Emre Varol

CS533 Information Retrieval Systems Assignment #2

1. Q1: D1, D3, D4, D6 Total Retrieved Documents: 10, Total Relevant Documents: 4
   Q2: D1, D3, D6 Total Retrieved Documents: 10, Total Relevant Documents: 5

   a.

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Q1 Recall	0.25	0.25	0.50	0.75	0.75	1	1	1	1	1
Q2 Recall	0.20	0.20	0.40	0.40	0.40	0.60	0.60	0.60	0.60	0.60
Q1 Precision	1	0.50	0.66	0.75	0.60	0.66	0.57	0.50	0.44	0.40
Q2 Precision	1	0.5	0.66	0.50	0.40	0.50	0.43	0.37	0.33	0.30

Table 1: Recall and Precision Values

0	0.1	0.2	0.3	0,4	0.5	0.6	0.7	0.8	0.9	1
1	1	1	0.705	0.705	0.625	0.375	0.33	0.33	0.33	0.33

Table 2: Average Interpolated Precision Values for Standard Recall Values

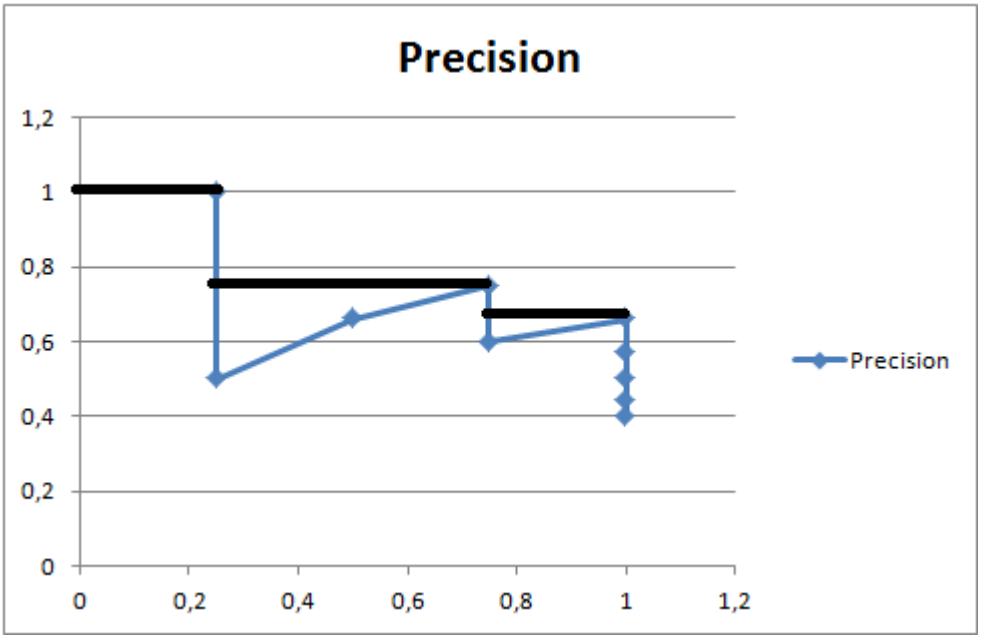

Figure 1: *Recall - Precision Graph for Query #1*

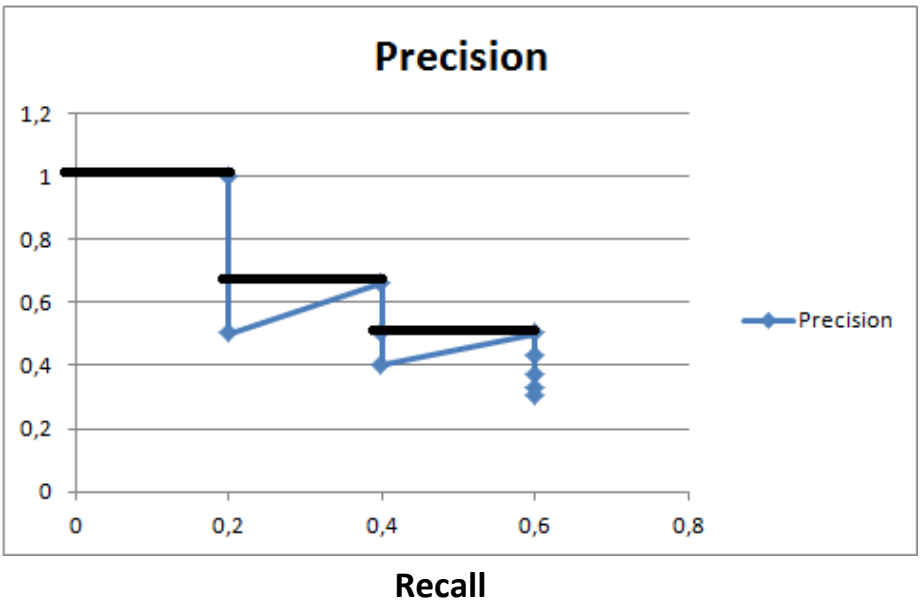

Figure 2: *Recall - Precision Graph for Query #2*

b. R – Precision for Query#1: 3/4
R – Precision for Query#2: 2/5
Average R – Precision: 0,575

c. MAP for Query#1: (1+ 0.66 + 0.75 + 0.66)/4 = 0.77
MAP for Query #2: (1+ 0.66 + 0.50 + 0 + 0)/5 = 0.43

2. # of Documents: 6

a. 5*4/2= 10 all possible similarity coefficients would be calculated.

b. T1 => <D1,1>, < D3,1>
T2 => <D2,1>, <D4,1>
T3 => <D1,1>, < D3,1>
T4 => <D2,1>, < D4,1>
T5 => <D2,1>, < D5,1>, < D6,1>
T6 => <D5,1>, <D6,1>
