On the relationship between click rate and relevance for search engines

K. Ali & C. C. Chang
Yahoo! Inc, USA

Abstract

Evaluation of search engine result relevance has traditionally been an expensive process done by human judges. Researchers have sought cheap automated proxies for such judgments. This paper examines the relationship between relative click rates (of two engines) and relative human judgments of result sets returned by those engines. Previous work has indicated that human judgments are more consistent if provided in a relative form. We additionally observe that clicks are a function not only of the clicked result, but also of its competing neighborhood. These observations force an experimental design where we collect relative judgments of *sets* of results, rather than judgments on individual results. We conduct a large empirical study using forty judges, thousands of live users and hundreds of queries. Our results comparing Yahoo with another search engine in October 2003 show that in aggregate, higher click rate is indicative of higher relevance but the strength of the association is only moderate 40%. Qualitative analysis suggests the association is not stronger because users click for reasons other than relevance such as curiosity and confusion. However, there are classes of queries (such as navigational queries) for which click rates are good indicators of relevance.
Keywords: information retrieval, evaluation, relevance, modeling, statistical tests, Bootstrap, Wilcoxon, correlation, association.

1 Introduction

The predominant methodology for evaluating the quality of information retrieval systems is based on per-document relevance judgments. Given a set of topics, documents for each topic, and per-document judgments, metrics for precision are computed and compared across different systems [1]. For search engines, implicit user behavior in the form of click data has been assumed to be a key proxy for

WIT Transactions on Information and Communication Technologies, Vol 37, © 2006 WIT Press
www.witpress.com, ISSN 1743-3517 (on-line)
doi:10.2495/DATA060221

relevance. Click data has been used by engines such as DirectHit to re-rank search results. Researchers have also investigated the use of click rates to evaluate and improve the engines. [2].

Our methodology differs from previous work in the following ways:

1. **Scale:** Using a large scale study on Yahoo's logs we confirm the results [3] Joachims obtained (for 3 users!) that at an aggregate level, relative click rate is *directionally* predictive of relative relevance.
2. **Set-level:** We use judgments of *sets* of results rather than individual results. We observe that click rates are a function of the *set* of results hence judgments also need to be at the set level. Set-level judgments are also sensitive to ordering, duplication and coverage of multiple meanings of the query.
3. **Stratification:** We show that for some query classes (e.g. navigational queries) there is a high degree assocation between click rate and relevance.

Judgments were obtained by 40 available in-house expert editors - although the use of experts introduces a bias, we have found in internal studies [4] that alternatives such as "random" panelists and live user surveys also have their own biases. Out of the judged queries, 236 of them were clicked during the two week live search engine experiment. Clicks from eighteen thousand anonymous users searching those queries were recorded.

Our key findings are:

- Averaged over hundreds of queries, greater click rate *is* directionally predictive of greater relevance.
- The degree of association between these variables is a moderate 40%.
- Qualitative study indicates users click for many reasons beyond relevance: such as curiosity, surprise or confusion.
- There are query classes (e.g. navigational) for which we do get a much higher association rate (70%).

The rest of the paper is organized as follows. Section 2 covers related work. Section 3 presents our methodology and section 4 presents results and discussion for our three major experiments.

2 Previous work

Frei, Schauble (1991): Frei and Schauble have promoted the use of relative judgments based on work from the 1960s [5] which showed humans give more consistent judgments when comparing objects. Frei and Schauble seek to judge two *sets* of results but by using pairwise result-level judgments. The orderings produced by the systems are compared with an idealized ordering as constructed by human judgments. Their work is similar to ours in using set-level notions but by seeking result-level judgments it does not penalize duplicates or missing meanings.

