Review of ways to evaluate an Information Retrieval Systems

Oct 29, 2019
Atri Basu

Introduction

Evaluation of information retrieval systems(IRS) has been actively researched for over 50 years and continues to be an area of exploration. The ability to holistically evaluate such as system is highly important for designing, developing and maintaining effective information retrieval or search systems as it allows the measurement of how successfully an information retrieval system meets its goal of helping users fulfil their information needs. Based on the need there can be many different ways that the performance of an information retrieval system can be analysed:

1. Does it retrieves relevant (compared with non-relevant) documents?
2. how quickly results are returned?
3. how well the system supports users' interactions?
4. are users are satisfied with the results?
5. how easily can users use the system?
6. does the system users carry out their tasks and fulfil their information needs?
7. how reliable is the system?

In this paper I discuss an evaluation framework that focuses on measuring how well an information retrieval system can separate relevant from non-relevant documents for a given user query. I discuss the construction of test collections and the use of standardised benchmarks for evaluating information retrieval systems.

Cranfield Evaluation Methodology

The Cranfield approach to information retrieval evaluation has remained popular for more than fifty years and much of this paper is based on the Cranfield Evaluation Methodology. The Cranfield experiments were computer information retrieval experiments conducted by Cyril W. Cleverdon at the College of Aeronautics at Cranfield in the 1960s, to evaluate the efficiency of indexing systems.They represent the prototypical evaluation model of information retrieval systems, and this model has been used in large-scale information retrieval evaluation efforts such as the Text Retrieval Conference (TREC).

The main purpose of test collection experimentation for information retrieval is to develop and optimize algorithms for locating and ranking a set of documents about the same topic to a given query. The evaluation model relies on three components:

1. a document collection (or corpus)

2. a set of queries
3. a set of relevance judgements, i.e. a file which for each query lists the documents regarded as relevant to answer the given query.

There are widely recognised weaknesses to the Cranfield approach which have not been discussed in this paper, however what this methodology does do is provide us with a good way to establish heuristics even when not all the information is available.

Building Test Collections

When constructing a test collection there are typically a number of practical issues that must be addressed (Sanderson and Braschler 2009). I've found that the easiest approach to solving the problem of building these test collections is:

1. crowdsourcing: the act of taking a job traditionally performed by a designated person and outsourcing to an undefined, generally large group of people in the form of an open call. Amazon Mechanical Turk (AMT) is one such example of a crowdsourcing platform. This system has around 200,000 workers from many countries that perform human intelligence tasks. Recent research has demonstrated that crowdsourcing is feasible for gathering relevance assessments (Alonso and Mizzaro 2009, Kazai 2011, Carvalho *et al.* 2011).
2. Inferred judgements: the internet allows for linking of content. In many cases the linkage may not be a direct implication of relevance judgement but can be used as a great substitute.

Assessing System Effectiveness

Evaluation measures provide a way of quantifying retrieval effectiveness (Manning *et al.* 2008, Croft *et al.* 2009). Together, the test collection and evaluation measure provide a simulation of the user of an information retrieval system. However an important assumption for everything that follows is that the user of the IRS starts browsing results at the top of a ranked list and works their way down examining each document in turn for relevance. This, of course, is an estimation of how users behave; in practice they are often far less predictable.

Set-based Quality Indicators

The first thing that needs to be evaluated for an IRS is how accurately it identifies relevant information. Two simple measures developed early on to do this were *precision* and *recall*. These are *set-based measures*, i.e. documents in the ranking are treated as unique and the ordering of results is ignored.

1. Precision measures the fraction of retrieved documents that are relevant
2. Recall measures the fraction of relevant documents that are retrieved.

