Assignment 5

Alan Ou
301320642

Question 1

		A		
		Yes	No	Total
B	Yes	320	480	800
	No	550	150	700
	Total	870	630	1500

First, we calculate P(A), which as stated in the notes, “is the proportion of the times that the judges [both] agree.” Therefore, we take where the numbers where both judge A and judge B say “yes” and the numbers where both judges say “no”, then divided by the total judgements.

P(A) = (320 + 150) / 1500 = **0.313333**

Then, we calculate P(nonrelevant), which is the total of when each judge says “no” (not relevant), then divided by twice the total number of judgements because each judge has their own 1500 judgements.

P(nonrelevant) = (630 + 700) / (1500 + 1500) = **0.443333**

Then, we calculate P(relevant), which is the total of when each judge says “yes” (relevant), then again divided by twice the total number of judgements.

P(relevant) = (800 + 870) / (1500 + 1500) = **0.556667**

Then, we calculate the probability that the two judges are expected to agree by chance.

P(E) = P(nonrelevant)^2 + P(relevant)^2 = 0.443333^2 + 0.556667^2 = 0.196544 + 0.309878 = **0.506422**

Finally, the kappa statistic.

k = (P(A) – P(E)) / (1 – P(E)) = (0.313333 - 0.506422) / (1 - 0.506422) = **- 0.391203**

As stated in the class notes, a Kappa value of 1 means that the judges always agree, a Kappa value of 0 means that the two judges agree at a rate of chance, while a negative Kappa value means that their rate of agreement is worse than random. In our case, we get **- 0.391203,** which is a negative value, therefore, no, the judges are not in agreement.

Question 2

The given ranking of answers to a query is:

4, 3, 1, 0, 3, 2, 1, 4, 0, 2

The discounted gains are:

4, **3**, 1/1.59 = **0.63**, **0**, 3/2.32 = **1.29**, 2/2.59 = **0.77**, 1/2.81 = **0.36**, 4/3 = **1.33**, **0**, 2/3.32 = **0.60**

The DCG values are:

4, 7, 7.63, 7.63, 8.92, 9.69, 10.05, 11.38, 11.38, 11.98

The perfect ranking is:

4, 4, 3, 3, 2, 2, 1, 1, 0, 0

Based on the perfect ranking, the discounted gains are:

4, **8**, 3/1.59 = **1.89**, 3/2 = **1.50**, 2/2.32 = **0.86**, 2/2.59 = **0.77**, 1/2.81 = **0.36**, 1/3 = **0.33**, **0**, **0**

The ideal DCG values are:

4, 12, 13.89, 15.39, 16.25, 17.02, 17.35, 17.67, 17.67, 17.67

The normalized DCG values are:

4/4 = **1**, 7/12 = **0.58**, 7.63/13.89 = **0.55**, 7.63/15.39 = **0.50**, 8.92/16.25 = **0.55**, 9.69/17.02 = **0.57**, 10.05/17.35 = **0.58**, 11.38/17.67 = **0.64**, 11.38/17.67 = **0.64**, 11.98/17.67 = **0.68**

The value of NDCG at 10 is **0.68**.

Question 3

By expressing both the query q and documents d1, …, dn as unit weight vectors, we can now use it to define the Euclidean distance.

Before we start, through normalization of the vectors q and d_i to unit weight vectors, we know that $||q|| = ||d_i|| = 1$

The Euclidean distance can be defined as $(q - d_i)$ or after normalizing

= $\frac{q-d_i}{|q-d_i|}$

= $\sqrt{(q-d_i)^2} = \sqrt{(q-d_i)(q-d_i)}$

= $\sqrt{q^2 - 2d_iq + d_i^2}$

Using the fact that normalizing the vectors q and d_i to unit weight vectors made them equal to 1

= $\sqrt{1 - 2d_iq + 1}$ = $\sqrt{2 - 2d_iq}$ (Equation 1)

Now onto the cosine similarity. We are given that the cosine similarity is defined as

sim(d_1, d_2) = **V**(d_1) x **V**(d_2)

Similarly, in terms of the query q and documents we can write

sim(q, d_i) = **V**(q) x **V**(d_i) = **V**(d_i) x **V**(q) = d_iq (Equation 2)

Now, focussing on equations 1 and 2, we can see that equation 2 is exactly what is under the square-root in equation 1.

Now suppose that we have two documents d_1 and d_2, and say that $q - d_1$ is lower than $q - d_2$, then using Euclidean distance (equation 1)

$$\sqrt{2 - 2d_1q} < \sqrt{2 - 2d_2q}$$

Similarly, using cosine similarity (equation 2)

$$d_1q < d_2q$$

Therefore, "the rank ordering produced by Euclidean distance is identical to that produced by cosine similarity."

