Project 8: Word Trend

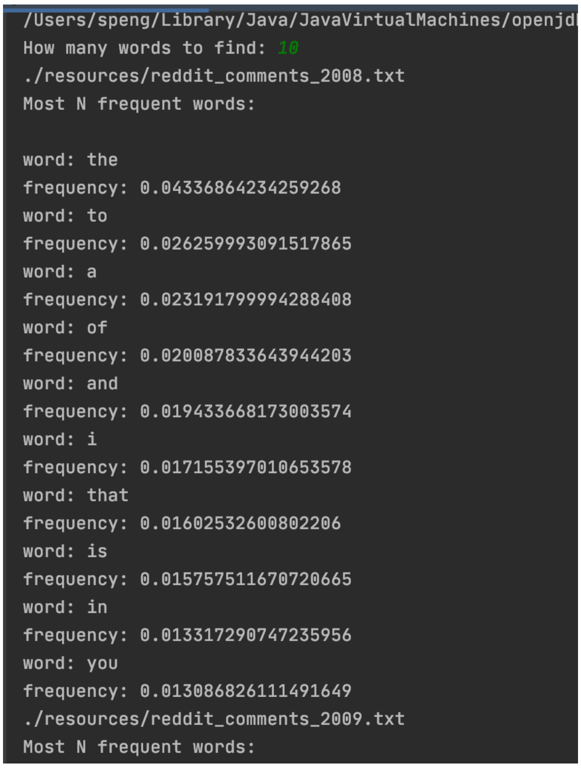
Abstract

In this week's project, I created a priority queue based on a heap data structure with a new comparator that determines the priority of KeyValuePair objects depending on its value field. A priority queue is a dynamic data structure that always returns the entry in the queue with the highest priority. The main purpose of the project is to analyze word trends in 8 years of Reddit posts. I made a CommonWordsFinder class, which uses the PQHeap to store my KeyValuePair objects read from the word count files. The main function shows the N (input by user) most common words in each text files. There's another class WordTrendsFinder that shows the frequency of several input words from 2008 to 2015. The result will show the 10 most common words, as well as a word trend graph in the theme of National Basketball Association.

Result

10 Most Common Words

The following screenshot shows the output of running the main function of CommonWordFinder.

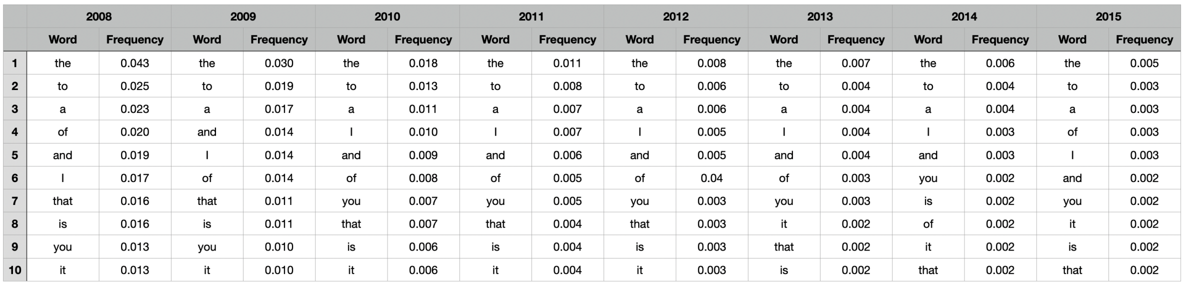


```
/Users/speng/Library/Java/JavaVirtualMachines/openjd
How many words to find: 10
./resources/reddit_comments_2008.txt
Most N frequent words:

word: the
frequency: 0.04336864234259268
word: to
frequency: 0.026259993091517865
word: a
frequency: 0.023191799994288408
word: of
frequency: 0.020087833643944203
word: and
frequency: 0.019433668173003574
word: i
frequency: 0.017155397010653578
word: that
frequency: 0.01602532600802206
word: is
frequency: 0.015757511670720665
word: in
frequency: 0.013317290747235956
word: you
frequency: 0.013086826111491649
./resources/reddit_comments_2009.txt
Most N frequent words:
```

The table below shows the most common words in each year.