Precision and recall hold an approximate inverse relationship: higher precision is often coupled with lower recall. However, this is not always the case as it has been shown that precision is affected by the retrieval of non-relevant documents; recall is not. Compared to other evaluation measures, precision is simple to compute because one only considers the set of retrieved documents (as long as relevance can be judged). However, to compute recall requires comparing the set of retrieved documents with the entire collection, which is impossible in many cases (e.g., for Web search). In this situation techniques, such as pooling, are used.

Often preference is given to either precision or recall. For example, in Web search the focus is typically on obtaining high precision by finding as many relevant documents in the top n results. However, there are certain domains, such as patent search, where the focus is on finding all relevant documents through an exhaustive search. Scores for precision and recall are often combined into a single measure to allow the comparison of information retrieval systems. Example measures include the *e* and *f* measures.

However for these metrics to be effective, we need a binary dataset consisting of queries and a list of acceptable solution. However it's important to recognise that this list will hardly be exhaustive and unless there are at least a certain number of relevance judgements for each query with respect to the size of the document collection, these scores may not be particularly reflective of the IRS's performance. These are important caveats to keep in mind when using these measures.

Ranked Relevance Quality Indicators

One of the most important functions of an information retrieval system is to be able to retrieve content relevant to a users query. However, not all content is of equal relevance. So importance needs to be placed, not only on obtaining the maximum number of relevant documents, but also for returning relevant documents higher in the ranked list. A common way to evaluate ranked outputs is to compute precision at various levels of recall (e.g., 0.0, 0.1, 0.2, ... 1.0), or at the rank positions of all the relevant documents and the scores averaged (referred to as *average precision*). This can be computed across multiple queries by taking the arithmetic mean of average precision values for individual topics. This single-figure measure of precision across relevant documents and multiple queries is referred to as *mean average precision* (or MAP). Another common measure is precision at a fixed rank position, for example *Precision at rank 10* (P10 or P@10). Because the number of relevant documents can influence the P@10 score, an alternative measure called *R-precision* can be used: precision is measured at the rank position *Rq*, the total number of relevant documents for query *q*.

nDCG with a twist

More recently, measures based on non-binary (or *graded*) relevance judgments have been utilised, such as *discounted cumulative gain* (Järvelin and Kekäläinen 2002). In such measures, each document is given a score indicating relevance (e.g., relevant=2; partially-relevant=1; non-relevant=0). Discounted cumulative gain computes a value for the number of relevant documents retrieved that includes a discount function to progressively reduce the importance of

relevant documents found further down the ranked results list. This simulates the assumption that users prefer relevant documents higher in the ranked list. The measure also makes the assumption that highly relevant documents are more useful than partially relevant documents, which in turn are more useful than non-relevant documents. The score can be normalised to provide a value in the range 0 to 1, known as *normalised DCG* (nDCG). The measure can be averaged across multiple topics similar to computing mean average precision.

Challenges

The standard formula for nDCG is as follows:

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

where IDCG is ideal discounted cumulative gain,

$$\mathrm{IDCG_p} = \sum_{i=1}^{|REL_p|} \frac{2^{rel_i} - 1}{\log_2(i+1)}$$

and |RELp| represents the list of relevant documents (ordered by their relevance) in the corpus up to position p.

However, I've found that normalising based on just the ideal or expected DCG value makes it hard to carry out evaluations where, due to sparse feedback, the ideal ranked list and the worst ranked list are not too different. In these situations the nDCG does not make it easy to determine how well the IRS is doing.

Optimization Suggestions

To compensate for this, I've found that normalising the score using both the ideal DCG for a query and the non-ideal DCG helps:

$$nDCG_p = \frac{DCG_p - NIDCG_p}{IDCG_p - NIDCG_p}$$

Here NIDCG or the non-ideal DCG is the exact opposite of IDCG:

$$NIDCG_p = \sum_{i=1}^{|IREL_p|} \frac{Feedback_i}{\log_2(i+1)}$$

and IRELp represents the list of relevant documents (ordered in decreasing order by their feedback) in the corpus up to position p.

