- **Description of the system, design tradeoffs, questions you had and how you resolved them, etc. List the software libraries you used, and for what purpose.**

  This system builds on top of previous projects and uses the same design as far as the indexer is concerned. The indexer was modified to create a new document vectors file. The whole project is split into the indexer and retrieval packages, and the retrieval package houses the cluster package. One of the design choices I made was to read the entire document vectors file into memory to be able to access it easily instead of compressing or reading it part-by-part. I also decided to manually dispatch (use switch case) for scoring method rather than using abstract classes and subclasses. Another design choice I made was to store the modified document vectors (with tf-idf weights) in the cluster as opposed to the ones containing pure term counts. I used google's gson jar to easily read/write the document vectors to disk.

- **Explain what happens as the threshold value increases. Table the clustering results (number of clusters, size of each) for all of the threshold values.**

As the threshold value increases, so does the possibility of no existing cluster matching the threshold requirement, leading to an increase in creation of new clusters. This is what we observe as the threshold value increases, the number of clusters also increases

| Thresholds | 0.05 | 0.1 | 0.15 | 0.2 | 0.25 | 0.3 | 0.35 | 0.4 | 0.45 | 0.5 | 0.55 | 0.6 | 0.65 | 0.7 | 0.75 | 0.8 | 0.85 | 0.9 | 0.95 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| no. of clusters | 25 | 65 | 104 | 137 | 176 | 222 | 265 | 328 | 385 | 444 | 500 | 563 | 616 | 658 | 699 | 722 | 738 | 744 | 748 |
| cluster size bins | cluster sizes | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 8 | 21 | 37 | 53 | 86 | 126 | 184 | 237 | 301 | 375 | 452 | 535 | 596 | 660 | 701 | 729 | 740 | 748 |
| 2 | 0 | 4 | 7 | 13 | 27 | 38 | 42 | 55 | 63 | 69 | 67 | 68 | 49 | 43 | 32 | 17 | 8 | 4 | 0 |
| 3-5 | 3 | 13 | 32 | 40 | 55 | 58 | 62 | 60 | 65 | 61 | 48 | 39 | 30 | 19 | 7 | 4 | 1 | 0 | 0 |
| 6-10 | 2 | 15 | 19 | 27 | 24 | 28 | 27 | 26 | 20 | 13 | 10 | 4 | 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11-30 | 16 | 24 | 25 | 20 | 17 | 12 | 8 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 31-100 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 101-250 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 251-400 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 401-750 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- **Explain the difference between the four linking strategies. What behavior would you expect from each of them. Compare to the results you obtained using mean.**

  Single linking strategy: Candidate document's similarity is calculated against every document in the cluster, and the minimum value encountered is considered to be the representative value for that cluster in comparison to other clusters. Clusters will be more stringy and the distance between the farthest points will be large as the cluster grows in size as compared to mean linkage

  Complete linking strategy: Candidate document's similarity is calculated against every document in the cluster, and the maximum value encountered is considered to be the representative value for that cluster in comparison to other clusters. Clusters will be denser as compared to mean linkage

  Average linking strategy: Candidate document's similarity is calculated against every document in the cluster, and the average of all values encountered is considered to be the representative value for that cluster in comparison to other clusters. The results will be quite similar to those of mean linkage

  Mean linking strategy: Candidate document's similarity is calculated against the mean document vector