Modifications made in Ranking Formulae

VAIBHAV KASTURIA

1 RELATIVENESS

A document should be considered more important than another document if query entities in the document are more at the beginning of the document compared to the other document. We consider the position of entities in the documents and modify the relativeness score of the documents (for `AND` and `OR` semantics) as follows:

$$score^{f}_{\wedge}(d, E_Q) = \frac{\sum_{e \in E_Q} (count(e,d) \cdot \sum\limits_{x=1}^{n} \exp(-ax))}{\sum_{e' \in ents(d)} (count(e',d) \cdot \sum\limits_{x=1}^{n} \exp(-ax))} \tag{1}$$

$$score^{f}_{\vee}(d, E_Q) = \frac{\sum_{e \in E_Q} (count(e,d) \cdot \sum\limits_{x=1}^{n} \exp(-ax))}{\sum_{e' \in ents(d)} (count(e',d) \cdot \sum\limits_{x=1}^{n} \exp(-ax))} \cdot \frac{|ents(d) \cap E_Q|}{|E_Q|} \tag{2}$$

In 1 and 2, a denotes the rate factor of the negative exponential function, x denotes position and n denotes the number of positions of the entity in the article. The user attention tends to decrease rapidly as he moves across the document. For example, a historian looking for an important document related to an entity may not read the complete document if he doesn't find the entity and the content he is looking for at the beginning. For this reason, we modelled the importance as negative exponential function rather than a linearly decreasing function. Sum of the negative exponential function for all positions of the query entities and related entities is performed keeping the rate factor values as 10^{-3}, 5×10^{-4} and 10^{-5}. The smaller we keep the decay factor, the slower will be the exponential decay. So, while setting the values of decay factor, we keep in mind the average length of the NYT articles. We observe that the negative exponential function converges to nearly zero at 5000 keeping rate factor as 10^{-3} and 7000 keeping rate factor as 5×10^{-4} and these numbers where the convergence nears zero nearly equal to the average number of words in a NYT article.

2 RELATEDNESS

When we define the relatedness score of a document, we should also keep in mind the difference in position between the query entities and the related entities. We define a measure called *proximity* score for each related entity which is based on the position difference of the related entity to the query entity.

Consider a document in which the red lines denote the positions of a query entity and blue lines denote the positions of a related entity along the length of the document as shown in Figure.

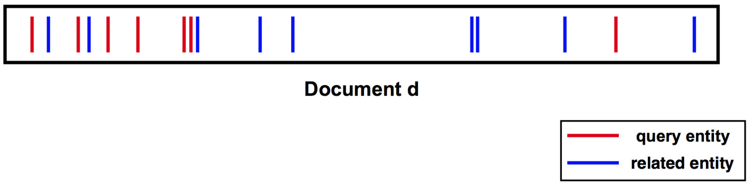

Figure 1: Positions of a query entity and a related entity inside a document d visualized across its length.

Define $Dist(e, e')$ and $avgDist(e, e')$ as the average distance of a related entity e w.r.t a query entity e' in a document d. The average distance is calculated by checking each red line(position of the query entity) in the document. If the red line that we are currently at, is preceded by a blue line and succeeded by a red line (or no line), or preceded by a another red line (or no line) and succeeded by a blue line, then we take $Dist(e, e')$ as the the absolute difference in position between current red line and the blue line. However, if the red line that we are currently at is preceded and succeeded by blue lines, then we take $Dist(e, e')$ as the average of the absolute difference in position between the current red line and the each blue line. Also, if we are currently at a red line which is preceded and succeeded by red lines as well, we do nothing and move to check the next red line. The $avgDist(e, e')$ obtained at the end is just an average of all the $Dist(e, e')$ in the document d.

We then define the *proximity* score of a related entity e w.r.t. a query entity e' in a document d as the inverse of the average distance between the related entity e and a query entity e', as in 3 :

$$proximityScore(e, e') = \frac{1}{avgDist(e, e')} \tag{3}$$

The *proximity* score of a related entity e for a document d for `AND` and `OR` Semantics is defined as follows:

$$proximityScore_{\wedge}(e, d) = \frac{\sum_{e' \in E_Q} proximityScore(e, e')}{|E_Q|} \tag{4}$$

$$proximityScore_{\vee}(e, d) = \sum_{e' \in E_Q \cap ents(d)} proximityScore(e, e') \cdot \frac{|ents(d) \cap E_Q|}{|E_Q|} \tag{5}$$

Consequently, 6 and 7 describe the *proximity* score of a related entity e for a time period t for `AND` and `OR` Semantics.

$$proximityScore_{\wedge}(e, t) = \frac{\sum_{d \in docs(t) \cap D_Q} proximityScore_{\wedge}(e, d)}{|docs(t) \cap D_Q|} \tag{6}$$

$$proximityScore_{\vee}(e, t) = \frac{\sum_{d \in docs(t) \cap D_Q} proximityScore_{\vee}(e, d)}{|docs(t) \cap D_Q|} \tag{7}$$

