Analyzing and Learning from User Interactions for Search Clarification

Hamed Zamani, Bhaskar Mitra, Everest Chen, Gord Lueck, Fernando Diaz, Paul N. Bennett, Nick Craswell, and Susan T. Dumais
Microsoft
{hazamani,bmitra,yuxche,gordonl,fdiaz,pauben,nickcr,sdumais}@microsoft.com

ABSTRACT

Asking clarifying questions in response to search queries has been recognized as a useful technique for revealing the underlying intent of the query. Clarification has applications in retrieval systems with different interfaces, from the traditional web search interfaces to the limited bandwidth interfaces as in speech-only and small screen devices. Generation and evaluation of clarifying questions have been recently studied in the literature. However, user interaction with clarifying questions is relatively unexplored. In this paper, we conduct a comprehensive study by analyzing large-scale user interactions with clarifying questions in a major web search engine. In more detail, we analyze the user engagements received by clarifying questions based on different properties of search queries, clarifying questions, and their candidate answers. We further study click bias in the data, and show that even though reading clarifying questions and candidate answers does not take significant efforts, there still exist some position and presentation biases in the data. We also propose a model for learning representation for clarifying questions based on the user interaction data as implicit feedback. The model is used for re-ranking a number of automatically generated clarifying questions for a given query. Evaluation on both click data and human labeled data demonstrates the high quality of the proposed method.

ACM Reference Format:
Hamed Zamani, Bhaskar Mitra, Everest Chen, Gord Lueck, Fernando Diaz, Paul N. Bennett, Nick Craswell, and Susan T. Dumais. 2020. Analyzing and Learning from User Interactions for Search Clarification. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20), July 25–30, 2020, Virtual Event, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Search queries are oftentimes ambiguous or faceted. The information retrieval (IR) community has made significant efforts to effectively address the user information needs for such queries. A general approach for obtaining more accurate query understanding is to utilize contextual information, such as short- and long-term interaction history [5, 23, 26, 45] and situational context [21, 49]. However, contextual features do not always help the system reveal the user information needs [38]. An alternative solution is diversifying the result list and covering different query intents in the top ranked documents [40]. Although result list diversification has been successfully deployed in modern search engines, it still can be a frustrating experience for the users who have to assess the relevance of multiple documents for satisfying their information needs [2]. On the other hand, in the search scenarios with limited bandwidth user interfaces, presenting a result list containing multiple documents becomes difficult or even impossible [2, 51]. These scenarios include conversational search systems with speech-only or small screen interfaces. To address these shortcomings, (conversational) search engines can *clarify* the user information needs by asking a question, when there is an uncertainty in the query intent.

Although generating plausible clarifying questions for open-domain search queries has been one of a long-standing desires of the IR community [4], it has not been possible until recently. Zamani et al. [51] has recently proposed a neural sequence-to-sequence model that learns to generate clarifying questions in response to open-domain search queries using weak supervision. They showed that clarifying questions can be of significance even for web search engines with the traditional ten blue link interface.

Despite the significant progress in exploring clarification in search [2, 51] and related areas [28, 35, 44], the way users interact with such conversational features of search engines is relatively unknown. Analyzing user interactions with clarifying questions would lead to a better understanding of search clarification, and help researchers realize which queries require clarification and which clarifying questions are preferred by users. Based on this motivation, we conduct a large-scale study of user interactions with clarifying questions for millions of unique queries. This study is based on a relatively new feature, called *clarification pane*, in the Bing search engine that asks a clarifying question in response to some queries. The interface is shown in Figure 1. We analyze user engagements with clarifying questions based on different attributes of the clarification pane. We also study user interactions with clarifying questions for different query properties, such as query length and query type (natural language question or not, ambiguous or faceted, tail or head). We further perform a preliminary study on click bias in clarification panes. Our comprehensive analyses lead to a number of suggestions for improving search clarification.

Following our user interaction analyses, we propose a model for learning representations for clarifying questions together with their candidate answers from user interactions as implicit feedback. Our model consists of two major components: Intents Coverage Encoder and Answer Consistency Encoder. The former encodes the intent coverage of the clarification pane, while the latter encodes the plausibility of the clarification pane, i.e., the coherency of the candidate

SIGIR '20, July 25–30, 2020, Virtual Event, China

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

answers and their consistency with the clarifying questions. Our model is solely designed based on the attention mechanism. We evaluate the model using click data as well as human labeled data. The experiments suggest significant improvements compared to competitive baselines.

In summary, the major contributions of this work include:

- Conducting the first large-scale analysis of user interactions with clarification panes in search. Our study provides suggestions for the future development of algorithms for search clarification.
- Performing preliminary experiments showing different click biases, including both position and presentation biases, in the user interaction data with clarification.
- Proposing a novel neural model, specifically designed for representation learning for clarifying questions. Our model outperforms competitive baselines for the task of clarifying question selection/re-ranking.

2 RELATED WORK

In this section, we review prior work on asking clarifying questions, query suggestion, and click bias estimation.

Asking Clarifying Question. Clarifying questions have been found useful in a number of applications, such as speech recognition [42] as well as dialog systems and chat-bots [6, 13, 33]. In community question answering websites, users often use clarifying questions to better understand the question [7, 35, 36]. Kiesel et al. [22] studied the impact of voice query clarification on user satisfaction. They concluded that users like to be prompted for clarification. Coden et al. [11] studied clarifying questions for entity disambiguation mostly in the form of "did you mean A or B?". Recently, Alianejadi et al. [2] suggested an offline evaluation methodology for asking clarifying questions in conversational systems by proposing the Qulac dataset. The importance of clarification has been also discussed by Radlinski and Craswell [34]. In the TREC HARD Track [3], participants could ask clarifying questions by submitting a form in addition to their runs. Most recently, Zamani et al. [51] proposed models for generating clarifying questions for open-domain search queries. In another study, Zamani and Craswell [50] developed a platform for conversational information seeking that supports mixed-initiative interactions, including clarification. In addition, Hashemi et al. [18] introduced a neural model for representing user interactions with clarifying questions in an open-domain setting. Asking clarifying questions about item attributes has been also explored in the context of conversational recommender systems [43]. For instance, Christakopoulou et al. [10] designed a system for preference elicitation in venue recommendation. Zhang et al. [52] automatically extracted facet-value pairs from product reviews and considered them as questions and answers. In contrast to prior work on search clarification, this work focuses on understanding user interactions with clarifying questions in a real system based on log analysis.

Query Suggestion and Auto-Completion. Query suggestion techniques [14, 30, 39] are used to suggest useful next queries to the users. They have been successfully implemented in search engines. Query suggestion, although related, is fundamentally different from search clarification. The reason is that candidate answers should clarify the intent behind the current search query. While, in query suggestion, the next search query might be a follow up query that is often searched after the query. The clarification examples presented in Figure 1 clearly show the differences. The provided candidate answers are not the expected query suggestions.

Query auto-completion, on the other hand, makes suggestion to complete the current search query [9, 27, 41]. In contrast to query auto-completion, search clarification asks a clarifying question and provides coherent candidate answers which are also consistent with the clarifying question. For more details on the differences between search clarification and query suggestion or auto-completion, we refer the reader to [51].