Joachims (2002) [3]: In similarity to our work, Joachims uses the interleaved setting (which he pioneered) to collect relative click rates but he differs in using result-level absolute judgments rather than relative set-level. To see why it is important to measure relative click rates via interleaved presentation, consider

the scenario where one engine is significantly worse than the other. If we were to collect click rates by presenting one set of users with engine A results, and the other with B, then we would find that the poor engine would nevertheless get a good number of clicks. This is because users faced with poor choices tend to click on some results anyway. However, in an interleaved setting, users would assiduously avoid the poor engine's results and hence provide a better comparison of the engines. Joachim's paper has a small empirical study based on 3 users from a university web log. Comparing Google and MSN in 2001, he shows that using a binomial test (e.g. [6]) Google has statistically significantly more clicks and statistically significantly more relevant results.

Workshop on Implicit Measures of User Interests and Preferences (2003): [7] The term "implicit measures" refers to user actions such as clicks as opposed to explicit measures such as answers to survey questions. Fox finds that models based on click rate alone are substantially improved using client-side features such as dwell time. Some of these powerful features cannot be measured at the server hence could not be used in our work.

3 Methodology

3.1 Query selection

Using Yahoo's web log, we collected a random sample of 500 queries. Internal studies [8] had shown that approximately 500 queries was sufficient for the statistical resolving power [6] we sought. There are 2 reasonable ways we could have formed the sample: A) unique the set of queries and then select from that set, or B) select queries randomly from our event log. We chose method B, thereby getting a frequency-biased set of queries. We did not filter out adult, foreign-language or misspelled queries.

3.2 Relative Set-level judgments (RJ)

Judges were shown queries 10 at a time and asked to select a query they felt they could judge. Such self-selection introduces a bias against obscure queries that may not be selected by any judge, but we feel self-selection is inevitable because we cannot ask judges to judge queries they are do not know. Out of the 500 queries, 486 were selected. After selecting a query, the judge was shown two unlabeled vertical panels of results. Both engines had a 50% chance of being on the each side. Internal studies confirmed sidedness was not statistically significant in predicting the winning engine. The same abstract-creation algorithm was used for results from both engines so as to remove "goodness of abstract" as a factor. The judge had the freedom to not click on any of the results, if she felt so. This means some judges' decisions would be a function only of the results page, ordering of results and abstracts - not of the website itself. We had to accept this methodological trade-off if were to allow the judges to not click. The judge renders a judgment on a scale from -3 to +3. In addition she could also select a reason for her decision. If a

query received more than one judgment (across judges) we averaged the judgments together.

3.3 Relative Click Rates (RCTR)

Relative click rates were collected by showing interleaved results to a small fraction of Yahoo's live users during a two week trial. We used Joachim's interleaving algorithm [3] to interleave results from the engines. The algorithm begins by picking one engine at random. This choice is made separately for each query so approximately 50% of the queries end up with engine A guaranteed to be at position 1 and 50% for engine B. The first result from this engine forms the first result of the interleaved set. A count is kept for each engine of the number of results used from that engine. If the result (URL) is also present (in the top 10) of the other engine, then the count for that engine is also incremented. For the next result, the algorithm picks the engine with the smaller count. If both engines have the same count, it picks an engine at random. Joachim shows that this process has the property that at any point in the process, the number of results chosen from A and B do not differ by more than 1.

Using this data, RCTR for a single query q is defined as follows.

$$RCTR(q) \equiv \frac{n_A(q) - n_B(q)}{n_A(q) + n_B(q)} \tag{1}$$

where $n_X(q)$ is the number of clicks on results of engine X. The denominator serves to normalize for the differing number of clicks across queries. Since in the end, to compute association (correlation) we will do a scatter-plot of RCTR versus RJ at the query level we need to do this normalization.

