- (0% but required) If it is not blatantly obvious, please indicate where in your source code the evaluation occurs.

  All measures are implemented as functions in my source code and are easy to find. Also, from line 281 to 315 in my code, I write the output to the output file and from there my evaluation code can be found.

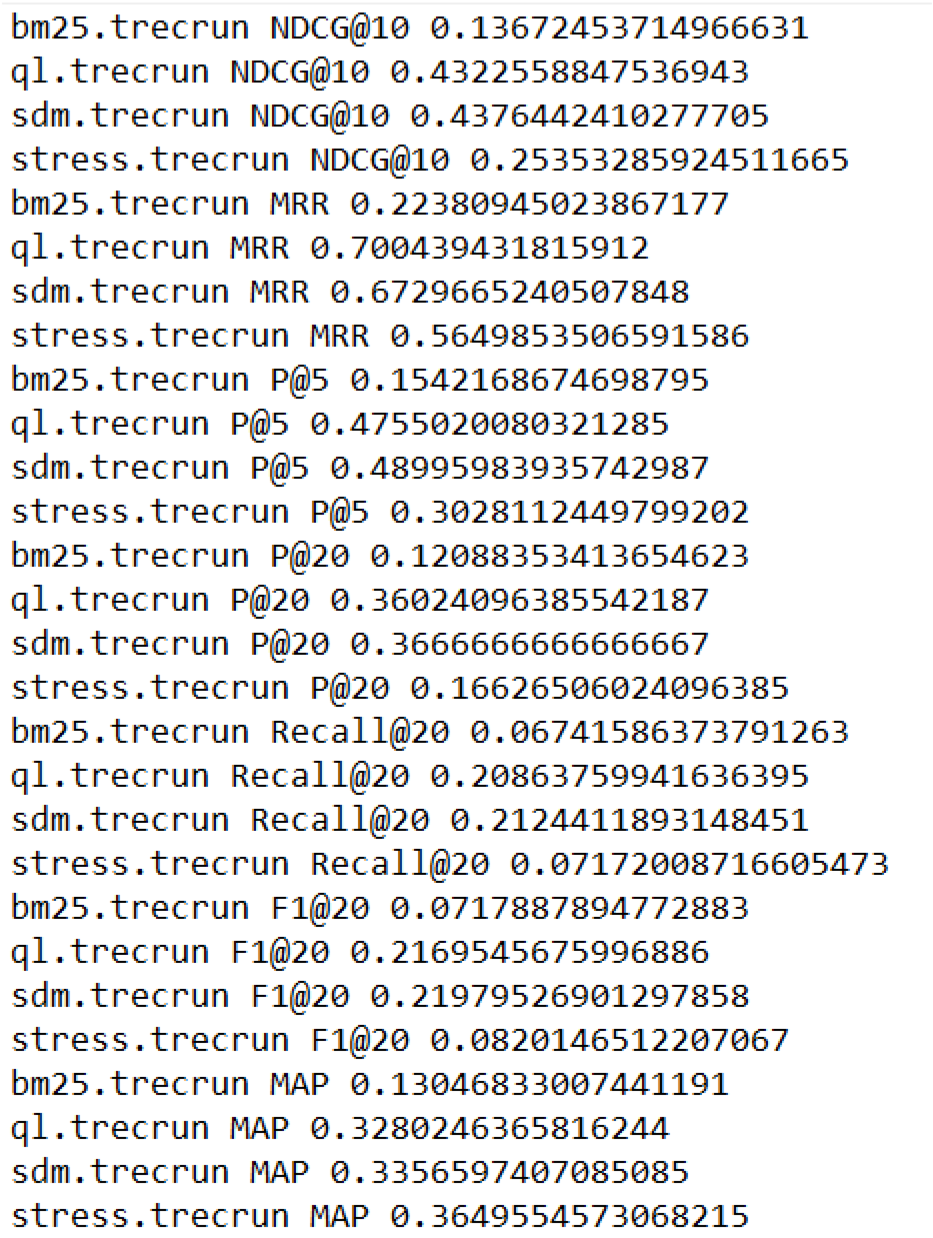
- (10%) Description of the system, design tradeoffs, questions you had and how you resolved them, etc.

  I use dictionaries often in this project because it is easy to make data connected and easy to understand their relationships. The file “stress.trecrun” is not sorted by its ranking, therefore it took me some time to figure out how to have it sorted.

- (5%) List the software libraries you used, and for what purpose.
  - The implementation must be your own work.

    I used “re” to split strings and replace strings, and used “math” to do calculations like log.