Click Bias. Click bias in user interactions with search engines has been extensively explored in the literature. It has been shown that users intend to click more on the documents with higher rank positions. There exist different biases, such as position bias [20], presentation bias [48], and trust bias [1]. To address this issue, several user models for simulating user behavior have been proposed, such as the Examination model [37] and the Cascade model [12]. In our clarification interface, the candidate answers are presented horizontally to the users. The answer length is also short, thus multiple answers can be seen at a glance. These unique properties make the click bias in clarification different from document ranking. It is even different from image search, in which the results are shown in a two dimensional grid interface [31, 47].

Table 1: Statistics of the data collected from the user interactions with the clarification pane.

Total impressions	74,617,653
# unique query-clarification pairs	12,344,924
# unique queries	5,553,850
# unique queries with multiple clarification panes	2,302,532
Average number of candidate answers	2.99 ± 1.14

3 ANALYZING USER INTERACTIONS WITH CLARIFICATION

In this section, we study user interactions with clarifying questions in Bing, a major commercial web search engine. We believe these analyses would lead to better understanding of user interactions and expectations from search clarification, which smooths the path towards further development and improvement of algorithms for generating and selecting clarifying questions.

In the following subsections, we first introduce the data we collected from the search logs for our analyses. We further introduce the research questions we study in the analyses and later address these questions one by one.

3.1 Data Collection

The search engine asks clarifying questions from users in response to some ambiguous or faceted queries. The user interface for this feature, which is called the *clarification pane*, is shown in Figure 1. The clarification pane is rendered right below the search bar and on top of the result list. Its location in the result page never changes. The clarification pane consists of a clarifying question and up to five clickable candidate answers. Note that the clarification pane is not triggered for navigational queries. To conduct the analyses, we obtained the clickthrough data for the clarification pane in Bing. For some queries, the data contains multiple clarification panes shown to different set of users. The difference between these clarification

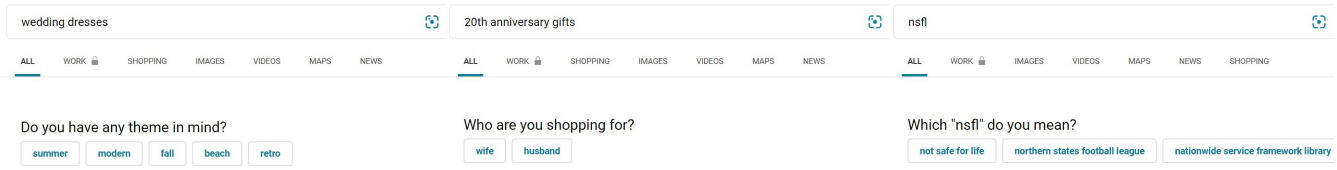
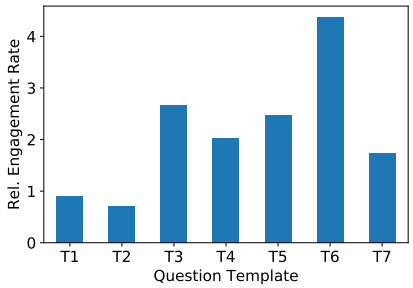


Figure 1: Few examples of clarification in web search.



T1: What (would you like | do you want) to know about _____?
T2: (Which | What) _____ do you mean?
T3: (Which | What) _____ are you looking for?
T4: What (would you like | do you want) to do with _____?
T5: Who are you shopping for?
T6: What are you trying to do?
T7: Do you have _____ in mind?

Figure 2: Relative engagement rate (compared to the average engagement rate) per question template for the most frequent templates in the data.

panes relies on the clarifying question, the candidate answer set, or even the order of candidate answers. For more information on generating clarification panes, we refer the reader to [51].

The collected data consists of over 74.6 million clarification pane impressions (i.e., the number of times the clarification pane was shown to users). The data consists of over 5.5 million unique queries. The average number of candidate answers per clarification pane is equal to 2.99. The statistics of the data is reported in Table 1. Note that we only focus on the query-clarification pairs with at least 10 impressions.

3.2 Research Questions

In the rest of Section 3, we study the following research questions by analyzing the user interaction data described in Section 3.1.

- **RQ1** Which clarifying questions would lead to higher engagements? (Section 3.3).
- **RQ2** For which search queries do users prefer to use clarification? (Section 3.4)
- **RQ3** How is the impact of clarification on search experience? (Section 3.5)

3.3 Characterizing Clarifications with High Engagement Rate

In this subsection, we address RQ1 defined in Section 3.2. To this end, we study the obtained engagement rate (i.e., click rate) by the clarification pane based on different clarification properties, including (1) the clarifying question template, (2) the number of candidate answers, and (3) the conditional click distribution across candidate answers.

Table 2: Relative engagement rate (w.r.t. average engagement) for clarification panes per number of answers.

# Candidate Answers	2	3	4	5
Relative Engagement Rate	0.95	1.05	1.03	1.03

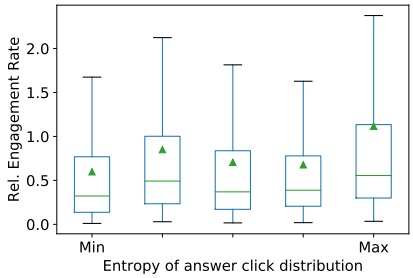


Figure 3: A box plot for the relative engagement rate (compared to the average engagement rate) with respect to the entropy in the conditional answer click distribution. This plot is only computed for clarifications with five options.

3.3.1 Analyzing Clarifying Question Templates. As recently discovered by Zamani et al. [51], most clarification types can be addressed using a set of pre-defined question templates. We identified all question templates used in the data and focused on the most frequent templates. The average engagement rate obtained by each template relative to the overall average engagement rate is presented in Figure 2. The templates are sorted with respect to their reverse frequency in the data. According to the figure, general question templates than can be potentially used for almost all queries, such as "what would you like to know about QUERY?" have the higher frequency in the data, while their engagement is relatively low. On the other hand, more specific question templates,[1] such as "what are you trying to do?", "who are you shopping for?", and "which _____ are you looking for?" lead to much higher engagement rates. The relative difference between the engagement rates received by the templates can be as large as 500% (T2 vs. T6).

3.3.2 Analyzing the Number of Candidate Answers. As mentioned earlier in Section 3.1, the number of candidate answers varies between two and five. Table 2 shows the relative engagement rate per number of candidate answers in the clarification pane. According to the results, the clarification panes with only two candidate answers receive a slightly lower engagement rate. The reason could be that the clarification panes with two candidate answers do not always cover all aspects of the submitted query. The clarifying questions with more than two candidate answers generally receive similar engagement rates with each other. Generally speaking, the number of candidate answers is not a strong indicator of user engagement rate.

[1] By more specific, we mean the questions that cannot be asked for all queries, as opposed to general templates, like T1, that can be asked in response to any query.

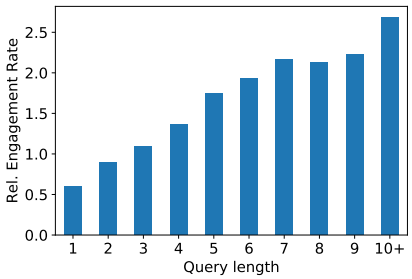


Figure 4: Relative engagement rate (compared to the average engagement rate) per query length.