Finally, we incorporate 6 and 7 into our relatedness score for an entity $e \in E_D \setminus E_Q$ `AND` and `OR` Semantics as below.

$$\begin{aligned} score^r_\wedge(e) &= idf_\wedge(e) \cdot \sum_{t \in T_Q} (score^t_\wedge(t) \cdot proximityScore_\wedge(e,t) \cdot \frac{|docs(t) \cap D_Q \cap docs(e)|}{|docs(t) \cap D_Q|}) \\ &= idf_\wedge(e) \cdot \sum_{t \in T_Q} (proximityScore_\wedge(e,t) \cdot \frac{|docs(t) \cap D_Q \cap docs(e)|}{|D_Q|}) \end{aligned} \tag{8}$$

$$\begin{aligned} score^r_\vee(e) &= idf_\vee(e)\ N(E_Q,e) \sum_{t \in T_Q} (score^t_\vee(t) \cdot proximityScore_\vee(e,t) \cdot \frac{|docs(t) \cap D_Q \cap docs(e)|}{|docs(t) \cap D_Q|}) \\ &= idf_\vee(e)\ N(E_Q,e) \sum_{t \in T_Q} (N(E_Q,t) \cdot proximityScore_\vee(e,t) \cdot \frac{|docs(t) \cap D_Q \cap docs(e)|}{|D_Q|}) \end{aligned} \tag{9}$$

We now take the new relatedness scores for an entity $e \in E_D \setminus E_Q$ when calculating the relatedness score of a document d.

3 EVALUATION

Observe the Tables 1, 2, 3, 4 and 5 for the NDCG and Precision values. The description of the different rankings is as follows:

- [A] : Relativeness Score keeping rate factor a as 10^{-5}
- [B] : Timeliness Score
- [C] : Relatedness Score considering position of entities
- [A′] : Relativeness Score keeping rate factor a as 5×10^{-4}
- [A″] : Relativeness Score keeping rate factor a as 10^{-3}

Table 1: Average NDCG and Precision of the probabilistic models for all queries (Q1-Q24).

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.47	0.27	0.31	0.50	0.41	0.34	0.42
NDCG@10	0.51	0.36	0.40	0.53	0.45	0.40	0.44
NDCG@all	0.78	0.69	0.72	0.79	0.76	0.73	0.76
P@5	**0.48**	0.28	0.34	**0.51**	0.37	0.34	0.38
P@10	0.38	0.30	0.34	**0.38**	0.30	0.30	0.29

Measure	[A′]	[A′][B]	[A′][C]	[A′][B][C]	[A″]	[A″][B]	[A″][C]	[A″][B][C]
NDCG@5	**0.49**	0.51	0.41	0.41	0.48	0.50	0.43	0.43
NDCG@10	**0.54**	**0.56**	0.44	0.45	**0.55**	**0.56**	0.46	0.47
NDCG@all	**0.80**	0.80	0.76	0.76	0.79	0.80	0.76	0.76
P@5	**0.51**	**0.53**	0.38	0.39	**0.49**	**0.52**	0.39	0.39
P@10	**0.41**	**0.40**	0.29	0.30	**0.41**	**0.41**	0.31	0.31

We notice that there is a good improvement in the new relativeness score over the previous relativeness score for single entity queries and categorical

Table 2: Average NDCG and Precision of the probabilistic models for single-entity queries (Q1-Q6).

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	**0.73**	0.30	0.14	**0.73**	0.49	0.24	0.52
NDCG@10	**0.73**	0.38	0.32	**0.74**	0.52	0.30	0.52
NDCG@all	**0.88**	0.67	0.66	**0.88**	0.80	0.67	0.79
P@5	**0.67**	0.23	0.13	**0.63**	0.33	0.23	0.37
P@10	**0.43**	0.27	0.25	**0.43**	0.25	0.22	0.25

Measure	[A′]	[A′][B]	[A′][C]	[A′][B][C]	[A″]	[A″][B]	[A″][C]	[A″][B][C]
NDCG@5	0**.73**	**0.76**	0.49	0.52	**0.72**	**0.76**	0.52	0.53
NDCG@10	**0.77**	**0.78**	0.52	0.53	**0.77**	**0.78**	0.54	0.55
NDCG@all	**0.89**	**0.90**	0.80	0.80	**0.89**	**0.90**	0.80	0.80
P@5	**0.67**	**0.67**	0.33	0.37	**0.63**	**0.67**	0.37	0.37
P@10	**0.48**	**0.48**	0.25	0.27	**0.48**	**0.48**	0.27	0.28

Table 3: Average NDCG and Precision of the probabilistic models for multiple-entity AND queries (Q7-Q12).

