The Evaluation Tool in LETOR

Wenying Xiong, Tie-Yan Liu, and Hang Li
Microsoft Research Asia
tyliu@microsoft.com

1. Introduction

Together with the benchmark datasets, we also release an evaluation tool (in the directory \EvalTool\Bin). The tool is written in perl, and can output precision at position n (P@n), mean average precision (MAP) and normalized discount cumulative gain (NDCG) for the ranking results of a given ranking algorithm. By using a single evaluation tool, the experimental results of different methods can be easily and impartially compared.
In this document, we describe the usage of this tool, as well as the evaluation metrics it supports.

2. Command Line

The usage of this tool is as follows.

```
perl  eval.pl  [test file] [prediction file] [output file] [flag]
```

There are four arguments for the command of the tool:
[test file]:
The input file of the evaluation tool. For example, "OHSUMED\Data\Fold*n*\testset.txt". Actually it contains the relevance (label) information of the testset and is used for evaluation.
[prediction file]:
The input file of the evaluation tool, containing the scores given by the ranking algorithm under investigation. Note that each score occupies one line, and corresponds to one document in the testset.
[output file]:
The output file of the evaluation tool, containing the evaluation results of P@n, MAP and NDCG. The content of a typical output file is as follows.

```
precision:
        0.454545454545455       0.454545454545455       0.469696969696970
        0.465909090909091       0.454545454545454       0.439393939393939
        0.415584415584416       0.380681818181818       0.378787878787879
        0.359090909090909       0.359504132231405       0.359848484848485
        0.349650349650350       0.337662337662338       0.333333333333333
        0.323863636363636
MAP:
        0.333595437720468
NDCG:
        0.3333333333333330.318181818181818      0.352612157563246
        0.360305483598161       0.363322348246144       0.368029511530625
        0.357612841369191       0.347125046267113       0.354728173889095
        0.349067397984305       0.351271282191462       0.356009117393668
        0.353631595821978       0.353351726744101       0.357680572013954
        0.35809011638549
```

[flag]:
If flag is '1', the evaluation tool will output per-query evaluation results; otherwise it will output the average evaluation results.

For the instance of using this evaluation tool, one can refer to "perf.cmd" in the directory "EvaTool\Results".

An EXE version of the evaluation tool is also available in the same directory. Its command line (which is shown as follows) is almost the same as the perl version, so we will not describe it with details.

eval.exe *[test file] [prediction file] [output file] [flag]*

3. Evaluation Metrics

As aforementioned, the evaluation tool can output P@N, MAP, and NDCG for a given ranking algorithm. We will describe the definitions of these three metrics in this section.

Precision at position n (P@n)

Precision at *n* measures the relevance of the top *n* results of the ranking list with respect to a given query.

$$P@n = \frac{\#\ relevant\ docs\ in\ top\ n\ results}{n}$$

For example, if the top 10 documents returned for a query are {*relevant, irrelevant, irrelevant, relevant, relevant, relevant, irrelevant, irrelevant, relevant, relevant*}, then P@1 to P@10 values will be {1, 1/2, 1/3, 2/4, 3/5, 4/6, 4/7, 4/8, 5/9, 6/10} respectively. For a set of queries, one averages the P@*n* values of all the queries to get the mean P@*n* value.

Mean average precision (MAP)

For a single query, average precision is defined as the average of the P@*n* values for all relevant documents.

$$AP = \frac{\sum_{n=1}^{N}(P@n * rel\,(n))}{\#total\ \ relevant\ \ \ docs\ \ for\ \ this\ query}$$

where *N* is the number of retrieved documents, and *rel*(*n*) is a binary function on the relevance of the *n*-th document.

$$rel(n) = \begin{cases} 1, if\ \ the\ n^{th}\ doc\ is\ relevant \\ 0, otherwise \end{cases}$$

Similar to mean P@*n*, over a set of queries, we get MAP by averaging the AP values of all the queries.

Normalized discount cumulative gain (NDCG)

Note that P@*n* and MAP can only handle cases with binary judgment: "relevant" or "irrelevant". Recently, a new evaluation metric called Normalized Discount Cumulative Gain (NDCG) has been proposed, which can handle multiple levels of relevance. While evaluating a ranking list, NDCG follows two rules:

1) Highly relevant documents are more valuable than marginally relevant document;
2) The lower ranking position a document (of any relevance level) has, the less valuable it is for the user, because it is less likely to be examined by the user.

According to the above rules, there are four steps to compute the NDCG value for a ranking list:

1) Compute the gain of each document;
2) Discount the gain of each document by its ranking position;
3) Cumulate the discounted gain of the list;

4) Normalize the discounted cumulative gain of the list.

In this evaluation tool, the NDCG value of a ranking list at position *i* is calculated as follow:

$$N(n) \equiv Z_n \sum_{j=1}^{n} \frac{2^{r(j)}-1}{\log(1+j)}$$

where $r(j)$ is the rating of the *j*-th document in the list, and the normalization constant Z_n is chosen so that the perfect list gets a NDCG score of 1. In the above equation, we can see that $2^{r(j)} - 1$ is the gain (G) of the *j*-th document, $\frac{2^{r(j)}-1}{\log(1+j)}$ is the discounted gain (DG), $\sum_{j=1}^{n} \frac{2^{r(j)}-1}{\log(1+j)}$ is the discounted cumulative gain (DCG) at position *n* of the list, and finally $Z_n \sum_{j=1}^{n} \frac{2^{r(j)}-1}{\log(1+j)}$ is the normalized discounted cumulative gain (NDCG) at position *n* of the list, which is called NDCG@*n*.

Note that for the TREC Datasets, there are two relevance levels {0, 1}; and for the OHSUMED Dataset, there are three relevance levels {0, 1, 2}.

4. Additional Note

Users of this evaluation tool need to sign the "Microsoft Research Shared Source License Agreement (Non-commercial Use Only)" provided at the download site, when they download it. For questions or requests, please send email to tyliu@microsoft.com.

This document was last updated on March 1st, 2007.