3.3.3 Analyzing Answer Click Distribution. Figure 3 plots the relative engagement rate received by the clarification pane with respect to the entropy of conditional click distribution on the candidate answers. In case of no observed click for a clarification pane, we assigned equal conditional click probability to all candidate answers. The box plot is computed for five equal-width bins between the minimum and maximum entropy. In this experiments, we only focus on the clarification panes with exactly five candidate answers. The goal of this analysis is to discuss whether higher click entropy (i.e., closer to the uniform click distributions on the candidate answers) would lead to higher engagement rate. According to the plot, the clarification panes with the highest click entropy lead to the highest average and median engagement rate. The second bin from the left also achieves a relatively high average and median engagement rate. This plot shows that the data points in the minimum entropy bin achieve the lowest average and median engagement rates, however, the increase in answer click entropy dose not always lead to higher engagement rate. The reason is that some clarification panes with high engagement rates contain a dominant answer. As an example, for the clarifying question "What version of Windows are you looking for?", we observe over 10 times more clicks on "Windows 10" compared to the other versions. Note that this may change over time. Analyzing the temporal aspect of click distribution and engagement rate is left for future work. In summary the majority of engagement comes for one of two reasons: (1) high ambiguity in the query with many resolutions (i.e., the high click entropy case); (2) ambiguity but where there is a dominant "assumed" intent by users where they only realize the ambiguity after issuing the query (e.g., the mentioned Windows 10 example).

3.4 Characterizing Queries with High Clarification Engagement

We address the second research question (RQ2: For which web search queries, do users prefer to use clarification?) by analyzing the user engagements with the clarification pane based on different query properties, such as query length, query type (natural language questions vs. other queries; ambiguous vs. faceted queries; head vs. torso vs. tail queries), and historical clicks observed for the query.

3.4.1 Analyzing Clarification Engagement Based on Query Length. In the research literature, long queries have often given rise to more challenges in producing quality results. One reason is that longer queries are more likely to be less frequent and among tail queries [17]. We study the engagement rates received by the clarification pane with respect to the query length. The result is shown in Figure 4. Interestingly, as the query length increases, we observe

Table 3: Relative engagement rate (compared to the average engagement rate) per query type.

Query type	Relative engagement rate
Natural language question	1.58
Other queries	0.96
Faceted queries	1.52
Ambiguous queries	0.70
Tail queries	1.01
Torso queries	1.02
Head queries	0.99

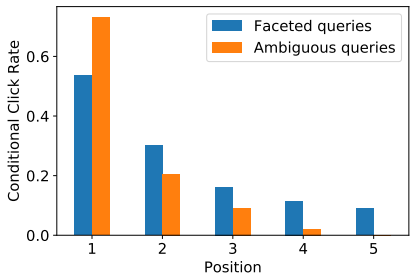


Figure 5: Conditional click rate per position for ambiguous vs. faceted queries for clarifications with five answers.

substantial increase in the average engagement rate. Note that the clarification pane is not shown to the user for navigational queries, thus the data does not contain such queries.

3.4.2 Analyzing Clarification Engagement for Natural Language Questions. According to the first two rows in Table 3, the average engagement rate observed for natural language questions are 64% (relatively) higher than the other queries. Therefore, users who issue natural language questions are more likely to interact with the clarification pane. This observation demonstrates yet another motivation for using clarifying questions in the information seeking systems with natural language user interactions, such as conversational search systems.

3.4.3 Analyzing Clarification Engagement for Ambiguous versus Faceted Queries. Clarifying questions in web search can be useful for revealing the user information needs behind the submitted *ambiguous* or *faceted* queries. In Figure 1, few clarification examples are shown. The third example in the figure (right) shows the clarification pane for an ambiguous query, while the other two are faceted queries. The middle part of Table 3 reports the relative engagement rate received by the clarification pane for ambiguous and faceted queries. The category of each query was automatically identified based on the clarifying question and the candidate answers generated in the clarification pane. According to the figure, the clarification pane for faceted queries are approximately 100% more likely to receive a click compared to the ambiguous queries. We plot the conditional click distribution per position for ambiguous and faceted queries in Figure 5. The graph shows that the gap between the first and the second position for ambiguous queries are substantially higher than the gap for faceted queries. This shows that for ambiguous queries, it is more likely that one query intent dominates the user information needs for the query. In fact, this might be one of the reasons that the clarification pane for ambiguous queries receives less engagement, because it is likely that the

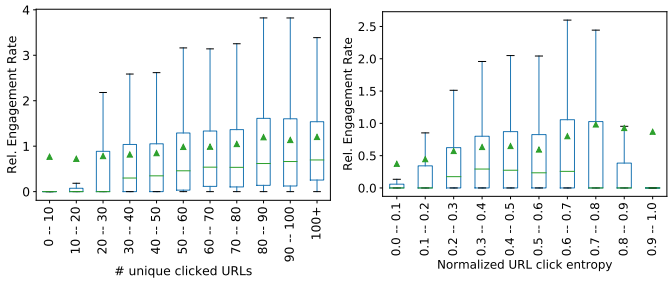


Figure 6: A box plot for the relative engagement rate with respect to (a) the number of unique clicked URLs for the query, and (2) the normalized entropy of click distribution on URLs.

SERP often covers the most dominant query intent in the top position, thus users skip the clarification pane and directly move to the result list.

3.4.4 Analyzing Clarification Engagement for Head, Torso, and Tail Queries. We use the search traffic to identify the query types. The most frequent queries for a third of search traffic was considered as head queries, the second third as torso, and the rest as tail queries. This results in a small number of high frequency head queries and a large number of low frequency tail queries. We further compute the average engagement rate per query types and report the results in the last part of Table 3. According to the results, all query types achieve similar clarification engagement. Note that the data contains the queries that the clarification pane was triggered for, therefore, there should be too many tail queries that the system does not generate a clarifying question for.

3.4.5 Analyzing Clarification Engagement Based on Historical Click Data. We hypothesize that as the number of aspects for the query increases, the necessity for clarification also increases. To study this hypothesis, we measure the number of aspects per query based on the following criteria:

- Using click data on SERP: for each query q in our data, we looked at a historical click logs and counted the number of unique URLs clicked for the query q.
- Since some clicked URLs may be very related and do not represent different aspects, we follow the approach used in [24] and computed the click distribution entropy normalized by the maximum entropy as an indicator of aspect diversity for the query.

The detailed description of click data used in this analysis is presented in Section 5.1.5. The results are plotted in Figure 6 and show that as the number of unique clicked URLs increases the relative engagement rate (both average and median) increases. This is also generally the case when the entropy of click distribution increases. Generally speaking, the unique number of clicked URLs and the click entropy are good indicators of user engagement with clarifying questions.

