Exploring Query Auto-Completion for Data Monetization

Jiahong Chen
Engineering and Applied Math Department
University of Virginia
USA
jc4tf@virginia.edu

Jingming Li
Engineering and Applied Math Department
University of Virginia
USA
jl7ha@virginia.edu

Yue Yin
Engineering and Applied Math Department
University of Virginia
USA
yy7da@virginia.edu

ABSTRACT

Query auto completion (QAC) refers to the functionality of assisting web search users in effectively formulating queries by predicting the following characters or words that are likely to meet users' requirements. It is also one of the first services that the users interact with as they search and form their queries. One major premise of our research has been posted in Query Auto-Completion via Markov Processes [1]. The paper proposed vertical position bias which states that queries that are ranked higher rank tend to attract more clicks regardless of their relevance to the search intent. As a result, if modern search engines want users to click on some set of queries which have advertisements in returned results, it is possible. Based on this premise, our research considers the probability of taking QAC into monetization use.

KEYWORDS

query auto-completion, Trie, query log, click log, monetization

1 INTRODUCTION

Query Auto-completion (QAC) helps users to formulate their queries by displaying a suggestion list (or drop-down list). Each list entry contains a query suggestion based on the input keystrokes. Users can submit a query without selecting query suggestions, or by clicking on any list entry. Typically, a user selects a suggested query if it reflects the user's query intent or out of interests. This fact leads to researches on how to improve QAC retrieved suggestion list, especially on ranking the suggested queries, such as popularity-based QAC [2], time-based QAC [3, 4], context-based QAC [2]. These researches help users get more efficient suggestion lists and improve the likelihood for selecting queries in the lists. In 2014, global users of Yahoo! Search saved more than 50% keystrokes when submitting English queries by selecting suggestions of QAC.

Meanwhile, as an increasing impact of QAC on users' searching behavior, this research paper [1] proposed vertical position bias which states that queries that are ranked higher rank tend to attract more clicks regardless of their relevance to the search intent. As a result, if modern search engines want users to click on some set of queries for monetization use, it is possible. For example, the suggested query at rank position 1 may contain more number of ads than the suggested query at rank position 2. This report proposes an approach to discover whether modern search engines, specifically, Google and Bing, are using QAC as a monetization strategy.

2 RELATED WORK

Query Auto-Completion. The query auto-completion is the process of suggesting queries that complete the prefixe given by the user as the user types each character. Most of existing works in this area concentrate on relevance ranking. The most common approach to QAC is to provide suggested queries from previous user's query log and rank them by their past popularity [5]. Recent QAC models take rare prefixes[9],

freshness in time[10], and personal features[11] etc into consideration. Nevertheless, no published study about whether QAC is used for monetization purpose by modern search engine could be found yet.

Click Models. In the field of document retrieval, we examine the intrinsic correlation between relevance document and query by explaining the position bias of links. This idea of position bias assumption was first raised by Granka et al, claiming that document at higher rank would attract more clicks. Similarly, the ranking of a suggested query also influences the possibility of it being clicked. This paper [6] proposed that a query on higher rank tends to attract more clicks regardless of its relevance to the prefix.

3 METHOD AND EXPERIMENTS

3.1 Method

A user starts a query session by issuing a query to a search engine. At every keystroke, a dynamically updating list of suggested queries is shown to the user. User can click on a given query to search. Given a query q where q is a completed word or phrase, normally Google will return a list gQac = {gq_1, gq_2, …, gq_{10}} where gq_i refers to the i-th suggested query given by Google. Given the same query q, Bing will return a list bQac = {bq_1, bq_2,...bq_3} where bqi refers to the i-th suggested query given by Bing. For the convenience of comparing and analyzing data, the last two queries in every gQac will be ignored, which means every gQac and bQac now has the same length.

To collect data from Google, newly generated queries in gQac one by one serves as an input query to Google and the number of Ad tags on the first result page is recorded respectively. The results are stored as C_gQac = {[gq_1:c_1], [gq_2:c_2], …, [gq_8:c_8]}, where ci refers to the number of Ad tag on the result page of query gqi. Same procedure is implemented on Bing using bQac to get C_bQac = {[bq_1:c_1], [bq_2:c_2], …, [bq_8:c_8]}. If the search engine generates less than eight suggested queries, for example, if a given query q has gQac = {gq_1, gq_2, …, gq_n} where n < 8, [x:0] will be appended to its C_gQac until the length of C_gQac is 8.

