**New Metric Design**

1. Data: Positions of results on SERP.

Now we have the position information (top, down, left and right) of each result on the four systems (sogou, sm, haosou and baidu).

1. Metric Design

As the first step to design a new metric (Position-biased Gain or something), I take the following four aspects into consideration:

1. Normalized or not?

A very intuitive form of the metric is like this:

where p is the position on the SERP.

Now we are using the Top-5 results of each SERP (since we have annotations from sogou). But they are not necessarily equal in length. Therefore, in my opinion the metric with Normalization would be better.

I compared the between-metric agreement and metric-user-preference agreement of different new designed metrics. I found that, for all version of metric pairs (we are controlling the other factors), the normalized one almost outperforms the non-normalized one.

This results are not surprising since the SERPs from different systems varies in length. In the following result part, I only report the performance of the normalized metrics to reduce the size of result table.

1. Decay Function

Now I tried the these four version of decay function:

1. nDCG, I used the 1/(1+log\_b\_p) form, p is estimated by the height of an organic result.
2. Linear, with regard to the first ViewPort. We assume that there is no need to decay the results on the first ViewPort( [0:440] ), after the first ViewPort, decay linearly.
3. Linear, without reward to the first ViewPort. Decay linearly from 0 to 1200 (an arbitrary number, at about the third ViewPort).
4. TBG-like decay function. It looks like , where h is a parameter (I can tune this parameter, also estimated from user study. At this moment, I just pick up a number to make sure that it looks like the other decay function).

The decay functions are presented in Figure. 1

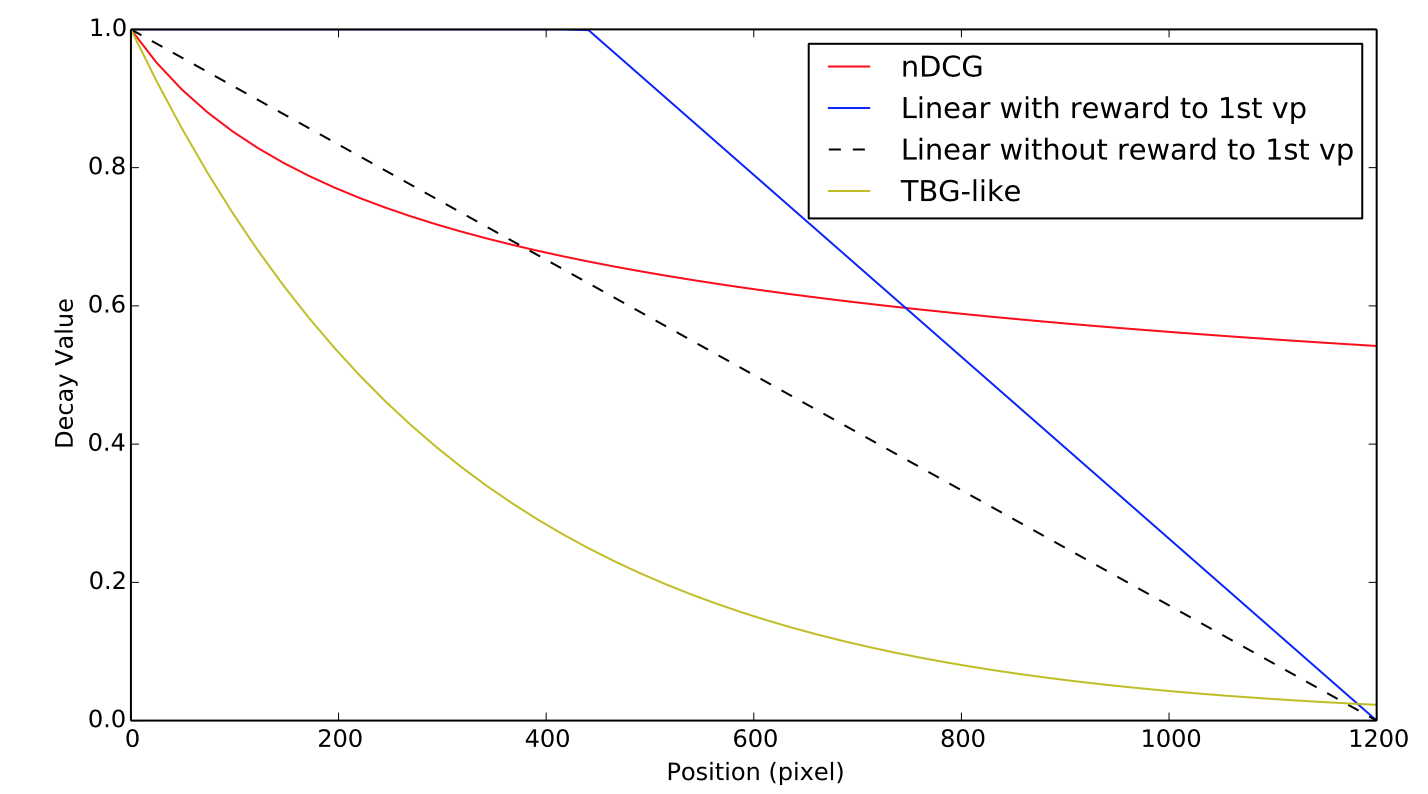


Figure 1. Decay function

1. Gain Function

Using the following two gain functions:

1. ‘raw’: the original score range from 0 to 5.
2. ‘exp’: using 2^r, which is similar with nDCG
3. Gain Distribution

The gain of a certain result could be distributed in the “result area” in different ways. I tried the following two distribution:

1. Uniformly distributed in the top 20% area of the result.
2. The gain follows a binominal distribution which is presented in Figure 2.