3.5 Analyzing Clarification Impact and Quality

In Sections 3.3 and 3.4, we analyze user interactions with clarification panes in web search. In the next set of analysis, we study the impact of clarification on search experience (i.e., RQ3 in Section 3.2). Since SERP contains multiple elements, such as the result list, the entity card, and the answer box, one cannot simply compute the satisfying click ratio as a full indicator of search satisfaction. Hassan et al. [19] shows that measuring user satisfaction can go beyond clicks and for example query reformulation can be used as a signal for user satisfaction. Therefore, because there are multiple SERP elements that can satisfy user satisfaction, we instead focus on dissatisfaction. Clicking on the result list with a small dwell time (i.e., unsatisfying clicks) or reformulating a query with a similar query within a time interval that is short enough (such as five minutes) implies dissatisfaction [19]. We measured dissatisfaction for the sessions in which users interact with clarification, and observed 16.6% less dissatisfaction compared to the overall dissatisfaction of the search engine. Note that there are many queries for which the clarification pane is not shown. Therefore, this relative number is *not* a completely representative comparison, however it gives us some idea on the overall impact of clarification on search quality. Since clicking on a candidate answer in clarification leads to a new query and a new SERP, A/B testing for measuring the impact of clarification in search could be also quite challenging here. Some of these challenges have been discussed by Machmouchi and Buscher [25]. A comprehensive study of user satisfaction while interacting with clarifying questions is left for future work.

Table 4: The human labels for the clarification panes.

Label	% Good	% Fair	% Bad
Overall label	6.4%	86.5%	7.1%
Landing page label	89.1%	6.6%	4.3%

We also observe that in 7.30% of the interactions with the clarification pane, users click on multiple candidate answers. This suggests that in many of these cases, the users would like to *explore* different candidate answers provided by the system. In other words, this observation shows that there is a promise in using clarification with candidate answers for *exploratory search*.

Another approach to measure the impact of search clarification is measuring search quality using human annotations. To do so, we sampled 2000 unique queries from the search logs and asked three trained annotators to provide labels for each query-clarification pair. Following [2, 51], we first asked the trained annotators to first skim multiple pages of search results for the query to have a sense on different possible intents of the query. We then asked them to provide the following labels for each clarification pane:

- Overall label: the overall label is given to the whole clarification pane in terms of its usefulness for clarification, comprehensiveness, coverage, understandability, grammar, diversity, and importance order. In summary, they are asked to assign a Good label, if all the mentioned criteria are met. While, the Fair label should be assigned to an acceptable candidate answer set that does not satisfy at least one of the above criteria. Otherwise, the Bad label should be chosen.
- Landing page quality: the search quality of the secondary SERP obtained by clicking on each candidate answer. A secondary SERP is considered as Good, if the answer to all possible information needs behind the the selected answer can be *easily* found in a prominent location in the page (e.g., an answer box on top of the page or the top three documents) and the retrieved information *correctly* satisfies the possible information needs. If the result page is still useful but finding the answer is not easy, the Fair label should be chosen. Otherwise, the landing page is Bad.

A detailed description of each label with multiple examples is provided to the annotators. In some rare cases (less than 2%), there is no agreement between the annotators (i.e., no label with more than 1 voter). In such cases, we dropped the query-clarification pair from the data. The overall Fleiss' kappa inter-annotator agreement is 72.15%, which is considered as good.

The results for human annotations are shown in Table 4. According to the table, the majority of secondary search results (i.e., landing page) after clicking on each individual option are labeled as Good, so the query intent was addressed in a prominent location of the SERP. For the overall label, most annotators tend to choose Fair as the label. Note that Fair still meets some high standards due to the description provided to the annotators. The reason is that they could mostly argue that there is an intent that is not covered by the clarification pane, and thus it should not get a Good label.

4 EXPLORING CLICK BIAS

In the last section, we study the engagement rates received by the clarification pane in web search. In this section, we extend our analysis to the interactions with individual candidate answers. Such analysis would be useful for developing effective models for re-ranking candidate answers or even replacing them. However, implicit feedback could be biased for a number of reasons, such as presentation. Figure 5 shows that for both query types, the conditional click probability decreases by the increase in the candidate answer position. Note that the candidate answers are presented horizontally in the interface and the first position means the far left candidate answer in Figure 1. This observation might be due to the fact that the clarification pane sorts candidate answers based on their popularity and relevance. On the other hand, this could be also due to position and presentation biases in user behaviors. This section provides a preliminary analysis of bias in the click data observed on each candidate answer.

In the experiments designed for this section, we followed the process used by Craswell et al. [12] for studying position bias in web search. In more detail, we created a data set D whose instances are in the form of (q, C, C'), where q is a query while C and C' are two difference clarification panes for q. We make sure that the clarifying question and the candidate answer set in both C and C' are the same. The only different between C and C' is that *two adjacent candidate answers* are swapped. Therefore, as suggested in [12], this data allows us to focus on the click distribution on two adjacent candidate answers where their contents and their relevance do not change, while their positions change. This resulted in 46,573 unique queries and 132,981 data points in our data.

To study click bias in D, we first solely focus on the position. To do so, for each triplet $(q, C, C') \in D$, assume that the candidate answer in position i is swapped with the one in position $i+1$. In other words, $C_i = C'_{i+1}$ and $C_{i+1} = C'_i$, where the subscripts show the position of candidate answer (note that $\forall j \neq i, i+1 : C_j = C'_j$). We then construct the following two-dimensional data points:

$$< \text{click rate for } C_i,\ \text{click rate for } C'_{i+1} >$$
$$< \text{click rate for } C'_i,\ \text{click rate for } C_{i+1} >$$

These pairs show what would be the click rate on the same candidate answer if it ranks higher for only one position. We repeat this process for all the data points in D. The scatter plots for the created data points in a log odds space ($\text{log\_odds}(p) = \log(\frac{p}{1-p})$) are shown in Figure 7. Note that in a perfect scenario, all points should be on the diagonal in the figures. However, this perfect scenario never happens in practice. We also fit a line (i.e., the solid line) to the data points in each scatter plot to better demonstrate the distribution of data points in this space. As shown in the figure, the slope of the line generally gets closer to the diagonal as the number of options increases. The reason is that as the number of options increases, the click bias in the lower positions are far less than the bias in the higher positions and this influences the overall click bias.

Table 5: Percentage of points that would receive higher click rate if moved to a higher position (i.e., % points above the diagonal in Figure 7). Note that the distance from diagonal is visualized by the line fitted on the data in Figure 7.

# candidate answers	$1 \leftrightarrow 2$	$2 \leftrightarrow 3$	$3 \leftrightarrow 4$	$4 \leftrightarrow 5$
2	56.34%			
3	56.17%	57.89%		
4	47.28%	57.63%	55.62%	
5	48.50%	52.32%	53.54%	49.77%

We also compute the percentage of points above the diagonal in each setting. This shows for what percentage of data points, the same answer with a higher positions would attract more clicks. The result is reported in Table 5. Each column in the table shows the position of swapped adjacent answers. The closer the percentage to 50%, the less likely there is a click bias. Moreover, all the percentages are typically expected to be higher than or equal to 50%, which means options with higher ranks (left) are more likely to be clicked. However, our observation in Table 5 is different. As shown in the table, when the number of answers are 4 or 5, the percentage of points above the diagonal is lower than 50% for the $1 \leftrightarrow 2$ setting (this also happens for $4 \leftrightarrow 5$ when the number of candidate answers is 5, but it is close to 50%). The reason for such observation is that position (i.e., rank) is not the only variable that influences click bias. The size of candidate answers also varies as the candidate answer content gets longer (e.g., see Figure 1). Proper visualization of the click bias considering all of these variables is difficult. Therefore, to study the influence of each variable on click distribution, we train a logistic regression for click prediction. This is similar to the technique used by Yue et al. [48] to study click bias based on different result presentations in web search (e.g., the number of bold terms in the snippets). Therefore, for each triplet $(q, C, C') \in D$, the goal is to predict the click rate for the swapped candidate answers in C' given the observation we had from C. We use the following features for the logistic regression model:

- CTR_L: The click rate observed for candidate answer C_i.
- CTR_R: The click rate observed for candidate answer C_{i+1}.
- SIZE_DIFF: The relative size difference between the candidate answers C_i and C_{i+1}. In other words, this feature is equal to $(\text{size}(C_i) - \text{size}(C_{i+1}))/(\text{size}(C_i) + \text{size}(C_{i+1}))$.
- OFFSET: The offset of the candidate answer C_i. For the first candidate answer, the offset is equal to zero.