Documents that have at least one common term with D1: D3
Documents that have at least one common term with D2: D4, D5, D6
Documents that have at least one common term with D3: D1,
Documents that have at least one common term with D4: D2
Documents that have at least one common term with D5: D2, D6
Documents that have at least one common term with D6: D2, D5
10/2= 5

3.

	D1	D2	D3	D4	D5	D6
D1	1	0	1	0	0	0
D2	0	1	0	0.8	0.4	0.4
D3	1	0	1	0	0	0
D4	0	0.8	0	1	0	0
D5	0	0.4	0	0	1	1
D6	0	0.4	0	0	1	1

Table 3*: S Matrix*

4.

a.

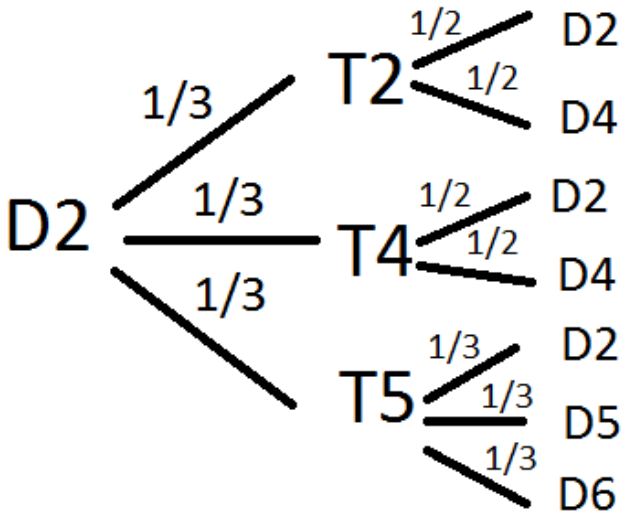

Figure 3: *Double Stage Probability Experiment Tree*

C_{24} = 1/3 * (1/2 + 1/2) = 1/3

b.

0.5	0	0.5	0	0	0
0	0.44	0	0.33	0.11	0.11
0.5	0	0.5	0	0	0
0	0.5	0	0.5	0	0
0	0.17	0	0	0.42	0.42
0	0.17	0	0	0.42	0.42

Table 4: *C Matrix obtained from D*

c. 3

d. Column sum technique is used to determine the seeds.

CS for document #1: 1
CS for document #2: 1.28
CS for document #3: 1
CS for document #4: 0.83
CS for document #5: 0.93
CS for document #6: 0.93

Seeds are 2, 1, 5 (1 and 3 are identical, 5 and 6 are identical. Thus, ones of them are used as a seed.)

e. T1 -><D1, 1> T2 -> <D2, 1>, <D4, 1>
T3 -> <D1, 1> T4 -> <D2, 1>, <D4, 1>
T5 -> <D2, 1>

f. Cluster #1: D2 Cluster #2: D1, D3 Cluster #3: D5, D6
To cluster a document, its cover coefficient values are compared with respect to the seed documents.

