Assignment Report

System Description

I have managed to implement the assignment to the full extent, so it works with all three weight measures – binary, term frequency and tf.idf. I have implemented the algorithm all in one method as there was no scope for reusability. It first selects only relevant documents – those that have at least one term that is in a query. Then loops tough each term; initiating an inner loop which goes through each document attached to that term. If document is relevant it continues to the actual vector space model calculations. First, adding to the document size and then computing the similarity. This is where the weight measure is selected using if statements. All results are initially stored in two dictionaries where they add up. At the end of the loop, final result is computed into a separate dictionary. It then sorts all the results and returns only top 10.

Result Discussion

All the results are displayed on the next page in Figure 1 – Table of Results, and Figure 2 – A graph of F-Measure relative to all 12 configurations, and in Figure 3 – Time taken to complete the retrieval under different configurations. First of all, by looking at a global trend in Figure 2 we can see that the binary weighting has the lowest average performance, term frequency slightly higher and tf.idf has a more significant increase in performance. As well as that, it's clear that for each individual configuration performance always increases as we go from binary to term frequency, to tf.idf. Binary also has the lowest scoring performance at 0.05 and tf.idf – highest at 0.25. That shows that an improved weighting schema significantly impacts the performance of the IR system and plays an important role in the vector space model. In terms of time for different weights, we can see from Figure 3 that the trend is not as clear. There is a slight increase in time for individual configurations from binary to term frequency and more of an increase to tf.idf. That makes sense as calculating idf is more computationally intensive but the change in time is still not significant enough to pay big attention to it.

Now, moving onto a local trend in performance and time within each term weighting method, for different index configurations. (Note: I will compare the performance for different configurations in terms of increase from the basic one – where all terms are included). It is clear from Figure 2 that when all terms are considered without any adjustments the performance is at its lowest. As such the lowest F-measure of 0.05 is for all items when using binary term weighting. Stemming does improve the performance in all cases, but more so under term frequency and tf.idf which makes sense considering having all terms normalised contributes significantly to the count of their occurrence within the document. For binary the increase in performance is only by 0.1 and at least by 0.4 for other measures. On the other hand, using a stop list causes a massive improvement under binary and term frequency – around 0.9, and only 0.3 under tf.idf. When using both stemming and a stop list there is a same increase in performance as for stop list under binary. This again, shows that stemming doesn't play an important role when using binary term weights. For term frequency and tf.idf when using both the score is at its highest, increasing by around 0.1. Most importantly the best performance is at 0.25 when using stemming and stop list under tf.idf. In terms of time, it clearly divides, where all items and just stemming have more than double the time of just stop list of when using both. This brings to conclusions that stop list saves a lot of time as it reduces the number of terms that the algorithm has to iterate through. The other differences in time are not significant and not worth mentioning. It is important to point out that the lowest time of 0.52s is when using only stop listing under term frequency and it takes longest to compute results when using all terms under tf.idf – 1.75s.

Term Weighting	Index Type	Time	Precision	Recall	F-measure
Binary	All terms	1.5	0.05	0.04	0.05
	Stemming	1.46	0.06	0.05	0.06
	Stop list	0.51	0.16	0.13	0.14
	Both	0.63	0.16	0.13	0.14
Term Frequency	All terms	1.53	0.07	0.06	0.06
	Stemming	1.66	0.11	0.09	0.1
	Stop list	0.52	0.16	0.13	0.15
	Both	0.65	0.19	0.15	0.17
Tf.idf	All terms	1.75	0.2	0.16	0.17
	Stemming	1.66	0.25	0.2	0.22
	Stop list	0.62	0.23	0.18	0.2
	Both	0.68	0.28	0.22	0.25

