Normalized Discounted Cumulative Gain – Multilabel Ranking Metrics | ML

Last Updated : 15 Jul, 2021

Discounted Cumulative Gain

Discounted Cumulative Gain (DCG) is the metric of measuring ranking quality. It is mostly used in information retrieval problems such as measuring the effectiveness of the search engine algorithm by ranking the articles it displays according to their relevance in terms of the search keyword.

Let's consider that a search engine that outputs 5 documents named (D_1, D_2, D_3, D_4, D_5) are output in that order. We need to define the relevance scale (0-3) where:

- **0 :** not relevant
- **1-2 :** somewhat relevant
- **3 :** completely relevant

Suppose these documents have relevance scores:

- D_1 : 3
- D_2 : 2
- D_3 : 0
- D_4 : 0
- D_5 : 1

The ***Cumulative Gain*** is the sum of these relevance scores and can be calculated as:

$CG = \sum_{i=1}^{5} (rel)_i = 3 + 2 + 0 + 0 + 1 = 6$

The discounted cumulative gain can be calculated by the formula:

$$CG = \sum_{i=1}^{5} \frac{rel_i}{\log_2(i+1)}$$

$$DCG_5 = \frac{3}{log_2(2)} + \frac{2}{log_2(3)} + \frac{0}{log_2(4)} + \frac{0}{log_2(5)} + \frac{1}{log_2(6)}$$
$$DCG_5 = 3 + \frac{2}{1.585} + 0 + 0 + \frac{1}{2.585}$$
$$DCG_5 = 3 + 1.26 + 0.3868$$
$$DCG_5 \simeq 4.67$$

Now we need to arrange these articles in descending order by rankings and calculate DCG to get the Ideal Discounted Cumulative Gain (IDCG) ranking.

$$IDCG_5 = \frac{3}{log_2(2)} + \frac{2}{log_2(3)} + \frac{1}{log_2(4)} + \frac{0}{log_2(5)} + \frac{0}{log_2(6)}$$
$$IDCG_5 = 3 + \frac{2}{1.585} + \frac{1}{2} + 0 + 0$$
$$IDCG_5 = 3 + 1.26 + 0.5$$
$$IDCG_5 = 4.76$$

Now, we calculate our **Normalized DCG** using the following formula :

$$nDCG = \frac{DCG_5}{IDCG_5}$$
$$nDCG = \frac{4.67}{4.76}$$
$$nDCG \simeq 0.98$$

Code : Python program for *Normalized Discounted Cumulative Gain*

Python3

```
# import required package
from sklearn.metrics import ndcg_score, dcg_score
import numpy as np

# Relevance scores in Ideal order
true_relevance = np.asarray([[3, 2, 1, 0, 0]])

# Relevance scores in output order
relevance_score = np.asarray([[3, 2, 0, 0, 1]])

# DCG score
dcg = dcg_score(true_relevance, relevance_score)
print("DCG score : ", dcg)

# IDCG score
idcg = dcg_score(true_relevance, true_relevance)
print("IDCG score : ", idcg)

# Normalized DCG score
ndcg = dcg / idcg
print("nDCG score : ", ndcg)

# or we can use the scikit-learn ndcg_score package
```

Output:

```
DCG score :  4.670624189796882
IDCG score :  4.761859507142915
nDCG score :  0.980840401274087
nDCG score (from function) :  0.980840401274087
```

Limitations of Normalized Discounted Cumulative Gain (NDCG):

- NDCG metrics does not penalize the bad documents outputs. For Example: *[3]* and *[3, 0, 0]* has same NDCG but in second out there are two irrelevant documents.
- Because no specific standard defined for the number of output documents. DCG does not seem to deal with missing any relevant document in the output. For example, two outputs *[3, 3, 3]* and *[3, 3, 3, 3]* of similar input are considered equally good. For output-1 the DCG_3 is calculated but for output-2 the DCG_4 is calculated.

References:

- DCG Wikipedia article
- Jarvelin, K., & Kekalainen, J. (2002). Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems (TOIS), 20(4), 422-446.

Like 2

Next

Listing out directories and files in

RECOMMENDED ARTICLES

Page : 1 2 3

01 MultiLabel Ranking Metrics - Ranking Loss | ML
05, Mar 20

02 Multilabel Ranking Metrics-Label Ranking Average Precision | ML
05, Mar 20

03 ML | MultiLabel Ranking Metrics - Coverage Error
04, Mar 20

04 An introduction to MultiLabel classification
15, Jul 20

05 Understanding Gain Chart and Lift Chart
16, Mar 21

06 Ranking Rows of Pandas DataFrame
14, Jan 19

07 tf-idf Model for Page Ranking
24, Aug 18

08 Python program to find Cumulative sum of a list
04, Jan 18

Article Contributed By :

pawangfg
@pawangfg

Vote for difficulty

Easy Normal Medium Hard Expert

Improved By : adnanirshad158

Improve Article Report Issue

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments

GeeksforGeeks
5th Floor, A-118,
Sector-136, Noida, Uttar Pradesh - 201305

feedback@geeksforgeeks.org

Company

About Us
Careers
In Media
Contact Us
Privacy Policy
Copyright Policy

Learn

Algorithms
Data Structures
SDE Cheat Sheet
Machine learning
CS Subjects
Video Tutorials
Courses

News

Top News
Technology
Work & Career
Business

Languages

Python
Java
CPP
Golang

Knowledge

Web Development

Web Tutorials

Django Tutorial

HTML

JavaScript

Bootstrap

ReactJS

NodeJS

Kotlin

Contribute

Write an Article

Improve an Article

Pick Topics to Write

Write Interview Experience

Internships

Video Internship