We train two logistic regressions to predict the following labels:

- L: The click rate for candidate answer C'_i.
- R: The click rate for candidate answer C'_{i+1}.

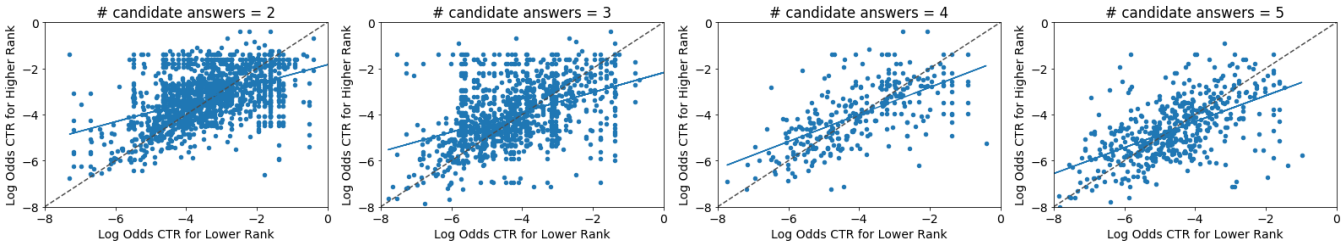


Figure 7: Log odds scatter plot for the click rates of the same candidate answer on the lower position (x axis) and the higher position (y axis) when swapping adjacent candidate answers.

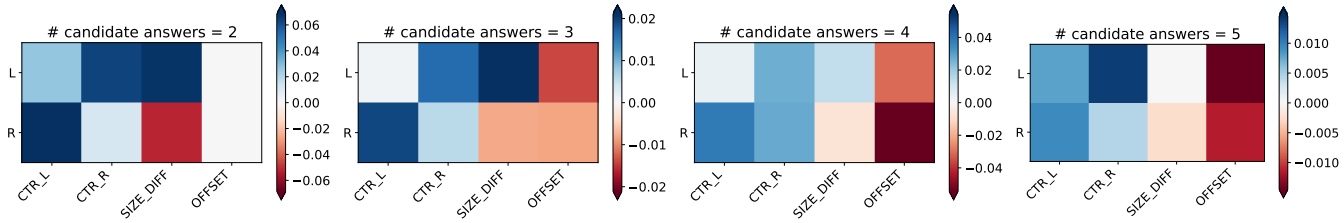


Figure 8: Feature weights learned by logistic regression for predicting click rate when two adjacent candidate answers are swapped. The figure should be viewed in color.

Table 6: Cross entropy for click rate estimation models. Lower cross entropy indicates more accurate click rate estimation.

Model	2 options	3 options	4 options	5 options
Best Possible	0.0216 ± 0.0058	0.0100 ± 0.0040	0.0097 ± 0.0049	0.0053 ± 0.0012
Blind click (relevance independent)	0.1193 ± 0.0294	0.0604 ± 0.0275	0.0561 ± 0.0330	0.0283 ± 0.0064
Baseline (no click bias)	0.1105 ± 0.0264	0.0578 ± 0.0254	0.0539 ± 0.0329	0.0272 ± 0.0064
Examination	0.1084 ± 0.0237	0.0544 ± 0.0186	0.0517 ± 0.0260	0.0275 ± 0.0093
Cascade	0.1063 ± 0.0145	0.0551 ± 0.0174	0.0510 ± 0.0189	0.0273 ± 0.0090
Logistic regression	0.0482 ± 0.0058	0.0336 ± 0.0055	0.0333 ± 0.0064	0.0264 ± 0.0012

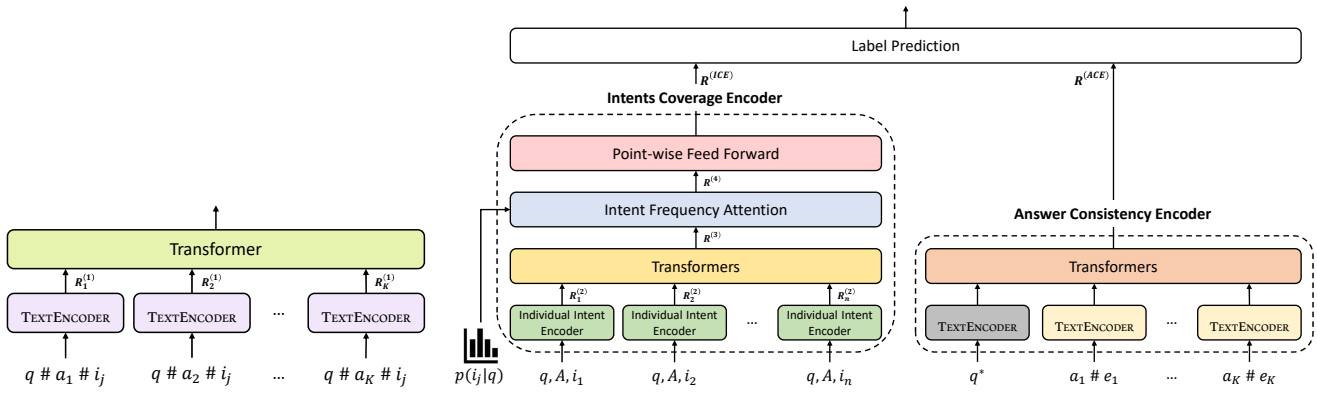
Note that the candidate answer C'_i (or C'_{i+1}) is in position $i+1$ (or i) in C. We perform 10 fold cross-validation for training the logistic regression model. The learned feature weights were consistent across folds. The average weights are shown in Figure 8. In all the plots, CTR_L gets a positive weight for the label R and CTR_R also gets a positive weight for the label L. This shows that the click rate on the same candidate answer in the reverse order is a positive signal for click prediction, which is expected. The weights for two candidate answers shows that the size difference of candidate answers are also very effective in predicting the click bias. As the number of answers increases, the influence of size difference decreases, while the influence of offset increases. The size difference for the label R always gets a negative weight, while this feature gets a positive weight for label L. This is again expected, showing that if we replace the left candidate answer with a larger size answer, the click rate on L would increase, and at the same time the click rate on R would decrease. In other words, the candidate answer size is a strong signal for predicting click rate. The offset has a negative weight for both labels L and R. This suggests that the further the candidate answers from the left, the less likely to observe a click. Note that when the number of candidate answers is two, the offset for all examples is equal to zero and thus it has no effect.