- (5%) Include a table of your results on each of the runs in your report.



```
bm25.trecrun NDCG@10 0.13672453714966631
ql.trecrun NDCG@10 0.4322558847536943
sdm.trecrun NDCG@10 0.4376442410277705
stress.trecrun NDCG@10 0.25353285924511665
bm25.trecrun MRR 0.22380945023867177
ql.trecrun MRR 0.700439431815912
sdm.trecrun MRR 0.6729665240507848
stress.trecrun MRR 0.5649853506591586
bm25.trecrun P@5 0.1542168674698795
ql.trecrun P@5 0.4755020080321285
sdm.trecrun P@5 0.48995983935742987
stress.trecrun P@5 0.3028112449799202
bm25.trecrun P@20 0.12088353413654623
ql.trecrun P@20 0.36024096385542187
sdm.trecrun P@20 0.3666666666666667
stress.trecrun P@20 0.16626506024096385
bm25.trecrun Recall@20 0.06741586373791263
ql.trecrun Recall@20 0.20863759941636395
sdm.trecrun Recall@20 0.2124411893148451
stress.trecrun Recall@20 0.07172008716605473
bm25.trecrun F1@20 0.0717887894772883
ql.trecrun F1@20 0.2169545675996886
sdm.trecrun F1@20 0.21979526901297858
stress.trecrun F1@20 0.0820146512207067
bm25.trecrun MAP 0.13046833007441191
ql.trecrun MAP 0.3280246365816244
sdm.trecrun MAP 0.3356597407085085
stress.trecrun MAP 0.3649554573068215
```

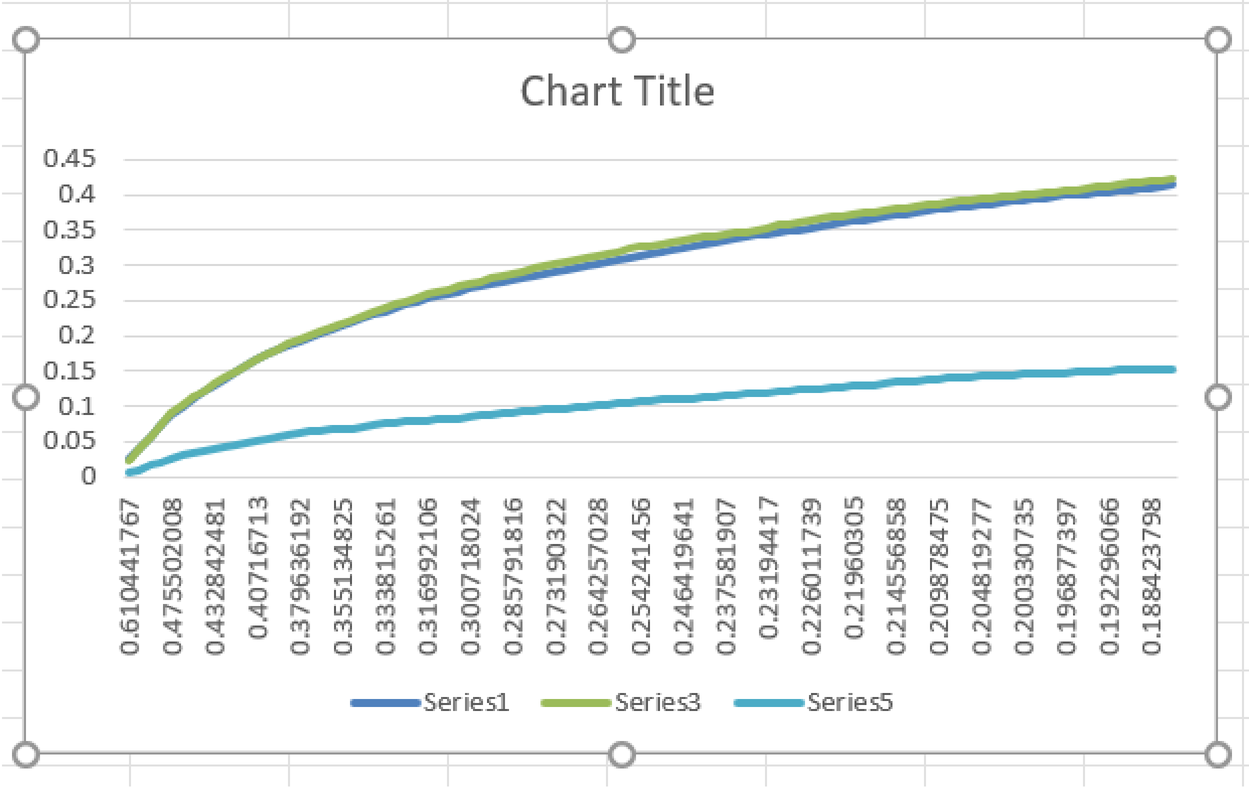
- (5%) Interpret the above table. Given what you know of the retrieval models involved, why might you see the results you are seeing?

  For NDCG@10, sdm and ql is higher than stress and bm25. For MRR, the value of ql is the highest and value of bm25 is the lowest. For P@5, ql > sdm > stress > bm25. For P@20, sdm > ql > stress > bm25. For Recall@20, sdm > ql > stress > bm25. For F1@20, sdm > ql > stress > bm25. For MAP, stress > sdm > ql > bm25. From these comparisons, we can see that bm25 always has the lowest value and ql and sdm are often in the top two. I think this is because bm25 is not very good and not accurate, however, ql and sdm are more accurate.

- (5%) How do you calculate the MAP of a query with no retrieved documents? Why is this a hard question? Argue for what you think the answer should be.

  I calculate the MAP of a query with no retrieved documents as 0. This is hard because the numerator and denominator of the value are both 0 and hard to determine the value. I think the answer should be 0 because there are no relevant retrieved documents to calculate MAP.

- (10%) Plot a precision / recall graph for query 450. Include data for QL, SDM and BM25 on the same graph.



Series1:ql
Series3:sdm
Series5:bm25