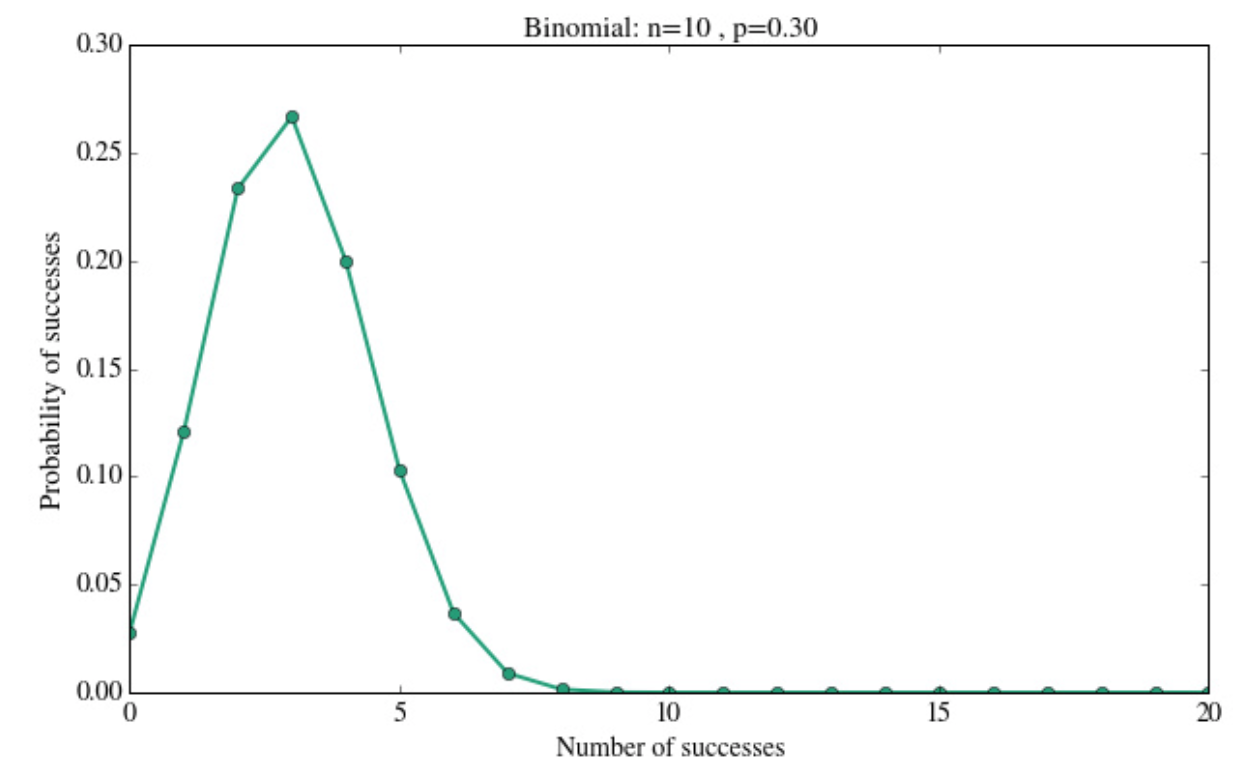


Figure 2. Gain Distribution: the x-axis is vertical position while y-axis is possibility.

1. New Metrics

Now We can develop metrics by combining these factors.

Each Metrics name can be divided into these parts:

Nor-Decay func-Gain func-Gain distribution

Nor: Normalized metric

Decay func: tbg/ndcg/ linear with reward to 1st vp (lwfv) /linear without reward to 1st vp (lwofv)

Gain func: exp/raw

Gain distribution: Step (uniformly on top 20%) / binominal

1. Results

We have 2 results here.

The first one is to compare the inner agreement (kendall’s tau) between different metrics including traditional metrics and our new developed metrics. The results are presented in attached *20161025metric\_inner\_consistent.xlsx.* The data in purple indicates the comparison between tradition metrics and new metrics.

This table shows that some of our new metrics correlates well (tau>0.7) with major classical evaluation metrics (nDCG@3 etc.) The metrics with linear decay functions have poor performance while the metrics with “ndcg” or “tbg” decay functions are much better.

A very natural explanation is that the metrics with “ndcg” or “tbg” decay functions are very similar to classical evaluation metrics.

The second result is the agreement between evaluation metrics and side-by-side user preference. The stats. is in attached *20161025\_metric\_sbs\_agreerate.xlsx.*

We can see that the new metric “Nor-tbg-raw-step” got the best performance among new developed metrics (agree rate 67.3%).

1. **What’s next?**

I guess I can tune the parameters in the metrics to get a “better” performance. Now I am wondering what’s the difference between mobile search evaluation and traditional desktop search evaluation.

Here are some points:

In heterogeneous environment, especially on mobile device, result redundancy becomes very important. Given the context of a query, just relevance seems not enough to describe how good a result is. For example, for a query “Tokyo weather”, maybe the first result is good enough and the remaining results are not important.