3.2 Algorithm

Step 1: input query q and collect eight suggested queries as gQac/bQac;

Step 2: input query in gQac/bQac one by one and count the number of ad tags on search result page to get C_gQac/C_bQac;

Step 3: accumulate the number of ad tags at each index position to get gSum and bSum. Both gSum and bSum includes {s1, s2,...s8} where si refers to the sum of ci in every C_gQac/C_bQac at index i;

Step 4: normalize each si in gSum:

$$Normalized(si) = \alpha(s_i - \frac{1}{8} * \sum_{i=1}^{8} s_{ig}) + \frac{1}{8} * \sum_{i=1}^{8} s_{ib}$$

where sig, sib refers to si in gSum/bSum and is a parameter used to optimize the graph so that the pattern becomes more obvious. We found that $\alpha = 2.5$ would return the best graph for our experiment.

3.3 Test

The input used for the experiment is the top 1000 most popular search query in Google. These queries are used as initial query inputs to collect suggested queries. For analysis purpose, after we get C_gQac from gQac and C_bQac from bQac, we also search each gQac on Bing and each bQac on Google.

4 RESULTS AND DISCUSSION

4.1 QAC Click Bias Assumption

We define two basic assumptions for the QAC problem. One is to address QAC click bias on vertical positions, and the other is to address the click bias regarding to web document retrieval.

• **VERTICAL POSITION BIAS ASSUMPTION:** According to study [6], queries ranked higher in QAC for a certain word receive more clicks. Search engines place queries they value more at higher positions due to this assumption. Therefore, we relate the importance of a query to its position in QAC list.

• **WEB DOCUMENT RETRIEVAL BIAS ASSUMPTION:** Existing studies [7,8] have already allow us to make the assumption that search results displayed on first page receive significantly more clicks than results displayed among the following pages. We therefore relate amount of advertisement to number of links marked Ads on first page of search result.

4.2 Relationship between Advertisement and Ranking of QAC in Google and Bing

By conducting experiments on 1000-Top-Searched-Query dataset, we got Google search result in Table 1 and Bing search result in Table 2.

Position	# Ads
1	449
2	428
3	518
4	486
5	475
6	466
7	479
8	450

Table 1: Accumulated Number of Ads on First Page in Google.

Position	# Ads
1	1593
2	1533
3	1546
4	1481
5	1673
6	1507
7	1477
8	1373

Table 2: Accumulated Number of Ads on First Page in Bing.