To show that this simple logistic regression predicts the click rate accurately, we compare this model against some simple baselines. The results are reported in Table 6. Following Craswell et al. [12], we use cross entropy between the true and the predicted click rates as the evaluation metric. The results show that a baseline model that assumes there is no click bias has a much higher cross entropy than the best possible cross entropy (i.e., the entropy of the true labels). The Examination model [37] and the Cascade model [12] are user models borrowed from the web search literature. The Examination model assumes each rank has a certain probability of being examined by the user. The Cascade model, on the other hand, assumes that the user views search results from top to bottom, deciding whether to click before moving to the next. Therefore, it also models a skip probability. The assumptions made by both of these models (and many other click models) may not hold in our scenario, where the answers are presented horizontally and their length is small and many of them can be examined by the user at a glance. The results also suggest that these models do not predict the click rate much better than the baseline which assumes there is no click bias. The logistic regression model, however, achieves a much lower cross entropy. Note that the goal of this section is providing some insights into the click bias in the data, and not proposing effective user models for click estimation.

We believe that this preliminary click bias analysis provides some insights into how bias is the user interactions with individual candidate answers. Deeper analyses, for example based on mouse movement and eye-tracking, can shed light on the user click behaviors with clarifying questions and can lead to accurate user models for click estimation and debiasing the data.

5 IMPROVING CLARIFICATION USING USER INTERACTION DATA

A fundamental task in search clarification is re-ranking and selecting the clarifying questions generated by different models under different assumptions. A few clarifying question generation models are presented in [51]. Based on the analyses presented in Section 3,



(a) The Individual Intent Encoder model. **(b) The RLC architecture.**

Figure 9: The neural network architecture for RLC. Same color indicates shared parameters.

we introduce the following features for re-ranking clarification panes in response to a query: (1) question template (a categorical feature), (2) query length, (3) query types (see Table 3), (4) the number of candidate answers, (5) the number of unique clicked URLs, and (6) the URL normalized click entropy. A number of these features are query-specific. To measure how much the clarification pane clarifies different query intents, we can use the Clarification Estimation model presented in [51]. However, some aspects of clarification (e.g., candidate answer coherency) is missing or is not effectively addressed in this feature. In the following, we propose an end to end neural model to fill these gaps. The model is mainly trained based on user interaction data and further fine-tuned using a small set of human labeled data.

Let us first introduce our notation. Let T denote a training set containing triplets of (q, C, L), where q is a unique query, $C = [c_1, c_2, \cdots, c_m]$ is a set of m clarification panes for the query, and $L = [l_1, l_2, \cdots, l_m]$ is the labels associated with the clarification panes. Each clarification pane c_j includes a clarifying question q^* and a list of K candidate answers $A = [a_1, a_2, \cdots, a_K]$, where $K = 5$ in our setting. Additionally, let I_q denote the intent set for the query q with n intents, whose j^{th} element is a pair (i_j, w_j), where denotes an intent (i_j) and its weight (w_j). Note that the query intent set is often unknown to a system, but there exist few approaches for estimating the intent set based on query logs and click data. We later explain how we built the intent set I_q for our experiments (See Section 5.1.5). The goal is to train a representation learning model for each query-clarification pair. This model can be used for selecting or re-ranking clarification panes.

5.1 Representation Learning for Clarification

We design our neural model based on the following assumptions:
Assumption 1. A good clarification pane should clarify different intents of the query, in particular the most frequent intents.
Assumption 2. The candidate answers in a good clarification pane should be coherent and also consistent with the clarifying question.

Assumption 1 is indirectly related to the analysis done in Figure 6, which shows that queries with more unique clicked URLs would lead to higher engagement rates. This shows that covering a wide range of intents is an important factor in clarifying questions, which leads us to the first assumption. Given these assumptions, our model, called RLC,[2] is built based on two major components:

[2]stands for Representation Learning for Clarification.

Table 7: The training and test data used in our experiments.

Data	# training queries	# test queries	# clarifications per query
Click data	137,392	3925	6.2
Labeled data	1848	122	10

Intents Coverage Encoder and *Answers Consistency Encoder*. The architecture of RLC is depicted in Figure 9.

5.1.1 Intents Coverage Encoder. This component learns a high-dimensional vector representing the intent coverage of the candidate answer set. We first create $K \times n$ triplets (q, a_k, i_j) for $1 \le k \le K$ and $1 \le j \le n$. For each of these triplets, we create a sequence `<b> query <s> answer <s> intent <e>` with some boundary tokens and feed the sequence to a text encoder network for obtaining the representation $R^{(1)}_{kj} = \text{TextEncoder}(q, a_k, i_j)$. See Section 5.1.4 for more information on TextEncoder. Next, we would like to see whether each intent is covered by the candidate answer set. Therefore, we concatenate all the representations $R^{(1)}_{kj}$ for all $1 \le k \le K$ and feed the obtained vector to a Transformer encoder, which consists of multiple Transformer layers [46]. The self-attention mechanism in Transformer helps the model learn a representation for the coverage of the j^{th} intent by the answer set. This results in n representations $R^{(2)}_j$, one per query intent.

Different query intents may be related, especially since they are automatically estimated using some algorithms. Therefore, we apply a Transformer Encoder layer on top of all individual intent representations, whose self-attention mechanism would lead to learning accurate representations for related intents. This layer gives us $R^{(3)}_j$ for each intent i_j. In addition, some intents are more common than the others. According to Assumption 1, we expect the model to particularly cover those common intents. Therefore, we use the intent weights as attentions for intent coverage representation. Formally, $R^{(4)}_j = \frac{w_j}{\sum_{j'} w_{j'}} R^{(3)}_j$. This layer is followed by two point-wise feed-forward layers to adjust the representation space and add non-linearity. This component returns the intent coverage encoding $R^{(ICE)}$.

5.1.2 Answers Consistency Encoder. This component focuses on the clarifying question and its answer set. Answer entity types are found useful for generating clarifying questions [51]. Therefore, in this component, we first learn a representation for each candidate

Table 8: Experimental results for re-ranking clarification panes for a query. The superscripts 1/2/3 indicate statistically significant improvements compared to Clarification Estimation/BERT/LambdaMART without RLC, respectively.

Method	Click Data	Labeled Data			Landing Pages Quality		
	Eng. Rate Impr.	nDCG@1	nDCG@3	nDCG@5	%Bad	%Fair	%Good
Clarification Estimation [51]	–	0.8173	0.9356	0.9348	11.68%	13.24%	75.08%
BERT [15]	25.96%^{1}	0.8515^{1}	0.9449	0.9425	10.52%	17.24%	72.24%
LambdaMART w/o RLC	67.27%^{12}	0.9001^{12}	0.9584^{1}	0.9565^{1}	5.21%	19.45%	75.34%
RLC	92.41%^{123}	0.9312^{123}	0.9721^{123}	0.9702^{123}	5.63%	12.33%	82.04%
LambdaMART w/ RLC	**106.18%**^{123}	**0.9410**^{123}	**0.9822**^{123}	**0.9767**^{123}	4.94%	10.21%	84.85%

answer a_k based on the answer text and its entity type (denoted as e_k) if exists, concatenated using a separation token and fed into TextEncoder. We also feed the clarifying question to the TextEncoder. This results in $K+1$ representations. We further apply a Transformer encoder whose self-attention mechanism helps the model identify coherent and consistent answers. In other words, the attention weights from each candidate answer to the others as well as the question help the model observe the similarity of answers and their entity types. The use of entity type would increase generalization and entity similarity better represents the answer coherency. $R^{(ACE)}$ is the output of this component.