5.

a. C^2CM is the incremental version of C^3M algorithm, which means the algorithm can cluster the new coming documents and updates its clusters in the case of document deletion or addition. Actually its basic structure is very similar to the C^3M's one. In C^3M, determining the number of clusters is done by summing all c_{ii} values. When the document set is changed, number of clusters and seed documents are determined. If there is no change in the seeds, then, newcomers and in-ragbag documents are put into clusters. Otherwise, clusters that lose their seed or give another seed (or seeds) are falsified. Documents of falsified clusters are also put into ragbag cluster (they would be clustered in the next step.)

b. First of all, in the paper, the concept of clustering similarity is explained with poor English. I spent around 10 minutes to understand a sentence. Clustering similarity is used to see the performance of the maintenance algorithm. The maintenance algorithm obviously performs faster than reclustering. However, the quality of its clustering has to be validated. Clustering similarity concept is used at this point. If the clusters of the maintenance algorithm are near enough to the clusters of the reclustering algorithm, then, we can say that the maintenance algorithm is good.

c. C1: {{a, b, c}, {d, e}, {f, g}} C2: {{a}, {b, c, d}, {e, f, g}}
The underlying observation of Rand coefficient measure is that if a document pair is in the same cluster in C1, these documents better to be in the same cluster in C2. If not, they better not to be in the same cluster in C2. The rand index value for given clusters is 13/21 = 0.62.

d. Rand coefficient does not take into account the fact that these two partitions may be created by chance alone [1]. In this case, rand index value could be high enough to say that these partitions are similar. However, corrected rand coefficient could detect the random nature of the clusters and corrected rand index value reveals the randomness.

6.

a. Without skipping, 15 comparisons would be made.
Proposed skipping structure: Chunk size is 10. There exists a descriptor for each chunk that divides it from the previous one. This time, 6 comparisons would be made.
As stated in the paper, if the chunk size is as large as the block size, then, most of data blocks would be decoded because of having a match. If the chunk size is small, number of chunks that are skipped would increase but most disk blocks will still contain a valid chunk and because of this again most data blocks would be decoded.

b.

i. Ordered by $f_{d,t}$,
Term-a: <38, 5>, <43, 5>, <72, 5>, <75,5>, <12, 4><17, 4><33, 4><68, 4>, <10, 3>, <18, 3>, <55, 3>, <1, 2>, <9, 2>, <22, 2>, <24, 2>, <64, 2>, <88, 2>, <3, 1>
Term-b: <12, 2>, <66, 1>

ii. Term-a: <5 : 4: 38,43,72,75>, <4 : 4 : 12,17,33,68>, <3 : 3 : 10,18,55>, <2 : 6 : 1,9,22,24,64,88>, <1 : 1 : 3>

Term-b: <2 : 1 : 12>, <1 : 1, 66>

The second approach is compatible with compression. Also a threshold value could be useful to eliminate less-frequent documents. Additionally, the space can be saved by compressing the data and making document ordering for the documents that contain that term less frequently.

7.

