Metrics

Let's explore some metrics that will help you define the “success” scenarios for a search session.

We'll cover the following

Choosing a metric for a machine learning model is of paramount importance. Machine learning models learn directly from the data, and no human intuition is encoded into the model. Hence, selecting the wrong metric results in the model becoming optimized for a completely wrong criterion.

There are two types of metrics to evaluate the success of a search query:

1. Online metrics
2. Offline metrics

We refer to metrics that are computed as part of user interaction in a live system as Online metrics. Meanwhile, offline metrics use offline data to measure the quality of your search engine and don’t rely on getting direct feedback from the users of the system.

feedback from the users of the system.

Online metrics#

In an online setting, you can base the success of a search session on *user actions*. On a *per-query level*, you can define success as the user action of *clicking on a result*.

A simple click-based metric is *click-through rate*.

Click-through rate#

The click-through rate measures the ratio of clicks to impressions.

> 📝 Click through rate = $\frac{Numberofclicks}{Numberofimpressions}$

In the above definition, an *impression* means a view. For example, when a search engine result page loads and the user has seen the result, you will consider that as an impression. A click on that result is your success.

Successful session rate#

One problem with the *click-through rate* could be that unsuccessful clicks will also be counted towards search success. For example, this might include short clicks where the searcher only looked at the resultant document and clicked back immediately. You could solve this issue by filtering your data to only successful clicks, i.e., to only consider clicks that have a long dwell time.

> 📝 *Dwell time* is the length of time a searcher spends viewing a webpage after they've clicked a link on a search engine result page (**SERP**).

Therefore, successful sessions can be defined as the ones that have a click with a ten-second or longer dwell time.

> 📝 Session success rate = $\frac{no.\ of\ successful\ sessions}{no.\ of\ total\ sessions}$

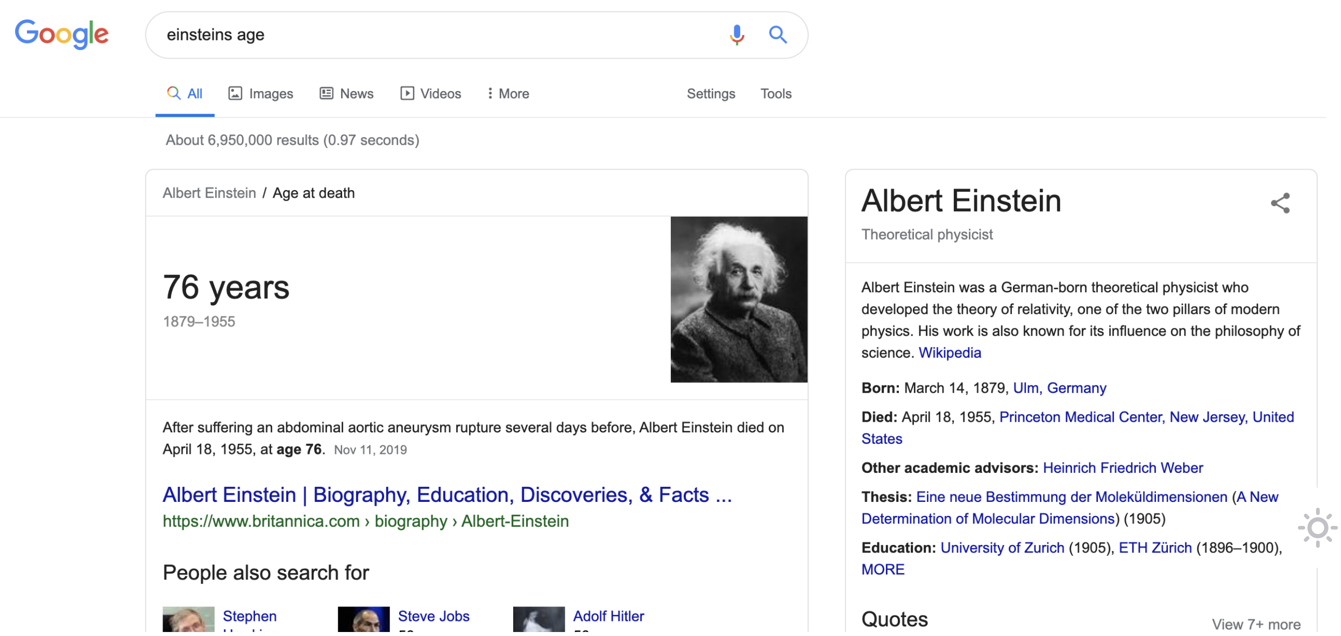
A session can also be successful without a click as explained next.