3.4 Correlation and association measures

To measure the degree of correlation between continuous RCTR and continuous RJ we used Spearman's rank correlation coefficient r [6] (this is as opposed to the more commonly used Pearson's linear correlation coefficient [6] which is only suitable for continuous *linear* relationships). Let c_q be the relative click rate for query q and let j_q be its average relative judgment. Spearman's correlation works as follows. Rank the relative click rates. Let c'_q be the rank of query q. Now rank the relative judgments. Let j'_q be the rank of query q. The set (c'_q) of ranks of relative click rates forms observations for a random variable C with sample mean $E(C)$ and sample standard-deviation $\sigma(C)$ (similarly for J corresponding to (j'_q)). Then Spearman's correlation coefficient is simply Pearson's linear correlation coefficient [6] but applied to the ranks. Thus, Spearman's correlation coefficient is given by:

$$\hat{r}(C, J) \equiv \frac{1}{n} \sum_q \frac{(c'_q - E(C))(j'_q - E(J))}{\sigma(C)\sigma(J)} \tag{2}$$

Values range from -1 (perfect negative correlation) to 0 (no correlation) to +1 (perfect positive correlation).

3.4.1 Discretization

In addition to measuring continuous correlation between RCTR and RJ, we also discretized these random variables to compute a discrete degree of association: a "-1" value indicates engine A was better, "+1" where B was better. (Queries for which either random variable is exactly zero are not counted.) Thus each query can have one of 4 possible outcomes with regards to RCTR and RJ. For this we use a classic 2x2 contingency matrix approach [9]. Correlation for such discretized (ordinal) variables is termed "association" and there are several measures which differ in their interpretation. We use Cramer's V (Φ) [9] because it has an easy interpretation of its values. Φ is a normalized chi-squared based measure of association which depends on the strength of association between two ordinal-level random variables and corrects for smaller sample sizes. Its interpretation is that it measures the observed level of association as a fraction of the maximum possible level of association [9] between the two random variables. So, for example, a value of 0.40 means that 40% of the variation in one random variable is reflected in the other.

3.4.2 Filters

In addition to measuring the association, we considered filters to remove queries for which we could not reliably measure click rates or judgments:

- A significant number of queries received less than 5 clicks. With this few clicks, under re-sampling [10], the RCTR could easily flip between -1 and +1. Thus we decided to remove these "low-data" queries. We chose 5 because, using a binomial test, five is the minimum number of Bernoulli trials needed to get a distribution (5,0) that is able to distinguish one engine from another at the 95% confidence level ($p = 0.05$).
- Filter 2 removes adult and pogo (e.g. "unclaimed money") queries. Users tend to click regardless of relevance for such queries so these queries add noise to our experiment.
- Filter 3 removes queries judged by a single judge. Using only multiply judged queries improves reliability of the RJ value.

3.4.3 Bootstrap re-sampling

Applying these filters means fewer queries will be left so we have to use statistical tests to determine if a filter statistically significantly improves the association level. Even though we only have one set of queries from which only one association level can be computed, using Bootstrap re-sampling [10] we can bootstrap this set into 1000 sets with 1000 values of association (for each row in table 2). Then we use the non-parametric Wilcoxon rank-sum test (e.g. [6]) to compare the 1000 pre-filter values with the 1000 post-filter values. Bootstrap re-sampling creates multiple sets from one in the following manner. It works by taking an initial set of observations and then sampling (*with replacement*) from that set to produce another set of the

same size. This set is called a replicate and it may contain duplicates of some observations in the initial set. We then measure the statistic of interest (in our case association) on the replicate. Then we repeat this procedure a large number (say 1000) of times to produce 1000 values of association. Finally, we submit these values to the Wilcoxon test which produces a p-value. If the p-value is less than 0.05, we say the filter has statistically significantly altered the association.

3.5 Methodology for query classes

In our third experiment we ask if there are query classes for which we observe significantly higher or lower levels of association than the general query population. We use three methods to collectively define six query classes. The "lexical" method categorizes queries by the number of words in the query. The distributional method categorizes queries according to the spread of distribution of clicks on various results for that query. For example, this method is used to define the "navigational" query class: one in which 95% or more of all clicks go to the first result. This method also defines the "pogo" class: a query is pogo if its mean click position is more than 4. The third ("Semantic") method for defining query classes is based on n-gram analysis [11] and explicit lists. Using this method we define two more query classes: Entertainment and Adult.