Example

Assume that while user is researching relevant documents related to the query document D1, the user provides feedback for six documents D2...D7 as shown below:

Query	Result	Relevance Score	Feedback	Feedback Score
D1	D2	0.87	3	3
D1	D3	0.76	2	2
D1	D4	0.62	3	3
D1	D5	0.59	1	0
D1	D6	0.55	1	0
D1	D7	0.38	2	2

In the above table, the following should be noted:

1. Relevance Score is the score used by the information retrieval system to determine the relevance/similarity between the query and the result document.
2. 1 star feedback is scored as 0 since that's bad feedback

$$DCG_6 = \sum_{i=1}^{6} \frac{Feedback_i}{\log_2(i+1)} = 3 + 1.262 + 1.5 + 0 + 0 + 0.712 = 6.474$$

Ideal Ordering for query D1: D2,D4,D3,D7,D5,D6

$$IDCG_6 = \frac{3}{1} + \frac{3}{1.584962500721156} + \frac{2}{2} + \frac{2}{2.321928094887362} + 0 + 0 = 6.754$$

Non-Ideal Ordering for query D1: D6,D5,D7,D3,D4,D2

$$NIDCG_6 = 0 + 0 + \frac{2}{2} + \frac{2}{2.322} + \frac{3}{2.585} + \frac{3}{2.8} = 4.093$$

Therefore the normalised Discounted Cumulative Gain for query D1 is

$$nDCG_6 = \frac{DCG_6 - NIDCG_6}{IDCG_6 - NIDCG_6} = \frac{6.474 - 4.093}{6.754 - 4.093} = 0.895$$

The average of the scores for each query is considered the overall performance of the information retrieval system

Semantic Relevance Quality Indicators

Apart from accuracy and ranking ability, the performance of an IRS also needs to be evaluated from the perspective of how well the model is able to create semantic separation between relevant and irrelevant documents. To gauge this I've found a combination of metrics to be helpful.

Cross Entropy Loss

One heuristic that tells us how well the IRS is able to separate relevant content from irrelevant content, is to score the results returned in such a way it allows us to penalise models that identify good results with a low semantic similarity and vice-versa. Cross Entropy Loss is one such measure function:

$$Cross\ Entropy\ Loss = \frac{\sum_{i=1}^{n}(y * \log_2(relevance\ score) + (1-y)\log_2(1 - relevance\ score))}{n}$$

Here, y represents whether that result was relevant to the query or not and the relevance score is the similarity score returned by ranker in the IRS.

Semantic Quality Indicator

Another good indicator of how well a model creates separation between good and bad results is by calculating:

1. the average delta, across all queries, in the average similarity score of the query:result pairs which are considered relevant vs that of the irrelevant ones for each query - the higher this value the better the separation between solutions and non-solutions.
2. percentage of queries where the previous delta is greater than 0 - For an ideal IRS this should be 100%.

Example

The same dataset used in Ranked Relevance Quality Indicator will be used for this evaluation.

Using the same sample dataset shown in the example for Ranked Relevance Quality Indicator, we can gather the following:

Q:R Pairs	Relevance Score	y(relevant or not)	Cross Entropy Loss
D1:D2	0.87	1	log2(0.87)
D1:D3	0.76	1	log2(0.76)
D1:D4	0.62	1	log2(0.62)
D1:D7	0.38	1	log2(0.38)
D1:D5	0.59	0	log2(1 - 0.59)
D1:D6	0.55	0	log2(1 - 0.55)

> average relevant relevance score = (0.87 + 0.76 + 0.62 + 0.38)/4 = 0.6575
> average irrelevant relevance score = (0.59 + 0.55)/2 = 0.57
> semantic quality indicator(average) = average relevant relevance score - average irrelevant relevance score = 0.0875

This indicates that there isn't much semantic separation between good results(solutions) and bad results(non-solutions) and so even though the nDCG for this query was high, the low quality indicator indicates there's considerable room for improvement.

average cross entropy loss = [-log2(0.87) - log2(0.76) - log2(0.62) - log2(0.38) - log2(1-0.59) - log2(1-0.55)]/5 = 1.024

A high cross entropy loss indicates that there isn't much semantic separation between good results(relevant) and bad results(irrelevant) which validates our findings from using the semantic quality indicator.