Caveat#

Another aspect to consider is *zero-click searches*.

> 📝 **Zero-click searches**: A SERP may answer the searcher's query right at the top such that the searcher doesn't need any further clicks to complete the search.

For example, a searcher queries "einstein's age", and the SERP shows an excerpt from a website in response, as shown below:



About 6,950,000 results (0.97 seconds)

Albert Einstein / Age at death

76 years

1879–1955

After suffering an abdominal aortic aneurysm rupture several days before, Albert Einstein died on April 18, 1955, at **age 76**. Nov 11, 2019

Albert Einstein | Biography, Education, Discoveries, & Facts ...

https://www.britannica.com › biography › Albert-Einstein

People also search for

Stephen Steve Jobs Adolf Hitler

Albert Einstein

Theoretical physicist

Albert Einstein was a German-born theoretical physicist who developed the theory of relativity, one of the two pillars of modern physics. His work is also known for its influence on the philosophy of science. Wikipedia

Born: March 14, 1879, Ulm, Germany

Died: April 18, 1955, Princeton Medical Center, New Jersey, United States

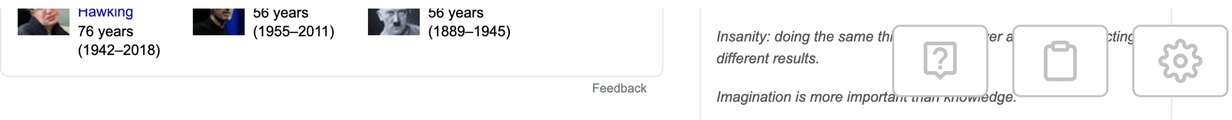
Other academic advisors: Heinrich Friedrich Weber

Thesis: Eine neue Bestimmung der Moleküldimensionen (A New Determination of Molecular Dimensions) (1905)

Education: University of Zurich (1905), ETH Zürich (1896–1900), MORE

Quotes

View 7+ more



Insanity: doing the same th… …er a… …cting different results.

Imagination is more important than knowledge.

The searcher has found what they were looking for without a single click!. The click-through rate would not work in this case (but your definition of a successful session should definitely include it). We can fix this using a simple technique shown below.

Time to success

Until now, we have been considering a single query-based search session. However, it may *span over several queries*. For example, the searcher initially queries: “italian food”. They find that the results are not what they are looking for and make a more specific query: “italian restaurants”. Also, at times, the searcher might have to go over multiple results to find the one that they are looking for.

Ideally, you want the searcher to go to the result that answers their question in the minimal number of queries and as high on the results page as possible. So, *time to success* is an important metric to track and measure search engine success.

> 📝 **Note:** For scenarios like this, a *low number of queries per session* means that your system was good at guessing what the searcher actually wanted despite their poorly worded query. So, in this case, we should consider a *low number of queries per session* in your definition of a successful search session.

Offline metrics#

The offline methods to measure a successful search session makes use of trained human raters. They are asked to rate the relevance of the query results objectively, keeping in view well-defined guidelines. These ratings are then aggregated across a query sample to serve as the *groun* *h.*

> 📝 Ground truth refers to the actual output that is desired of the system. In this case, it is the ranking or rating information provided by the human raters.

Let's see *normalized discounted cumulative gain* (NDCG) in detail as it's a critical evaluation metric for any ranking problem.

NDCG#

You will be looking at NDCG as a common measure of the quality of search ranking results.

NDCG is an improvement on *cumulative gain* (CG).

> 📝 $CG_p = \sum_{i=1}^{p} rel_i$
>
> $where\ rel = relevance\ rating\ of\ a\ document,\ i = position\ of\ document,\ and\ p = the\ posit$

Example

Let's look at this concept using an example.

A search engine answers a query with the documents D_1 to D_4. It ranks them in the following order of decreasing relevance, i.e., D_1 is the highest-ranked and D_4 is the lowest-ranked:

Ranking by search engine: D_1, D_2, D_3, D_4

A human rater is shown this result and asked to judge the relevance of each document for the query on a scale of $0 - 3$ (3 indicating high e

They rank the documents as follows:

Documents	D_1	D_2	D_3	D_4
Ranking/Rating by human rater	3	2	3	0

The *cumulative gain* for the search engine's result ranking is computed by simply adding each document's relevance rating provided by the human rater.

$$CG_4 = \sum_{i=1}^{4} rel_i = 3 + 2 + 3 + 0 = 8$$

In contrast to cumulative gain, discounted cumulative gain (DCG) allows us to *penalize the search engine's ranking if highly relevant documents (as per ground truth) appear lower in the result list*.