5.1.3 Label Prediction. For the label prediction sub-network, we simply concatenate $R^{(ICE)}$ and $R^{(ACE)}$ and feed the obtained vector to a feed-forward network with two layers. The output dimensionality of this component is 1, which indicates the final score for the given query-clarification pair.

5.1.4 TextEncoder. As mentioned above, each major component in the network starts with a TextEncoder. There are several approaches for implementing this component. In this paper, we use BERT [15] – a Transformer-based network pre-trained on a masked language modeling task. BERT has recently led to significant improvements in several NLP and IR tasks [15, 29, 32]. We use BERT-base which consists of 12 layers, 768 representation dimensions, 12 attention heads, and 110M parameters.[3] The BERT parameters are fine-tuned in our end-to-end training. The components with the same color in Figure 9 share parameters. Note that the TextEncoder functions with different colors still share the embedding layer (i.e., the first layer), while their attention weight matrices are different and learned for the specific input type.

5.1.5 The Intent Set I_q. We use two datasets for estimating the intents of each query.[4] The first one is the query reformulation data and the second one is click data on documents. These two datasets were obtained from the Bing query logs, randomly sub-sampled from the data collected in a 2 year period of the EN-US market. The query reformulation data is a set of triplets (q, q', w), where w is the frequency of the $q \longrightarrow q'$ query reformulation in the same session. We use the reformulations in which q' contains q as an estimation for query intent. A similar assumption has been made in [51]. From the click data, we use the title of the clicked URLs as an additional source for estimating query intents. We only kept the query reformulations and clicks with a minimum frequency of 2.

5.2 Training

We train our model using a pair-wise loss function. For two clarification panes for the same query, we get the score from RLC and use the softmax operator to convert the scores to probabilities. We use the binary cross entropy loss function for training, i.e., the label for the clarification pane with higher engagement rate is 1. We further fine tune the model using a small set of human labeled data. We optimize the network parameters using Adam with L_2 weight decay, learning rate warm-up for the first 5000 steps and linear decay of the learning rate. The learning rate was set to 10^5. In the following, we introduce our datasets:

Clarification Click Data: From the data described earlier in Table 1, we kept clarifying questions with at least 10 impressions, and at least two different clarification panes that have different engagement rates, i.e., click rates. We split the data randomly into train and test based on the queries. For more details, see Table 7.

Clarification Labeled Data: We obtained an overall label for clarification and the secondary search result page (landing page) quality labels using the instructions mentioned in Section 3.5. We split the data into train and test sets and no query is shared between the sets. The statistics of this data is also reported in Table 7. Note that in the labeled data we re-rank 10 clarifying questions per query. If the number of labeled clarifying questions are less than 10, we randomly add negative samples with label 0 from the clarifying questions for other queries.

Entity Type Data: For answer entity types, we used an open information extraction toolkit, i.e., Reverb [16], to extract "is a" relations from a large-scale corpus (over 35 petabyte of search snippets). We only kept the relations with the confidence of at least 96%. This results in over 27 millions relations for over 20 millions unique phrases. The data contains over 6 millions entity types.

5.3 Clarification Re-Ranking Results

We first trained the model using 90% of the training set and use the remaining 10% for hyper-parameter tuning of all models, including the baselines. Once the hyper-paraters were selected, we trained the final model on the whole training set and computed the result on the test set. The results for the proposed method and some baselines are reported in Table 8. For the click data, we re-rank the clarification panes and select the first one and report the engagement rate. We finally compute the average engagement rates across queries. The engagement rates are reported relative to the performance of Clarification Estimation [51]. The BERT model uses all the inputs we used in RLC, i.e., the query, the clarification pane and the estimated intents. All of these inputs are concatenated using separation tokens and fed to BERT-base with different segment embeddings. LambdaMART [8] w/o RLC uses all the features described earlier in Section 5 plus the BERT-base output. The results show that the proposed method outperforms all the baselines. According to the paired t-test with Bonferroni correction, the improvements are statistically significant ($p\_value < 0.05$). The best model (i.e., LambdaMART w/ RLC) achieves an nDCG@1 of 0.9410.

[3] The pre-trained models can be found at https://github.com/google-research/bert.
[4] Therefore, there are two Intents Coverage Encoders whose outputs are concatenated.

6 CONCLUSION AND FUTURE DIRECTIONS

In this paper, we provided a thorough analysis of large-scale user interactions with clarifying questions in a major web search engine. We studied the impact of clarification properties on user engagement. We further investigated the queries for which users are more likely to interact with the clarification pane. We also explored the impact of clarification on web search experience, and analyzed presentation bias in user interactions with the clarification panes in web search. Our preliminary analysis on click bias showed that users are often intended to click on candidate answers in higher positions and with larger size. Motivated by our analysis, we proposed a set of features and an end to end neural model for re-ranking clarifying questions for a query. The proposed models outperform the baselines on both click data and human labeled data.

In the future, we intend to study click models for clarification panes to reduce the impact of click bias in ranking candidate answers. We would like to explore user interactions with clarification in devices with limited bandwidth interfaces, such as mobile phones with a focus on speech interactions. Multi-turn clarification is also left for future work.

REFERENCES

[1] Aman Agarwal, Xuanhui Wang, Cheng Li, Michael Bendersky, and Marc Najork. 2019. Addressing Trust Bias for Unbiased Learning-to-Rank. In *WWW '19*. 4–14.

[2] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. 2019. Asking Clarifying Questions in Open-Domain Information-Seeking Conversations. In *SIGIR '19*. 475–484.

[3] James Allan. 2004. HARD Track Overview in TREC 2004: High Accuracy Retrieval from Documents. In *TREC '04*.

[4] Nicholas J. Belkin, Colleen Cool, Adelheit Stein, and Ulrich Thiel. 1995. Cases, scripts, and information-seeking strategies: On the design of interactive information retrieval systems. *Expert Systems with Applications* 9, 3 (1995), 379–395.

[5] Paul N. Bennett, Ryen W. White, Wei Chu, Susan T. Dumais, Peter Bailey, Fedor Borisyuk, and Xiaoyuan Cui. 2012. Modeling the Impact of Short- and Long-term Behavior on Search Personalization. In *SIGIR '12*. 185–194.

[6] Marco De Boni and Suresh Manandhar. 2003. An Analysis of Clarification Dialogue for Question Answering. In *NAACL '03*. 48–55.

[7] Pavel Braslavski, Denis Savenkov, Eugene Agichtein, and Alina Dubatovka. 2017. What Do You Mean Exactly?: Analyzing Clarification Questions in CQA. In *CHIIR '17*. 345–348.

[8] Christopher J. C. Burges. 2010. *From RankNet to LambdaRank to LambdaMART: An Overview*. Technical Report. Microsoft Research.

[9] Fei Cai and Maarten de Rijke. 2016. *A Survey of Query Auto Completion in Information Retrieval*. Now Publishers Inc., Hanover, MA, USA.

[10] Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016. Towards Conversational Recommender Systems. In *KDD '16*. 815–824.