4 Results

We will conduct three experiments:

- E1: **Aggregate Direction:** Does the engine receiving more clicks *in aggregate over our query-set* also receive the greater aggregate relative relevance score?
- E2: **Quantify Degree of Association:** What is the degree of association between relative click rates and relative judgments?
- E3: **Query Classes:** How does this level of association differ for some common query classes?

4.1 E1: Aggregate Direction results

Table 1 shows that at an aggregate level, we have agreement between RCTR and RJ: both methods picked engine A (since the "A $>$ B" number was higher than "A $<$ B" for both). Next, we seek to understand more deeply how the association holds up on particular queries and query classes and this is discussed in the next section.

4.2 E2: Association level of RCTR and RJ

We now present the main results of the paper. The first result is the finding of a 28% association level (table 2). Improvements to this association level were obtained by applying quality filters defined in section 3.4. Next, table 3 shows five randomly

Table 1: Directional agreement between Relative Set-level judgments and Relative CTR.

	A > B	A = B	A < B
Relative Judgments	40%	27%	34%
Relative CTR	32%	45%	23%

Table 2: Filtering queries with unreliable values of RCTR or RJ produces higher association level. Bold font indicates statistically significant change with respect to the preceding row.

Filter	Φ	N
None	0.28	236
Remove low-data queries	0.29	178
Remove adult and pogo queries	**0.34**	108
Remove singly-judged queries	**0.40**	89

Table 3: Sample queries in 2x2 contingency table.

	RJ = -1	RJ = +1
RCTR = -1	craigslist cheap tickets fhm friendfinder playstation 3	whitehouse slovenia baseball briana banks cards
RCTR = +1	amex auto trader holy grail kazaa nike golf	aim plus buddyicons yahoochat lord and taylor chicago

chosen queries in each of the four cells of the contingency table. We examined the judge's comments for queries in the off-diagonal cells in detail to understand why, for those queries, judgments favored one engine while clicks favored the other.

4.2.1 Discussion

Detailed examination of the results in the first row of table 2 inspired the filters in that table. Queries receiving few (less than 5, say) clicks would be prone to switching their signs of RCTR and RJ (bouncing around the quadrants of the 2x2 contingency table) if we were to repeat the experiment and hence they are adding noise to the association measurement. However, as table 2 shows in the row 'Remove low-data queries', removing such queries did not significantly increase the association level. Next, we removed adult and pogo queries because we knew users click on results of such queries pretty much independently of relevance. Removing these queries did statistically significantly (Bootstrap analysis, Wilcoxon test) increase the association to a 34% level. Finally, we removed queries that received judgment from just a single judge. In examining comments of the judges, we had seen some disagreement between the judges specially for queries where both engines were close in quality. Removing such queries increased association by a statistically significant amount to 40%. Note that all these filters have the bias of removing niche queries so our we are careful to note our conclusions are only for non-niche queries.

4.2.2 Qualitative analysis

In order to get a deeper understanding of particular queries, we look at the judges' reasons for their judgments. Of course there is a danger in this form of analysis that one can generalize from anecdotes, but on the other hand, looking only at numeric association values also has the danger that one does not gain insight.

For this analysis we examine queries in off-diagonal cells in table 3. Consider the query "whitehouse": judges demoted one engine because its first result was an adult site. However, RCTR favored that engine. If most live users were searching for the adult site, this discrepancy would mean the judges are not representative of the live users. Or, if the live users were searching for the government site, more clicks could arise due to curiosity and surprise rather than greater relevance. This illustrates the limits of analyzing niche queries. Unless our judge panel has sufficient number of judges familiar with that query and why it would be issued by live users, we cannot expect RJ to model RCTR for niche queries.