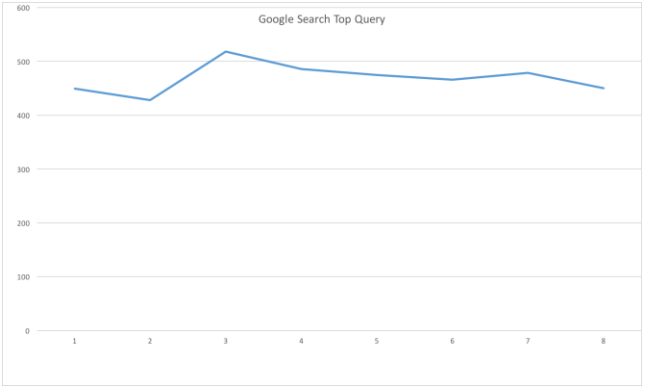

Figure 1: Accumulated Number of Ads vs. position in QAC in Google

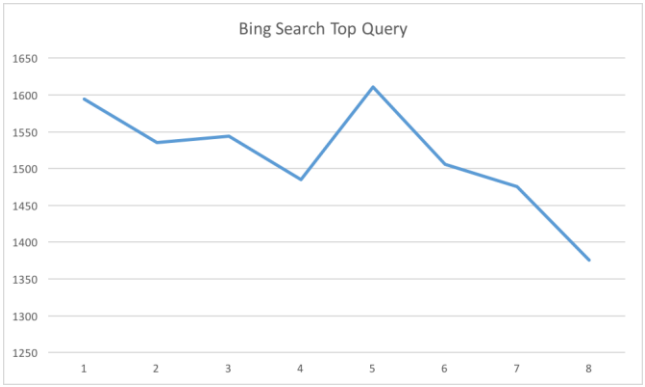

Figure 4: Accumulated Number of Ads vs. position in QAC in Bing

Based on the result, we analyzed the pattern by producing plots. Fig. 1 displays the total amount of advertisements starting around 450, decreasing from position 1 to position 2. The amount of ads then peaks at position 3 and has a decreasing trend. Except for position one and two, the higher the rank, the more advertisements the search result contains. Under the strict control of using same search engine, search method, and field that contain advertisements, one can conclude a correlation between ranking of query auto completion list and monetization of queries in the list. This is a clear indication that monetization feature has been introduced to QAC by Google. However, at position 1 and 2, monetization has not been used or has been normalized to provide more accurate recommendation for users.

Similar to Fig. 1, Fig. 2 exhibits an overall decreasing trend of advertisement amount as QAC ranking position gets lower. The trend can be seen in Fig. 3 with the linear trendline applied. One interesting point is the amount of ads at position 5, which displays as the peak. This is agreed by existing click models showing an increase of click at position five among documents retrieved. Here one can see a clear correlation between QAC ranking and amount

of advertisements. Based on assumptions made in 4.1, the graph demonstrates use of QAC monetization by Bing.

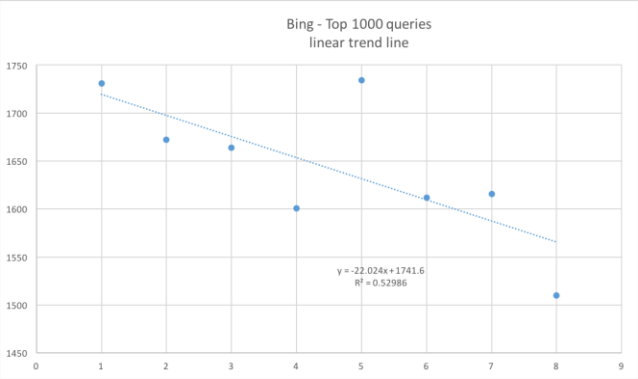

Figure 3: Accumulated Number of Ads vs. position in QAC in Bing with Trendline

4.3 Analysis of Decreasing Trend

To further prove demonstrate the use of monetization in QAC, paired t-test is performed to show the clear decrease of #Ads at lower position. Null hypothesis states that the difference of mean #Ads at different ranking positions is caused by chance. Table 3 rejects the null hypothesis that at position 3 and 6. The truth that there are significantly more advertisements at position 3 than position 6 of Google QAC confirmed conclusion in 4.2. Table 4

Pair	P(T<=t) two-tail
(1st, 2nd)	0.66723614
(3rd, 6th)	0.04892321

Table 3: Paired t-test of Google Result

Pair	P(T<=t) two-tail
(1st, 2nd)	0.26497338
(1st, 4th)	0.09217273

Table 4: **Paired t-test of Bing Result**

4.4 Normalize Google Search Result and Compare to Bing

Note that in general, Google has fewer ads displayed per page than Bing. Normalization by mean mentioned in 3.2 was performed to unify the scale of Google and Bing results. Figure 4 shows the plot after normalization in 3.2. Both Google and Bing show decreasing amount of advertisement at lower rank, while the main difference is that Google has fewer advertisements for the first two QAC recommendations.

