No skips
2. Skip size = L/2
3. Skip size = L/4
4. Skip size = L/8
5. Skip size = L/16
6. Skip size = $\sqrt{L}$
7. Skip size = 100
8. Skip size = 500
9. Skip size = 1000

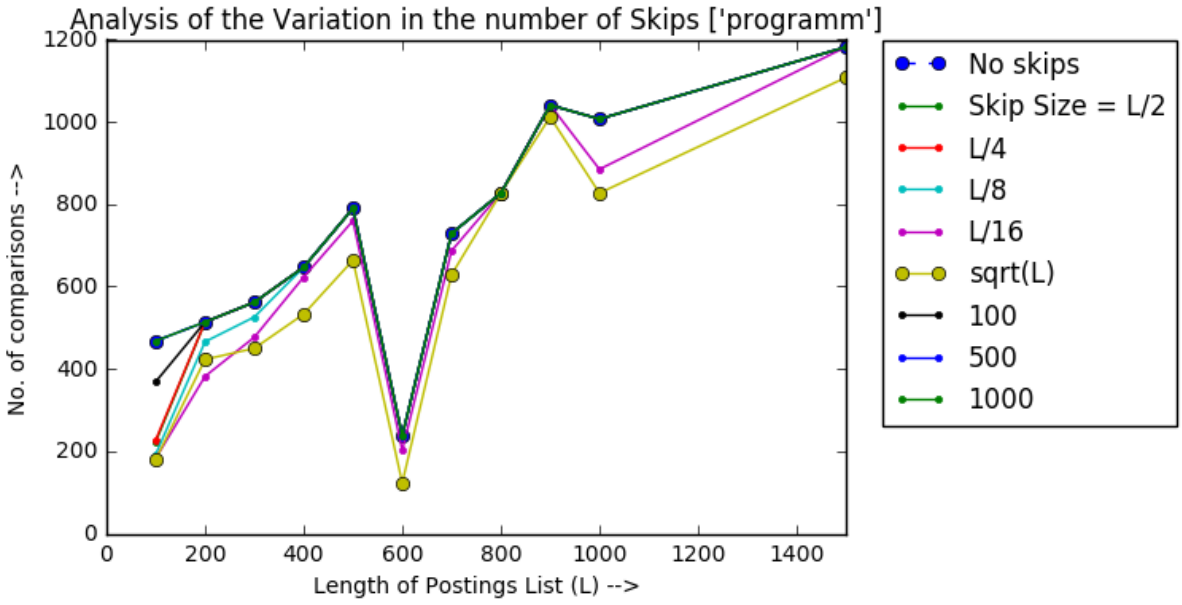
The following graph shows the results:



Submitted By:
Sanidhya Singal
B.Tech. CSE 3rd Yr.
2015085

Similarly, we repeat the process for 'price' and 'programm', our mean and median respectively. Following are the graphs obtained:





Observations and Conclusions:

1. In all the graphs, we observe that we do a lesser no. of comparisons when we use skip pointers than when we do not use skip pointers.
2. On an average, $\sqrt{L}$ provides the least no. of comparisons. So, this can be used as a nice skip size for a postings list of size L.