10 Most Common Words from 2008 to 2015

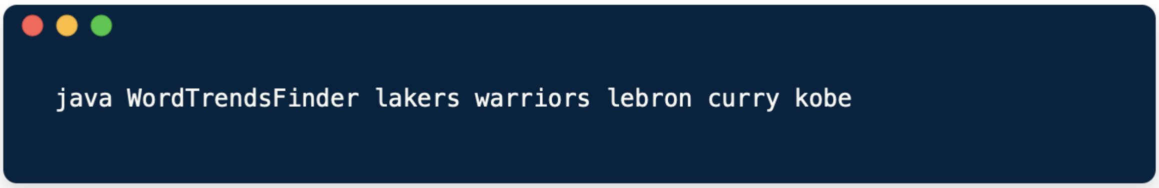


	2008		2009		2010		2011		2012		2013		2014		2015	
	Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency	Word	Frequency
1	the	0.043	the	0.030	the	0.018	the	0.011	the	0.008	the	0.007	the	0.006	the	0.005
2	to	0.025	to	0.019	to	0.013	to	0.008	to	0.006	to	0.004	to	0.004	to	0.003
3	a	0.023	a	0.017	a	0.011	a	0.007	a	0.006	a	0.004	a	0.004	a	0.003
4	of	0.020	and	0.014	I	0.010	I	0.007	I	0.005	I	0.004	I	0.003	of	0.003
5	and	0.019	I	0.014	and	0.009	and	0.006	and	0.005	and	0.004	and	0.003	I	0.003
6	I	0.017	of	0.014	of	0.008	of	0.005	of	0.04	of	0.003	you	0.002	and	0.002
7	that	0.016	that	0.011	you	0.007	you	0.005	you	0.003	you	0.003	is	0.002	you	0.002
8	is	0.016	is	0.011	that	0.007	that	0.004	that	0.003	it	0.002	of	0.002	it	0.002
9	you	0.013	you	0.010	is	0.006	is	0.004	is	0.003	that	0.002	it	0.002	is	0.002
10	it	0.013	it	0.010	it	0.006	it	0.004	it	0.003	is	0.002	that	0.002	that	0.002

We can see that each year each word's frequency has changed, but the most frequent 10 words from 2008 to 2015 remain the same, being *the*, *to*, *a*, *of*, *i*, *that*, *is*, *it*, and *you*. I have expected the above result because these words are the most used in our daily writing. For example, this paragraph has used already 5 *the*.

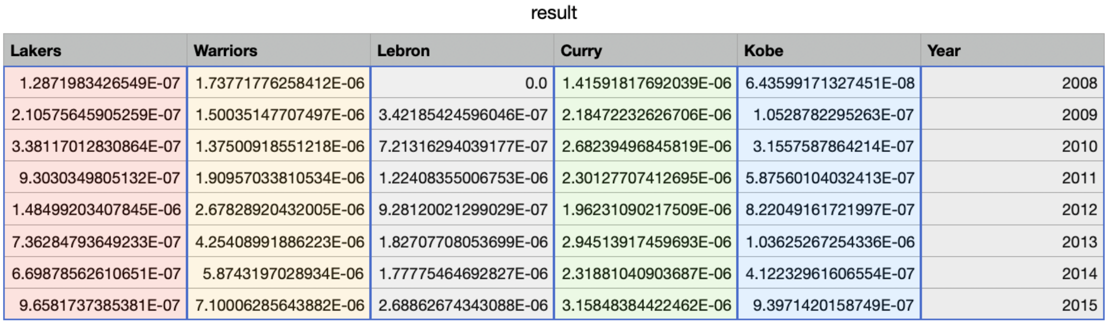
Word Trends

I also created the WordTrendsFinder to find the frequencies of chosen words from text files from 2008 to 2015. The command line arguments made it easier to input the chosen words from the terminal.



```
java WordTrendsFinder lakers warriors lebron curry kobe
```