Summary

Coming up with a holistic evaluation of an Information Retrieval system requires planning:

1. The goals of the evaluation must be defined
2. a suitable test collection must be selected from those already in existence, or must be created specifically for the retrieval problem being addressed
3. different information retrieval systems or techniques must be developed or chosen for testing and comparing
4. evaluation measures and statistical tests must be selected for evaluating the performance of the information retrieval system for comparing whether one version of the system is better than another.

Challenges to evaluating the overall performance of an IRS

Some of the major problems surrounding the process of evaluating IRSs have been discussed in the paper by Tague-Sutcliffe 1996. Here I've outlined some of the challenges I ran into when trying to build an evaluation framework for my IRS:

1. Modelling ad hoc retrieval - the original and most common problem is addressing the situation in which an information retrieval system is presented with a previously unseen query. For obvious reasons, it is impossible to tell how well an IRS will perform for a completely unseen query. However, using this methodology one can get a reasonable intuition into how it might perform.
2. Lack of labelled data - by labelled data, I mean relevance judgements pertaining to query:result pairs. Human tendency is to only give feedback when things aren't working. In the case of an IRS, that means if users are getting good results they have no incentive to rate those results or provide feedback and when things aren't working all of the feedback is negative, which isn't particularly helpful in evaluating how *well* the IRS is doing. Requiring users to label specific query:result pairs is a tedious affair which has multiple drawbacks. Given the bottleneck in gathering relevance judgments, 'low-cost evaluation' techniques have been proposed. These include approaches based on focusing assessor effort on runs from particular systems or topics that are likely to contain more relevant documents (Zobel 1998), sampling documents from the pool (Aslam *et al.* 2006), supplementing pools with relevant documents found by manually searching the document collection with an information retrieval system, known as *interactive search and judge* or ISJ (Cormack *et al.* 1998) and simulating queries and

relevance assessments based on user's queries and clicks in search logs (Zhang and Kamps 2010). These methods will not be covered in this paper.

3. Lack of SME agreement and Errors in human labelling - even when the problem of collecting relevance judgements is solved, I have found that it is not easy to get any two Subject Matter Experts(SME) to agree on the relevance judgement for many query:result pairs. previous studies have also demonstrated that domain expertise can have an impact on the quality and reliability of relevance judgments (Bailey *et al.* 2008, Kinney *et al.* 2008) and is particularly problematic when using crowdsourcing in specialised domains (Clough *et al.* 2012). Furthermore, inappropriate understanding of the feedback scales, and definitions can also lead to multiple errors in judgement.

There are also further complications that must be considered. For example, research has shown that users are more likely to select documents higher up in the ranking (*rank bias*); measures typically assume that no connection exists between retrieved documents (*independence assumption*); and particularly in the case of Web search, one must decide how to deal with duplicate documents: should they be counted as relevant or ignored as they have been previously seen?

These decisions are important, as they will affect the quality of the benchmark and impact the accuracy and usefulness of results. In practice it is important to select an evaluation measure that is suitable for the given task; for example, if the problem is known-item search then the mean reciprocal rank might be appropriate; for an *ad hoc* search task with non-binary relevance judgements, the averaged normalised discounted cumulative gain would be more applicable.

Over the years, test collection-based evaluation has been highly influential in shaping the core components of modern information retrieval systems, including Web search. This has been particularly visible in the context of TREC and related studies, which have not only provided the necessary benchmarks to compare different approaches to retrieval, but also provided a community and forum in which it can be discussed and promoted.