In internal studies [4] we recruited OpinionSite panelists (purportedly random Internet users) and measured the association between their relative set-level judgments and RCTR. Surprisingly, the association level between their judgments and RCTR was 20% *less* than between expert judgments and RCTR (from live users). In conclusion, whether using judges or "random" panelists, it is difficult to get judgments in line with intents of live users. Even surveying live users would not solve the problem since the set electing to take the survey would also have a significant bias.

4.3 E3: Query Classes

In this section we examine six interesting query classes as defined earlier in section 3.5 to see if their association levels differ from the general query population. The results are in table 4.

Table 4: RCTR-RJ association levels for various query classes.

Method	Query Class	Φ
Distributional	Navigational	0.71
Lexical	Single word	0.70
Semantic	Entertainment	0.59
Lexical	Two word	0.38
Semantic	Adult	0.23
Distributional	Pogo	0.06

4.3.1 Discussion

Table 4 shows that navigational queries have the highest relevance to click-rate association: about 70% (recall that navigational queries are those for which at least 95% of clicks are at the first result). Examples of such queries are "American Airlines" and "waltdisneyworld". The high level arises because user intents are homogeneous for such queries so there is less disagreement between users and judges and amongst judges: if one engine is better, everyone decides the same way. Surprisingly, single word queries also have a high association. We had expected a low association level, thinking such queries are ambiguous. However, many of these queries (e.g. "Madonna") turned out to be iconic with well-understood meanings. Adult and pogo queries on the other hand had a low association level as expected and just add noise to studies such as ours.

5 Conclusions

This paper has examined whether relative click rate is correlated with relative (set-level) judgments. We have conducted a relatively large-scale study using dozens of experts, hundreds of queries from Yahoo web logs, and clicks from eighteen thousand anonymous live Yahoo users.

Our first finding is that averaged over hundreds of queries, higher click rate does indeed imply higher relevance (over non-niche queries). Our second finding is that the strength of this association is only a middling 40% given the methodology of our study (expert judges, self-selected queries, judgments based on abstract and/or landing page). Digging deeper with qualitative analysis confirms widely-held beliefs that this is because clicks are issued for reasons beyond relevance, such as curiosity and surprise. Our third finding is that there are query classes with high levels of association that may permit click rate to be a proxy for human-judged relevance. These classes are navigational and one-word queries. Conversely, we also found classes with very low correlation (e.g. adult queries). If one's intent is to use click rate as an automated proxy for relevance to tune search engines, this study shows one can only do so for limited query classes.

References

[1] Cleverdon, M.K.C.W. & Mills, J., *Factors determining the performance of indexing systems. Volume I - Design, Volume II - Test Results*. ASLIB Cranfield Project, 1966.
[2] Joachims, T., Optimizing search engines using clickthrough data. *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*, ACM, 2002.
[3] Joachims, T., Evaluating retrieval performance using clickthrough data. *Proceedings of the SIGIR Workshop on Mathematical and Formal Models in Information Retrieval*, 2002.
[4] Ali, K. & Chang, C., Comparison of editor and panelist judgments. Technical Report Alatheia 2004-01, Yahoo, 2004.
[5] Lesk, M. & Salton, G., Relevance assessments and retrieval system evaluation. *Information Storage and Retrieval*, pp. 343–359, 1968.
[6] Venables, W.N. & Ripley, B.D., *Modern applied statistics, 4th edition*. Springer-Verlag: New York, 2002.
[7] Dumais, S., Bharat, K., Joachims, T. & Weigend, A., Workshop on implicit measures of user interests and preferences. *Proceedings of SIGIR 2003*, 2003.
[8] Scarr, M., Power study. Technical Report MLR 2005-01, Yahoo, 2005.
[9] Agresti, A., *Categorical Data Analysis*. Wiley, 1990.
[10] Efron, B. & Tibshirani, R.J., *An Introduction to the Bootstrap*. Chapman & Hall, 1993.
[11] Manning, C.D. & Schutze, H., *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.