[11] Anni Coden, Daniel Gruhl, Neal Lewis, and Pablo N. Mendes. 2015. Did you mean A or B? Supporting Clarification Dialog for Entity Disambiguation. In *SumPre '15*.

[12] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An Experimental Comparison of Click Position-Bias Models. In *WSDM '08*. 87–94.

[13] Marco De Boni and Suresh Manandhar. 2005. Implementing Clarification Dialogues in Open Domain Question Answering. *Nat. Lang. Eng.* 11, 4 (2005), 343–361.

[14] Mostafa Dehghani, Sascha Rothe, Enrique Alfonseca, and Pascal Fleury. 2017. Learning to Attend, Copy, and Generate for Session-Based Query Suggestion. In *CIKM '17*. 1747–1756.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL '19*. Minneapolis, Minnesota, 4171–4186.

[16] Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying Relations for Open Information Extraction. In *EMNLP '11*. 1535–1545.

[17] Manish Gupta and Michael Bendersky. 2015. Information Retrieval with Verbose Queries. *Foundations and TrendsÂő in Information Retrieval* 9, 3-4 (2015), 209–354.

[18] Helia Hashemi, Hamed Zamani, and W. Bruce Croft. 2020. Multi-Source Transformers: Leveraging Multiple Information Sources for Representation Learning in Conversational Search. In *SIGIR '20*.

[19] Ahmed Hassan, Xiaolin Shi, Nick Craswell, and Bill Ramsey. 2013. Beyond clicks: query reformulation as a predictor of search satisfaction. In *CIKM '13*. 2019–2028.

[20] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. 2005. Accurately Interpreting Clickthrough Data as Implicit Feedback. In *SIGIR '05*. 154–161.

[21] Eugene Kharitonov and Pavel Serdyukov. 2012. Demographic Context in Web Search Re-ranking. In *CIKM '12*. 2555–2558.

[22] Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen. 2018. Toward Voice Query Clarification. In *SIGIR '18*. 1257–1260.

[23] Weize Kong, Rui Li, Jie Luo, Aston Zhang, Yi Chang, and James Allan. 2015. Predicting Search Intent Based on Pre-Search Context. In *SIGIR '15*. 503–512.

[24] Danai Koutra, Paul N. Bennett, and Eric Horvitz. 2015. Events and Controversies: Influences of a Shocking News Event on Information Seeking. In *WWW '15*. 614–624.

[25] Widad Machmouchi and Georg Buscher. 2016. Principles for the Design of Online A/B Metrics. In *SIGIR '16*. 589–590.

[26] Nicolaas Matthijs and Filip Radlinski. 2011. Personalizing Web Search Using Long Term Browsing History. In *WSDM '11*. 25–34.

[27] Bhaskar Mitra, Milad Shokouhi, Filip Radlinski, and Katja Hofmann. 2014. On User Interactions with Query Auto-Completion. In *SIGIR '14*. 1055–1058.

[28] Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. 2016. Generating Natural Questions About an Image. In *ACL '16*. Berlin, Germany, 1802–1813.

[29] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. (2019). arXiv:cs.IR/1901.04085

[30] Umut Ozertem, Olivier Chapelle, Pinar Donmez, and Emre Velipasaoglu. 2012. Learning to Suggest: A Machine Learning Framework for Ranking Query Suggestions. In *SIGIR '12*. 25–34.

[31] Neil O'Hare, Paloma de Juan, Rossano Schifanella, Yunlong He, Dawei Yin, and Yi Chang. 2016. Leveraging User Interaction Signals for Web Image Search. In *SIGIR '16*. 559–568.

[32] Harshith Padigela, Hamed Zamani, and W. Bruce Croft. 2019. Investigating the Successes and Failures of BERT for Passage Re-Ranking. (2019). arXiv:cs.IR/1905.01758

[33] Luis Quintano and Irene Pimenta Rodrigues. 2008. Question/Answering Clarification Dialogues. In *MICAI '08*. 155–164.

[34] Filip Radlinski and Nick Craswell. 2017. A Theoretical Framework for Conversational Search. In *CHIIR '17*. 117–126.

[35] Sudha Rao and Hal Daumé III. 2018. Learning to Ask Good Questions: Ranking Clarification Questions using Neural Expected Value of Perfect Information. In *ACL '18*. Melbourne, Australia, 2737–2746.

[36] Sudha Rao and Hal Daumé III. 2019. Answer-based Adversarial Training for Generating Clarification Questions. In *NAACL '19*. Minneapolis, Minnesota.

[37] Matthew Richardson, Ewa Dominowska, and Robert Ragno. 2007. Predicting Clicks: Estimating the Click-through Rate for New Ads. In *WWW '07*. 521–530.

[38] Mark Sanderson. 2008. Ambiguous Queries: Test Collections Need More Sense. In *SIGIR '08*. 499–506.

[39] Rodrygo L. Santos, Craig Macdonald, and Iadh Ounis. 2013. Learning to Rank Query Suggestions for Adhoc and Diversity Search. *Inf. Retr.* 16, 4 (2013), 429–451.

[40] Rodrygo L. T. Santos, Craig Macdonald, and Iadh Ounis. 2015. Search Result Diversification. *Found. Trends Inf. Retr.* 9, 1 (2015), 1–90.

[41] Milad Shokouhi. 2013. Learning to Personalize Query Auto-Completion. In *SIGIR '13*. 103–112.

[42] Svetlana Stoyanchev, Alex Liu, and Julia Hirschberg. 2014. Towards Natural Clarification Questions in Dialogue Systems. In *AISB '14*, Vol. 20.

[43] Yueming Sun and Yi Zhang. 2018. Conversational Recommender System. In *SIGIR '18*. 235–244.

[44] Jan Trienes and Krisztian Balog. 2019. Identifying Unclear Questions in Community Question Answering Websites. In *ECIR '19*. 276–289.

[45] Yury Ustinovskiy and Pavel Serdyukov. 2013. Personalization of web-search using short-term browsing context. In *CIKM '13*. 1979–1988.

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *NeurIPS '17*. 6000–6010.

[47] Xiaohui Xie, Jiaxin Mao, Maarten de Rijke, Ruizhe Zhang, Min Zhang, and Shaoping Ma. 2018. Constructing an Interaction Behavior Model for Web Image Search. In *SIGIR '18*. 425–434.

[48] Yisong Yue, Rajan Patel, and Hein Roehrig. 2010. Beyond Position Bias: Examining Result Attractiveness as a Source of Presentation Bias in Clickthrough Data. In *WWW '10*. 1011–1018.

[49] Hamed Zamani, Michael Bendersky, Xuanhui Wang, and Mingyang Zhang. 2017. Situational Context for Ranking in Personal Search. In *WWW '17*. 1531–1540.

[50] Hamed Zamani and Nick Craswell. 2020. Macaw: An Extensible Conversational Information Seeking Platform. In *SIGIR '20*.

[51] Hamed Zamani, Susan T. Dumais, Nick Craswell, Paul N. Bennett, and Gord Lueck. 2020. Generating Clarifying Questions for Information Retrieval. In *WWW '20*. 418–428.

[52] Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W. Bruce Croft. 2018. Towards Conversational Search and Recommendation: System Ask, User Respond. In *CIKM '18*. 177–186.