Figure 1 – Table of results. **(Both refers to Stemming + Stop list)**

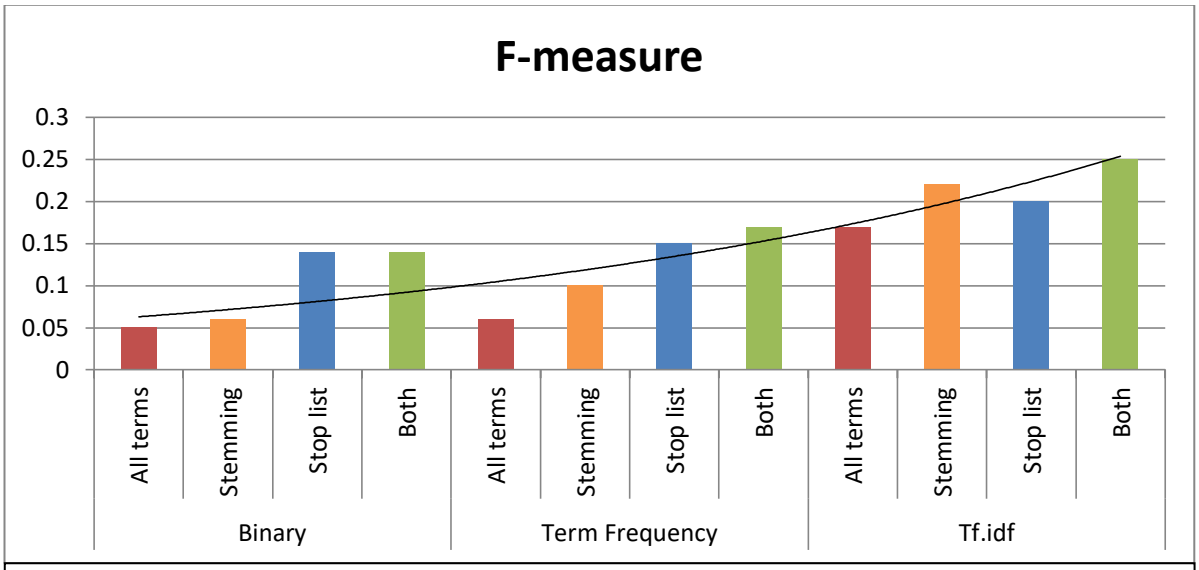

Figure 2 – Performance for different configurations in terms of F-measure.

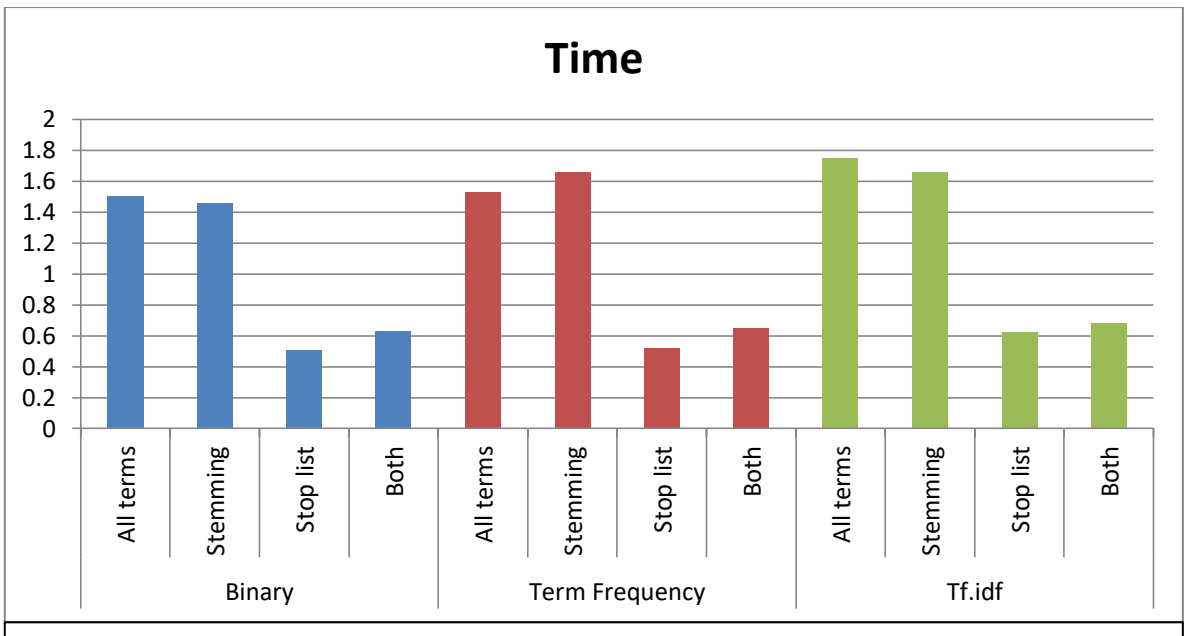

Figure 3 – Time taken for different configurations in seconds.