> 📝 DCG discounts are based on the position of the document in human-rated data. The intuition behind it is that the search engine will not be of much use if it doesn't show the most relevant documents at the top of search result pages. For example, showing starbucks.com at a lower position for the query *"Starbucks"*, would not be very useful.

> 📝 $DCG_p = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)}$

DCG for the search engine's ranking is calculated as follows:

In the above calculation, you can observe that the denominator penalizes the search engine's ranking of D_3 for appearing later in the list. It was a more relevant document relative to D_2 and should have appeared earlier in the search engine's ranking.

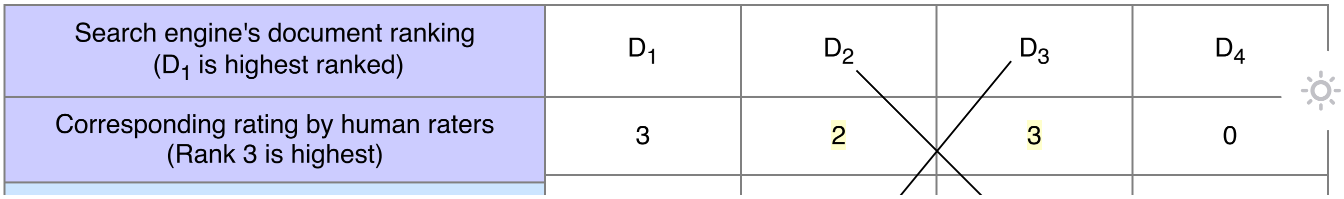
However, DCG can't be used to compare the search engine's performance across different queries on an absolute scale. The is because the length of the result list varies from query to query. So, the DCG for a query with a longer result list may be higher due to its length instead of its quality. To remedy this, you need to move towards NDCG.

NDCG normalizes the DCG in the 0 to 1 score range by dividing the DCG by the max DCG or the IDCG (ideal discounted cumulative gain) of the query. IDCG is the DCG of an ideal ordering of the search engine's result list (you'll see more on IDCG later).

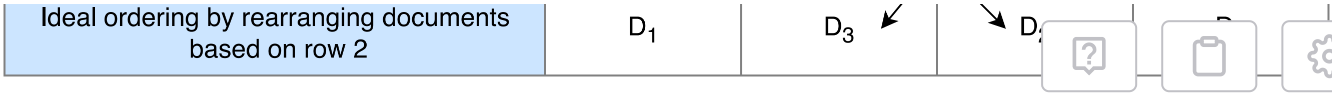
NDCG is computed as follows:

> 📝 $NDCG_p = \frac{DCG_p}{IDCG_p}$ $where\ IDCG\ is\ ideal\ discounted\ cumulative\ gain$

In order to compute IDCG, you find an ideal ordering of the search engine's result list. This is done by rearranging the search engine's results based on the ranking provided by the human raters, as shown below.



Search engine's document ranking (D_1 is highest ranked)	D_1	D_2	D_3	D_4
Corresponding rating by human raters (Rank 3 is highest)	3	2	3	0



Finding the ideal ordering of search engine's result based on ground truth

Now let's calculate the DCG of the ideal ordering (also known as the IDCG) as shown below:

$IDCG_4$ = 5.898

Now, let's finally compute the NDCG:

NDCG = $\frac{DCG}{IDCG}$ = $\frac{5.762}{5.898}$ = 0.976

An NDCG value near one indicates good performance by the search engine. Whereas, a value near 0, indicates poor performance.

To compute NDCG for the overall query set with N queries, we take the mean of the respective NDCGs of all the N queries, and that's the overall relevance as per human ratings of the ranking system.

> NDCG = $\frac{\sum_{i=1}^{N} NDCG_i}{N}$

Caveat#

NDCG does not penalize irrelevant search results. In our case, it didn't penalize D_4, which had zero relevance according to the human rater.

Another result set may not include D_4, but it would still have the same NDCG score. As a remedy, the human rater could assign a negative relevance score to that document.

← Back

Problem Statement

Next →

Architectural Components

☑ Mark as Completed

Report an Issue