Contents

Discounted cumulative gain (**DCG**) is a measure of effectiveness of a Web search engine algorithm or related applications, often used in information retrieval. Using a [graded relevance][] scale of documents in a search engine result set, DCG measures the usefulness, or *gain*, of a document based on its position in the result list. The gain is accumulated from the top of the result list to the bottom with the gain of each result discounted at lower ranks.1

[graded relevance]: Relevance (information retrieval) "wikilink"

0.1 Overview

Two assumptions are made in using DCG and its related measures.

1. Highly relevant documents are more useful when appearing earlier in a search engine result list (have higher ranks)
2. Highly relevant documents are more useful than marginally relevant documents, which are in turn more useful than irrelevant documents.

DCG originates from an earlier, more primitive, measure called Cumulative Gain.

0.1.1 Cumulative Gain

Cumulative Gain (CG) is the predecessor of DCG and does not include the position of a result in the consideration of the usefulness of a result set. In this way, it is the sum of the graded relevance values of all results in a search result list. The CG at a particular rank position p is defined as:

$$\mathrm{CG_p} = \sum_{i=1}^{p} rel_i$$

Where rel_i is the graded relevance of the result at position i.

The value computed with the CG function is unaffected by changes in the ordering of search results. That is, moving a highly relevant document d_i above a higher ranked, less relevant, document d_j does not change the computed value for CG. Based on the two assumptions made above about the usefulness of search results, DCG is used in place of CG for a more accurate measure.

0.1.2 Discounted Cumulative Gain

The premise of DCG is that highly relevant documents appearing lower in a search result list should be penalized as the graded relevance value is reduced logarithmically proportional to the position of the result. The discounted CG accumulated at a particular rank position p is defined as:

$$\mathrm{DCG_p} = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i)}$$

Previously there has not been shown any theoretically sound justification for using a [logarithmic][] reduction factor1 other than the fact that it produces a smooth reduction. An alternative formulation of DCG[2] places stronger emphasis on retrieving relevant documents:

$$\mathrm{DCG_p} = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2(i+1)}$$

0.1.3 Normalized DCG

Search result lists vary in length depending on the query. Comparing a search engine's performance from one query to the next cannot be consistently achieved using DCG alone, so the cumulative gain at each position for a chosen value of p should be normalized across queries. This is done by sorting documents of a result list by relevance, producing the maximum possible DCG till position p, also called Ideal DCG (IDCG) till that position. For a query, the *normalized discounted cumulative gain*, or nDCG, is computed as:

$$\mathrm{nDCG_p} = \frac{DCG_p}{IDCG_p}$$

The nDCG values for all queries can be averaged to obtain a measure of the average performance of a search engine's ranking algorithm. Note that in a perfect ranking algorithm, the DCG_p will be the same as the $IDCG_p$ producing an nDCG of 1.0. All nDCG calculations are then relative values on the interval 0.0 to 1.0 and so are cross-query comparable.

The main difficulty encountered in using nDCG is the unavailability of an ideal ordering of results when only partial relevance feedback is available.

0.2 Example

Presented with a list of documents in response to a search query, an experiment participant is asked to judge the relevance of each document to the query. Each document is to be judged on a scale of 0-3 with 0 meaning irrelevant, 3 meaning completely relevant, and 1 and 2 meaning "somewhere in between". For the documents ordered by the ranking algorithm as

$$D_1, D_2, D_3, D_4, D_5, D_6$$

the user provides the following relevance scores:

$$3, 2, 3, 0, 1, 2$$

That is: document 1 has a relevance of 3, document 2 has a relevance of 2, etc. The Cumulative Gain of this search result listing is:

$$\mathrm{CG_p} = \sum_{i=1}^{p} rel_i = 3 + 2 + 3 + 0 + 1 + 2 = 11$$

Changing the order of any two documents does not affect the CG measure. If D_3 and D_4 are switched, the CG remains the same, 11. DCG is used to emphasize highly relevant documents appearing early in the result list. Using the logarithmic scale for reduction, the DCG for each result in order is:

i	rel_i	$\log_2 i$	$\frac{rel_i}{\log_2 i}$
1	3	0	N/A
2	2	1	2
3	3	1.585	1.892
4	0	2.0	0
5	1	2.322	0.431
6	2	2.584	0.774

So the DCG_6 of this ranking is:

$$\mathrm{DCG_6} = rel_1 + \sum_{i=2}^{6} \frac{rel_i}{\log_2 i} = 3 + (2 + 1.892 + 0 + 0.431 + 0.774) = 8.10$$

Now a switch of D_3 and D_4 results in a reduced DCG because a less relevant document is placed higher in the ranking; that is, a more relevant document is discounted more by being placed in a lower rank.

The performance of this query to another is incomparable in this form since the other query may have more results, resulting in a larger overall DCG which may not necessarily be better. In order to compare, the DCG values must be normalized.

To normalize DCG values, an ideal ordering for the given query is needed. For this example, that ordering would be the [monotonically decreasing][] sort of the relevance judgments provided by the experiment participant, which is:

$$3, 3, 2, 2, 1, 0$$

The DCG of this ideal ordering, or *IDCG*, is then:

$$\text{IDCG}_6 = 8.69$$

And so the nDCG for this query is given as:

$$\text{nDCG}_6 = \frac{DCG_6}{IDCG_6} = \frac{8.10}{8.69} = 0.932$$

0.3 Limitations

1. Normalized DCG metric does not penalize for bad documents in the result. For example, if a query returns two results with scores $1, 1, 1$ and $1, 1, 1, 0$ respectively, both would be considered equally good even if later contains a bad result. One way to take into account this limitation is use $1 - 2^{rel_i}$ in numerator for scores for which we want to penalize and $2^{rel_i} - 1$ for all others. For example, for the ranking judgments $Excellent, Fair, Bad$ one might use numerical scores $1, 0, -1$ instead of $2, 1, 0$.

2. Normalized DCG does not penalize for missing documents in the result. For example, if a query returns two results with scores $1, 1, 1$ and $1, 1, 1, 1, 1$ respectively, both would be considered equally good. One way to take into account this limitation is to enforce fixed set size for the result set and use minimum scores for the missing documents. In previous example, we would use the scores scores $1, 1, 1, 0, 0$ and $1, 1, 1, 1, 1$ and quote nDCG as nDCG@5.

3. Normalized DCG may not be suitable to measure performance of queries that may typically often have several equally good results. This is especially true when this metric is limited to only first few results as it is done in practice. For example, for queries such as "restaurants" nDCG@1 would account for only first result and hence if one result set contains only 1 restaurant from the nearby area while the other contains 5, both would end up having same score even though later is more comprehensive.

0.4 References

[2]: Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. 2005. Learning to rank using gradient descent. In Proceedings of the 22nd international conference on Machine learning (ICML '05). ACM, New York, NY, USA, 89-96. DOI=10.1145/1102351.1102363

[3]: Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Wei Chen, Tie-Yan Liu. 2013. A Theoretical Analysis of NDCG Ranking Measures. In Proceedings of the 26th Annual Conference on Learning Theory (COLT 2013).