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	**0.35**	0.28	0.19	0.39	0.22	0.19	0.22
NDCG@10	0.43	0.33	0.22	0.44	0.23	0.22	0.23
NDCG@all	0.76	0.72	0.67	0.77	0.68	0.67	0.68
P@5	**0.47**	0.33	0.17	**0.53**	0.23	0.17	0.23
P@10	0.48	0.32	0.20	0.47	0.23	0.20	0.23

Measure	[A′]	[A′][B]	[A′][C]	[A′][B][C]	[A″]	[A″][B]	[A″][C]	[A″][B][C]
NDCG@5	**0.37**	0.39	0.19	0.19	0.34	0.33	0.20	0.20
NDCG@10	**0.47**	0.46	0.23	0.23	**0.45**	0.45	0.23	0.23
NDCG@all	**0.77**	0.77	0.67	0.67	0.76	0.76	0.67	0.67
P@5	**0.50**	**0.53**	0.20	0.20	**0.47**	0.47	0.20	0.20
P@10	**0.52**	**0.50**	0.23	0.23	0.48	0.47	0.23	0.23

Table 4: Average NDCG and Precision of the probabilistic models for multiple-entity OR queries (Q13-Q18).

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.64	0.24	**0.45**	0.65	0.67	0.46	0.66
NDCG@10	0.65	0.36	**0.56**	0.66	0.68	0.54	0.66
NDCG@all	0.85	0.69	0.79	0.85	0.86	0.79	0.85
P@5	0.57	0.27	0.57	0.60	0.60	**0.47**	0.60
P@10	0.38	0.30	0.38	**0.42**	0.40	0.42	0.40

queries. There is also a nice improvement for combination of timeliness and relativeness score for single entity queries. Further, the improvement is higher for rate factors of 10^{-3} and 5×10^{-4} where convergence to zero is faster than for a rate factor of 10^{-5} where convergence to zero is comparatively slower.

Measure	[A′]	[A′][B]	[A′][C]	[A′][B][C]	[A″]	[A″][B]	[A″][C]	[A″][B][C]
NDCG@5	0.65	0.64	0.67	0.66	0.65	0.64	0.66	0.64
NDCG@10	0.64	0.62	0.65	0.63	0.65	0.63	0.65	0.64
NDCG@all	0.85	0.85	0.85	0.85	0.84	0.84	0.84	0.83
P@5	0.60	0.60	0.63	0.63	**0.63**	0.63	0.63	0.63
P@10	0.38	0.35	0.38	0.37	0.40	0.38	0.40	0.40

Table 5: Average NDCG and Precision of the probabilistic models for category queries (Q19-Q24).

Measure	[A]	[B]	[C]	[A][B]	[A][C]	[B][C]	[A][B][C]
NDCG@5	0.18	0.26	0.46	0.22	0.24	0.48	0.27
NDCG@10	0.25	0.36	0.48	0.27	0.35	0.54	0.35
NDCG@all	0.64	0.69	0.77	0.66	0.69	0.77	0.70
P@5	**0.20**	0.27	0.50	**0.27**	0.30	0.50	0.33
P@10	**0.22**	0.30	0.40	**0.22**	**0.30**	0.38	0.27

Measure	[A′]	[A′][B]	[A′][C]	[A′][B][C]	[A″]	[A″][B]	[A″][C]	[A″][B][C]
NDCG@5	**0.23**	0.27	0.27	0.29	**0.23**	0.27	0.33	0.34
NDCG@10	**0.31**	0.36	0.35	0.41	**0.34**	**0.37**	**0.41**	**0.45**
NDCG@all	**0.67**	0.68	0.70	0.70	**0.68**	0.69	**0.72**	0.73
P@5	**0.27**	**0.33**	**0.33**	0.37	**0.23**	**0.30**	**0.37**	**0.37**
P@10	**0.27**	**0.28**	**0.30**	0.33	**0.28**	**0.30**	**0.33**	**0.33**

The relatedness score, however, becomes worse considering position of entities than before. The reason to why the new method fails to give improvement could be attributed to the disambiguation error or multiple detection of a word by the entity linking system as both a query entity and a related entity. As an example, consider the query entity as `President_of_Colombia` and the following snippet inside a document: "*...the President of Colombia César Gaviria today declared ...*". Suppose that the entity system links the word *President of Columbia* to the entity `President_of_Colombia` and the word *President* to the entity `President_of_the_United_States` due to the high popularity of the word President associated with the US President. In such a case, the average distance between for the related entity `President_of_the_United_States` w.r.t the query entity `President_of_Colombia` for a document containing both these entities just once becomes zero since position is same for both these words and hence the proximity score for the related entity `President_of_the_United_States` w.r.t the query entity `President_of_Colombia` in the document becomes infinity. We tackled this problem by changing proximity score as zero wherever it becomes infinity. However, even after such a change, such a detection by the entity linking system causes our relatedness score to not improve.