Question 4

The relationship between F_1 and the break-even point is that at the break-even point, when precision and recall are identical, F_1 is also equal to precision which is equal to recall.

Say for example, precision = 0.5 and recall = 0.5. Plugging the numbers into the equation

F_1 = 2PR / (P + R) => 2*0.5*0.5 / (0.5 + 0.5) => 0.5 / 1 = 0.5

We can see that F1 = precision = recall at the break-even point.

Yes, there must always be a break-even point between precision and recall. Consider the following cases:

Case 1: If the first document retrieved is not relevant, then in this case, tp = 0, causing precision and recall to both equal 0, and therefore be the same.

Case 2: If the first document retrieved is relevant, then in this case, looking at the equations for precision and recall, as the number of documents retrieved increases, the number of false negatives will decrease, and the number of false positives will increase. As this continues, eventually, the number of false positives will surpass the number of false negatives, and therefore, there must be a break even point.

To summarize case 2, at the start, the number of false positives is less than the number of false negatives, but at the end, the number of false positives is more than the number of false negatives. As a result, there must be a point where the number of false positives and the number of false negatives were equal, and this would be the break-even point.

Question 5

The effect of this "boosting" is to improve the relevance, precision, and recall of the search results.

This is helpful when a user does not exactly know the keywords of their information need and can only describe it. For example, if the user didn't know the name of the whale character "Pearl" in Spongebob and instead types "Spongebob whale character", by adding the most frequent term in the top-10 documents can help return results that the user wanted.

On the other hand, this type of "boosting" would not be effective if for example, none or only a small portion of the top-10 query results were relevant to the query. In this case, adding the most frequent term in the snippets would not help make the existing query return any "better" results. In fact, automatically adding the most-frequent term from the snippets may cause the results to be even more irrelevant than before.

Question 6

$$\mathbf{Adjacency\ Matrix\ A}\ = \begin{matrix} \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{matrix}$$

$$\boldsymbol{Transition\ Probability\ Matrix} = \begin{matrix} \frac{\mathbf{3}}{\mathbf{80}} & \frac{\mathbf{37}}{\mathbf{80}} & \frac{\mathbf{37}}{\mathbf{80}} & \frac{\mathbf{3}}{\mathbf{80}} \\ \frac{\mathbf{37}}{\mathbf{80}} & \frac{\mathbf{3}}{\mathbf{80}} & \frac{\mathbf{37}}{\mathbf{80}} & \frac{\mathbf{3}}{\mathbf{80}} \\ \frac{\mathbf{3}}{\mathbf{80}} & \frac{\mathbf{3}}{\mathbf{80}} & \frac{\mathbf{3}}{\mathbf{80}} & \frac{\mathbf{71}}{\mathbf{80}} \\ \frac{\mathbf{1}}{\mathbf{4}} & \frac{\mathbf{1}}{\mathbf{4}} & \frac{\mathbf{1}}{\mathbf{4}} & \frac{\mathbf{1}}{\mathbf{4}} \end{matrix}$$

X_0	**1**	**0**	**0**	**0**
X_1	**1549/6400**	**393/6400**	**1651/6400**	**2807/6400**
X_2	**80281/512000**	**23917/102400**	**132947/512000**	**179187/512000**
X_3	**8648069/40960000**	**7311733/40960000**	**11377623/40960000**	**544903/1638400**
X_4	**603062697/3276800000**	**648498121/3276800000**	**897097043/3276800000**	**1128142139/3276800000**
X_5	**51057752477/262144000000**	**49512948061/262144000000**	**2862475367/10485760000**	**90011415287/262144000000**
X_6	**4000066293953/20971520000000**	**4052589644097/20971520000000**	**5736029878171/20971520000000**	**7182834183779/20971520000000**
X_7	**322810789023541 /1677721600000000**	**64204999023729 /335544320000000**	**458813043017942 /1677721600000000**	**575072772839871 /1677721600000000**
	.....	**....**	**....**	**...**

PageRank of the four pages is **Rank D > Rank C > Rank A = Rank B**

Hub and Authority

K = 1

Page	Hub	Authority
a	0.688247	0.37796
b	0.688247	0.37796
c	0.229416	0.75593
d	0	0.37796

Assignment 5

Alan Ou
301320642

K = 2

Page	**Hub**	**Authority**
a	0.688250	0.229413
b	0.688250	0.229413
c	0.229417	0.91766465
d	0	0.229413

The order of hub scores for each page would be **a = b > c > d**. In words, the hub score for pages a and b are equal, but both are greater than the hub score for page c, and all the hub scores are greater than the hub score for page d.

The order of authority scores for each page would be **c > a = b = d**. In words, the authority score for page c is greater than the authority scores for all other pages, and the authority scores for pages a, b, and d are equal.