Analysis of Ranking of Dcouments provided by Lucene system and BM_25 system:

- Lucene uses Vector Space Information Retrieval model and Boolean Retrievalmodel.

- Lucene's conceptual scoring formula is

core(q,d) = coord-factor(q,d) · query-boost(q) · V(q) · V(d)/|V(q)|· doc-len-norm(d)· doc-boost(d)

- Lucene's practical scoring formula is

score(q,d) = coord(q,d)· queryNorm(q)· ∑(tf(t in d)· idf(t)2·t.getBoost()·norm(t,d))

^^^

Generic inferences got by analyzing the queries and documents produced by both Lucene 4.7.2 and BM_25 retrieval model:

- From the below query by query analysis its clear that both systems produce almost similar documents when top 5 documents are taken into account.

- But when ranking is considered, there is little difference in the ranking for these 5 documents produced by both the systems. This is because Lucene considers various normalizing factors which play a good role when huge corpus is considered.

- Lucene also considers a boosting factor which is why some documents are scored high by Lucene and not by BM_25.

  - Document level boosting

  - Document's field level boosting

  - Query level boosting

- Scoring is also very much dependent of the way documents are indexed.

^^^

Queries given to us for comparison:

Q1: hurricane isabel damage

Ranking by Lucene:

1.hurricane_isabel.txt 0.97974527

2.list_of_category_5_atlantic_hurricanes 0.25317204

3.eyewall.txt 0.19157135

4.eyewall_mesovortices 0.19157135

5.eye_(cyclone).txt 8.14510296975

Ranking by BM_25(Our implementation):

1.list_of_category_5_atlantic_hurricanes 11.2862183911

2.hurricane_isabel 10.5451890926

3.accumulated_cyclone_energy 8.41273674571

4.eye_(cyclone) 8.14510296975

5.eyewall 8.14510296975

- From the above two tables we can see that document ranking produced by both the systems are almost similar. 4 documents are common

in both the systems. Even though their rankings do not match exactly, it does not vary to a considerable extent.

- There is difference in the scoring since Lucene4.7.2 and BM_25(our implementation) use two different scoring schemes

- fi for each query term is greater in list_of_category_5_atlantic_hurricanes than hurricane_isabel.

- That is why List_of_category_5_atlantic_hurricanes comes first in according to our implementation. But Lucene considers much other normalization factors because of which the same file comes in rank 2 under Lucene.

^^

Q2: forcast models

Ranking by Lucene:

1. tropical_cyclone_prediction_model.txt 0.07050216

2. catastrophe_modeling.txt 0.0580369

3. orlan_space_suit.txt 0.054072693

4. orlan_suits.txt 0.054072693

5. space_environment.txt 0.048364084

Ranking by BM_25(our implementation):

1.tropical_cyclone_prediction_model 4.51772109827

2.tropical_cyclone_rainfall_forecasting 4.23255375471

3.orlan_space_suit 4.22426540654

4.orlan_suits 4.22426540654

5.thinkpad 4.21566374191

- From the above two tables, both the systems produce there documents in common and which are ranked almost similarly

- The term forecast does not appear in the document 'thinkpad' but the term models appear 36 times in 'thinkpad' hence its ranked

in the 5th position according our implementation.

- Since Lucene does different document length normalization when compared to BM_25 thinkpad does not occur in the top 5 of Lucene

^^

Q3: green energy canada:

Ranking by Lucene:

1.energy_source.txt 0.27908075

2.weather_radar.txt 0.14515151

3.taipei_101.txt 0.109090924

4.thunderstorm.txt 0.104241334

5.hurricane_sandy.txt 0.102737784

Ranking by BM_25:

1.weather_radar 8.10665939891

2.energy_source 6.70452314256

3.taipei_101 6.67988495167

4.laser 6.03329214798

5.lightsaber 5.95831231454

- The two systems produce three documents in common with similar rankings.