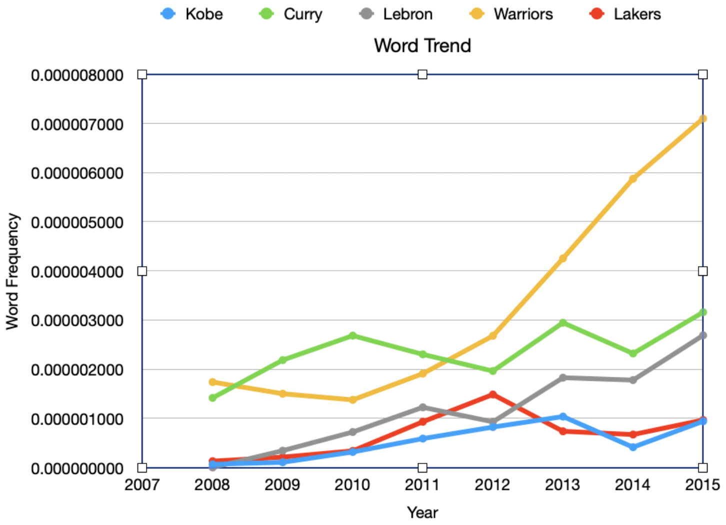
The words theme I chose is National Basketball Association. Lakers, Warriors are two NBA teams, while the other three are three NBA stars. I used a straight-forward method of creating 8 wordcounters which analyzes one year of text files. Though it is time-consuming, I managed to get the output stored in the csv file.



result

Lakers	Warriors	Lebron	Curry	Kobe	Year
1.2871983426549E-07	1.73771776258412E-06	0.0	1.41591817692039E-06	6.43599171327451E-08	2008
2.10575645905259E-07	1.50035147707497E-06	3.42185424596046E-07	2.18472232626706E-06	1.0528782295263E-07	2009
3.38117012830864E-07	1.37500918551218E-06	7.21316294039177E-07	2.68239496845819E-06	3.1557587864214E-07	2010
9.3030349805132E-07	1.90957033810534E-06	1.22408355006753E-06	2.30127707412695E-06	5.87560104032413E-07	2011
1.48499203407845E-06	2.67828920432005E-06	9.28120021299029E-07	1.96231090217509E-06	8.22049161721997E-07	2012
7.36284793649233E-07	4.25408991886223E-06	1.82707708053699E-06	2.94513917459693E-06	1.03625267254336E-06	2013
6.69878562610651E-07	5.8743197028934E-06	1.77775464692827E-06	2.31881040903687E-06	4.12232961606554E-07	2014
9.6581737385381E-07	7.10006285643882E-06	2.68862674343088E-06	3.15848384422462E-06	9.3971420158749E-07	2015

I made the following word trend graph according to the data above.



The above graph shows how the frequencies of these five words changed from 2008 to 2015.

The result was, to me, somewhat expected. I expected to be a spike in the trend of Warriors because in 2012, which does, shown from the sudden increase of the yellow line from 2010 to 2015. This happened because team Warriors started to have excellent season performances from 2010. The green trend line, representing the frequency of word Curry, which corresponds to the NBA Star Stephen Curry from the Golden State Warriors, also shows an increase from 2008 to 2010 and from 2012 to 2015. This was because Curry started his career early in 2007, and quickly grabbed everyone's attention that year onwards. From 2012 onwards, they started to win more games, and Curry's performance improves all the way. In fact, if we had data from 2016 onwards, Curry's frequency will even increase more. The grey trend line, represents the word lebron which corresponds to LeBron James, another NBA star. It has a steady but slow increase. In fact, in years from 2010 to 2014, he should have more numbers of mention in Reddit because that was the period where he entered the NBA finals four year straight. The reason that the frequency is not as high as others might be that, people also used the word "James" to refer to him, not only "LeBron". The other two lines, corresponding to NBA Player Kobe Bryant and his team Los Angelas Lakers, show a lower trend that the other three because from 2008 onwards, their performance became worse and worse.

Reflection

1. I learnt more about Heap data structure.