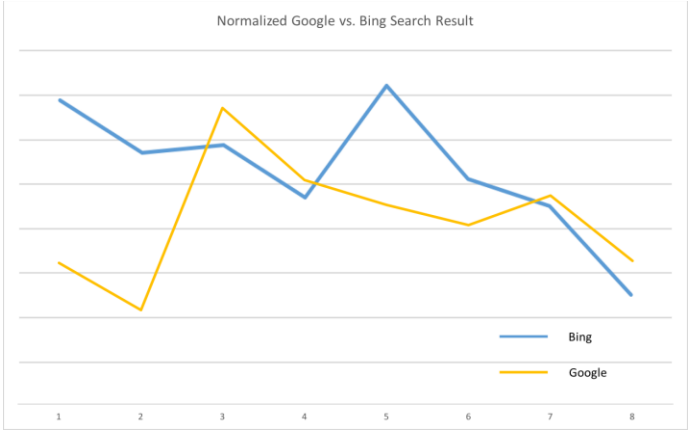

Figure 4: **Normalized #Ads in Google and Original #Ads in Bing**

4.5 Analysis of searching Google QAC in Bing

Further test on Bing's search engine is shown in Figure 5. The figure shows an overall difference between the two lines. Since the same data set is used, searching Google's QAC in Bing results in fewer advertisements at every position than searching Bing's original QAC. One can easily see the difference of using other QAC ranking algorithm form using the search engine's own algorithm. Figure 5 further proves the introduction of monetization in QAC. For Google and Bing, monetization is incorporated differently depending on their different business cooperation.

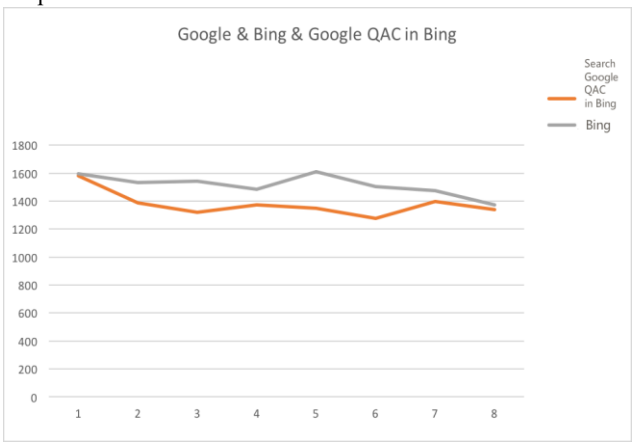

Figure 5: **Search Google QAC in Bing and Search Bing QAC in Bing**

5 CONCLUSIONS

In summary, we have performed an experimental study of the number of ads in each suggested query in QAC suggestion list from both Google and Bing. We collected data and compared the trend in both of the search engines by visualizations. Decreasing number of ads along increasing ranking positions implied that Google and Bing intentionally put more ads in top-ranked suggested queries, and they do use QAC as a strategy for monetization use.

ACKNOWLEDGMENTS

This work was supported by Hongning Wang and his TAs at CS 4501 Information Retrieval class.

REFERENCES

[1] Liangda Li, Hongbo Deng, Anlei Dong, Yi Chang, Hongyuan Zha, and Ricardo Baeza-Yates. 2015. Analyzing User's Sequential Behavior in Query Auto-Completion via Markov Processes. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15). ACM, New York, NY, USA, 123-132. DOI: http://dx.doi.org/10.1145/2766462.2767723

[2] Z. Bar-Yossef and N. Kraus. Context-sensitive query auto-completion. In WWW, 2011.

[3] M. Shokouhi and K. Radinsky. Time-sensitive query auto-completion. In SIGIR, 2012.

[4] S. Whiting and J. M. Jose. Recent and robust query auto-completion. In WWW, 2014.

[5] Bar-Yossef, Ziv, and Naama Kraus. “Context-Sensitive query auto-Completion.” Proceedings of the 20th international conference on World wide web - WWW 11, 2011, doi:10.1145/1963405.1963424.

[6] Li, Yanen, et al. “A two-Dimensional click model for query auto-Completion.” Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval - SIGIR 14, 2014, doi:10.1145/2600428.2609571.

[7] F. Radlinski and T. Joachims. Query chains: learning to rank from implicit feedback. In KDD’05

[8] T.Joachims.Optimizingsearchenginesusingclickthroughdata.In KDD’02

[9]Mitra, Bhaskar, and Nick Craswell. “Query Auto-Completion for Rare Prefixes.” Proceedings of the 24th ACM International on Conference on Information and Knowledge Management - CIKM 15, 2015, doi:10.1145/2806416.2806599.

[10]Shokouhi, Milad, and Kira Radinsky. “Time-Sensitive query auto-Completion.” Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval - SIGIR 12, 2012, doi:10.1145/2348283.2348364.

[11]Shokouhi, Milad. “Learning to personalize query auto-Completion.” Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval - SIGIR 13, 2013, doi:10.1145/2484028.2484076.