- fi is 25 for the doc weather_radar when fi for each term is added. Hence its ranked 1st according BM_25

^^

Q4:heavy rains

Ranking by Lucene:

1.list_of_wettest_tropical_cyclones_by_country.txt 0.27286336

2.wet_season.txt 0.24413592

3.kona_storm.txt 0.2103371

4.rainband.txt 0.19729161

5.rainbands.txt 0.19729161

Ranking by BM_25:

1.list_of_wettest_tropical_cyclones_by_country 9.04522537721

2.wet_season 8.08991130141

3.tropical_cyclone_rainfall_forecasting 7.69287359654

4.kona_storm 7.38407179963

5.rainbands 7.2775753871

- rainband does not come in top 5 in BM_25 implementation. Since the word rains appears only once in the document rainband.

- But Lucene ranks in top 5 because of other boosting factors.

^^^

Q5:hurricane music lyrics:

Ranking by Lucene:

1.audioboxer.txt 0.9201529

2.helios_(album).txt 0.34505734

3.hurricane_(disambiguation).txt 0.24497277

4.david_bowie.txt 0.20739545

5.hamilton_(musical).txt 0.18122782

Ranking by BM_25:

1.audioboxer 14.6185053392

2.helios_(album) 12.5761266596

3.david_bowie 10.6145008028

4.hamilton_(musical) 9.93502228226

5.ilse_delange 7.20630389724

- The query terms appear considerable number of times so BM_25 ranks in top 5. But Lucene considers corpus as a whole and

It considers other normalization factors because of which it does not come under top 5.

^^

Q6:accumulated snow

Ranking by Lucene:

1.storm.txt 0.22011475

2.precipitation.txt 0.17591068

3.precipitation_(meteorology).txt 0.17591068

4.list_of_wettest_tropical_cyclones_by_country.txt 0.14452274

5.flood.txt 0.14095986

Ranking by BM_25:

1.storm 7.58116406887

2.precipitation_(meteorology) 6.81636016544

3.precipitation 6.81636016544

4.list_of_wettest_tropical_cyclones_by_country 6.75136752541

5.accumulated_cyclone_energy 6.2117729838

- The above tables have 4 documents that are common in both. accumulated_cyclone_energy does not occur in the top 5 because of the difference in the way lucene normalizes the weights differently.

- Since BM_25 only considers ni and fi of majorly accumulated_cyclone_energy comes in top 5 in case of BM_25.

^^^

Q7:snow accumulation

Ranking by Lucene:

1.ku_band.txt 0.2526866

2.storm.txt 0.24358174

3.flood.txt 0.16845772

4.climate_oscillation.txt 0.14186487

5.hurricane_sandy.txt 0.11655435

Ranking by BM_25

1.ku_band 10.4568622114 BM_25

2.storm 9.26031654731 BM_25

3.flood 7.81078044193 BM_25

4.climate_oscillation 6.86651756274 BM_25

5.weather_radar 5.76461913037 BM_25

- The above two systems produce 4 documents in common for the query Q7.

- The 5th document varies because of the other factors like boosting factor which Lucene uses

^^^

Q8:massive blizzards blizzard

Ranking by Lucene:

1.storm_(novel).txt 0.05480083

2.storm.txt 0.052411508

3.cape_cod.txt 0.03930863

4.nor%27easter.txt 0.038750038

5.winter.txt 0.02777615

Ranking by BM_25:

1.storm_(novel) 7.04001706193

2.nor%27easter 6.60761853573

3.storm 5.18012238741

4.la_ni%c3%b1a 5.00709898295

5.winter 4.99028000303

- The two systems produce 4 documents in common which are having almost same ranks.

- But the document winter does not come in top 5 ranked by Lucene system.

^^^

Q9:new york city subway

Ranking by Lucene:

1.new_york_city.txt 0.6375816

2.hurricane_sandy.txt 0.3177774

3.new_england.txt 0.26265183

4.galveston_hurricane_of_1900.txt 0.14503837

5.dj_hurricane.txt 0.14184204

Ranking by BM_25:

1.new_york_city 18.0340946824

2.hurricane_sandy 16.0409806222

3.new_england 11.4633059365

4.galveston_hurricane_of_1900 9.40943198775

5.ibm 9.21385871329

- The two systems produce 4 documents in common with documents possessing same ranks in both the systems.

- The total fi of all the terms in the query for the document ibm is 34. Because of this large fi ,it comes in top 5 under BM_25. But because of various normalization the lucene uses, it does not come under top 5 when ranked by Lucene.