Omar Sanchez

(40%) The report discusses your work.

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

  BM25 – Lines 167 - 231

  QL – Lines 310 - 383

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

  A Series of nested for loops was used to calculate ranking values for each scene. The outermost for loop was the loop to iterate each scene. The second for loop was used to iterate each word in the query. Since the resulting rank for each scene was the sum of each result for each word in the query this was the method chosen. Many of the questions I had were on the logarithm portion of the equations. I used natural log to calculate the ranks and you cannot take the natural log of 0. To resolve this I simply created a if statement that would check to see if the logarithm portion of the equation resulted to 0. If it did I would simply make the rank of the word in the query result to 0.

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

  Simple JSON was used to parse the Shakespeare file. This was done to iterate the JSON array which contained all the information for each scene, also know as the documents.

- (5%) Do you expect the results for ***Q6*** to be good or bad? Why?

  I expect the results for Q6 to be bad. Since words in the query appear more than once, the summation of these query word results can appear smaller.

- (5%) Do you expect the results for a new query `setting the scene` to be good or bad? Why?

  I expect the results to be bad because the word “the” and “scene” is contained in the query. These are very commonly used words in the text of each document so when we get the rank of each document it might be misleading. The only word that might considerably affect the results would be setting which appears less.

- (15%) Look at the top ten results for ***Q3*** for both QL and BM25 (20 results in total).
  - For each result, record a number from 0 to 3, detailing whether you believe it to be:
    - 0: Not relevant
    - 1: Not really relevant
    - 2: Somewhat relevant
    - 3: Relevant. Note that something can be relevant even if it doesn't contain exactly those query words as a phrase. Ultimately you are the judge of relevance.

    Judgements.txt

  - (5%) Include this data in your submission. Each line should be a scene id and a number. Please ***only*** include a scene and a number.

    Judgements.txt

  - (10%) How does your system do on ***Q3***? Which method appears to be better, BM25 or QL? Justify your answer.

    Looking at the results I would say that my system provides a moderate amount of relevant results for Q3. BM25 seems to provide better results than QL. I concluded this by looking at the differences in values. The first few results for each document seems to be close to each other, but as we move down the ranks we can start to see the average difference increase. QL provides consistent results for only the first few rankings and can see this change quickly as we move past the second rank where there is a huge spike in difference.