a. **Feature Selection**: All the elements have some features. Some of these features are important enough to be considered in the concept of clustering. Thus, they must be determined.
   Feature Extraction: Some new features could be attained by looking the existing ones.
   Inter-pattern Similarity: Calculation of similarities among elements.
   Grouping: The name is self-explanatory: Putting the similar elements together.
   Feedback Loop: Improvement on the previous iteration results.
b. First of all, until the clustering of non-seed documents stage, the same algorithm (C3M) would be processed. Then, a document d would be partitioned into clusters, whose seeds have a similarity value (with d) greater than a threshold. Partition would be done proportional to the similarity values. It is like division of the parliament among parties in Turkey's democratic system.
c. Tabu Search is a heuristic optimization algorithm that finds valuable the statement "repetition of the same movement in a short amount of time would not make the solution better effectively". Thus, by listing previously done moves as tabu, it decreases the probability of having a cycle. By the advance of time, tabu moves would go out from the tabu list and they can again be available to improve the solution. In the concept of simulated annealing based clustering, definition of a move can be "moving a document from one cluster to another". Thus, when a document is moved from cluster *k* to cluster *l*, other documents cannot go from *k* to *l* until this movement is dropped from the tabu list.
d. As explained in the paper, the clustering methodology is:
   1. Define the feature representation style and assign feature values to the elements
   2. Define a similarity measure appropriate for the data domain
   3. Cluster and group the elements
   4. Data abstraction
   5. Assessment of the output
e. This part consists of my thoughts, thus, some sentences that seems to be exactly true may not actually be true.
   Generally, clustering algorithms assume that the input is worth to cluster; the input could be divided into categories. However, it may not be the case for any input document collection. Detection of the clustering tendency is done at the beginning of the clustering. One way is to use a naïve clustering algorithm, which performs in a short amount of time. Then, according to its result quality, clustering tendency could be calculated. Another way is to check whether a random distribution of the

documents into the clusters makes sense or not because if the documents have a tendency to be clustered, a random distribution could also show some signs. Another approach is to select a sub-set of the input documents and cluster them. If this clustering is successful, then it can be interpreted as the input could be clustered well too. Finally, success of the first primitive clustering can be measured with this trivial algorithm:
Step 1: Select 2 documents from each cluster
Step 2: Merge each pair of documents and split into 2 new documents randomly
Step 3: Check whether newly created documents would go into the same seed or not
Step 4: If they go to the same seed, ok, clustering is fine. If not, clustering is not successful.

8. No. Consider this example: Similarity values are;
   1. AB => 0.5 2.BC => 0.5 3.AC => 0.4. If the threshold value is 0.5 and the complete-link clustering algorithm performs in this order, clusters are {A, B} and {C}. However, if 2 is considered first (corresponding order is 2, 1, 3); then, clusters are {B, C} and {A}. This example shows that order of documents may have an impact on the results of the complete-link clustering algorithm.

9. Components of an information retrieval test collection are a set of documents, a set of queries and relevance information about each document with respect to each query.
   In order to measure the effectiveness on a large database system, the number of documents should be limited because there are too many of them. In this point, every sub-system gives the same amount of documents to a pool and this pool is considered to evaluate the effectiveness of it. At first glance, pooling may seem to be fair. However, it has some lacking points. First of all, still manual judgment is costly. Additionally, some systems may have many relevant documents and they can even contribute to the result after reaching to the pooling depth. However, some of these documents could come from the other systems. Thus, the greater measurement depth may or may not give better results. It decreases the reliability of pooling approach. Also different measurement depth and pooling depth may boost the effectiveness of biased systems. Finally, if some portions of the relevant documents are identified, the system that cannot contribute to the pool could be underestimated.
   Note: In this part, all the ideas are borrowed from Zobel's paper.
10. This 11-watt energy per query legend is stated at [2], although there is no proved citation behind this assumption, the author claims that the calculation is done by considering the server count of Google. In this article, not only the cost of power consumption but also its effect on climate has been discussed. I think what he claims is valid for all computer systems that are working on a network, but it is out of scope. Thus, I will just focus on the power consumption part. Brian [3] discussed that during 4^{th} quarter of 2007, Google reported capital expenses of $678 million with operating costs of $1.43 billion. According to ComScore, 17.6 billion core searches were conducted by Google during the same period. Using Google's formula and financial data along with the ComScore's estimates, it appears as though Google's average cost per core search query is nearly 12 cent. This is a rough calculation but still the answer is surprising. In Turkey, cost for 1 watt electricity is 0.15 TL [4].

11-watt energy costs 1.65 TL. However, as you can see from the above, cost of a query is nearly 0.19 TL. From this point of view, if Google achieves to obtain electricity 7 times cheaper than Turkey, which I believe they can do, then, this query cost legend would be real. As a result, from my opinion, it is a fair statement.

References

1. http://darwin.phyloviz.net/ComparingPartitions/index.php?link=Tutorial6
2. http://www.tagesschau.de/inland/energieverbrauch2.html
3. http://www.beussery.com/blog/index.php/2008/08/google-cost-per-query/
4. http://www.gazetemetropol.com/haberdetay.php?hit=10584