Ghosting: Contextualized Query Auto-Completion on Amazon Search

Lakshmi Ramachandran and Uma Murthy
Amazon Search
Palo Alto, United States of America
lramach@amazon.com,umamurth@amazon.com

ABSTRACT

Query auto-completion presents a ranked list of search queries as suggestions for a customer-entered prefix. Ghosting is the process of auto-completing a search recommendation by highlighting the suggested text inline i.e., within the search box. We present a behavioral recommendation model that uses customer search context to ghost on high confidence queries. We tested ghosting on over 140 million search sessions. Session-context ghosting increased the acceptance of offered suggestions by 6.18%, reduced misspelled searches by 4.42% and improved net sales by 0.14%.

KEYWORDS

query auto-completion, session context, retail search

ACM Reference Format:
Lakshmi Ramachandran and Uma Murthy. 2019. Ghosting: Contextualized Query Auto-Completion on Amazon Search. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19), July 21–25, 2019, Paris, France.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3331184.3331432

1 INTRODUCTION

Query auto-completion (QAC) in retail search provides a ranked list of suggestions based on a user-typed prefix to help customers formulate queries and find products that they desire to purchase. QAC aims to improve the query formulation experience for users leading to a better overall search experience [1].

Ghosting is the combined process of: (1) identifying a high confidence query recommendation, (2) auto-completing the query inline, i.e., within the search box, and (3) highlighting the suggested text (part of the query that is not the prefix) to make it stand out to the customer [5]. Figure 1 shows the default experience and QAC with ghosting for the prefix "wireless bl".
We study the impact of ghosting on the following:

- *Reduced likelihood of misspelled queries:* Fewer misspellings during query formulation can improve the accuracy of search results. We show that ghosting reduces misspelled searches by auto-completing queries within the search box before customers can misspell them. Misspelled searches are one of the leading contributors to poor search results [2]. The onus of handling misspellings falls to downstream spelling correction and ranking services, which may not succeed in correcting the query, yet resulting in increase in latencies. Response delays of a few milliseconds could worsen customer experience [3].
- *Reduced effort in query formulation:* Auto-completing by highlighting the suggested text in the search box allows a customer to easily identify and (a) accept the offered suggestion by hitting enter or clicking on the magnifying glass, or (b) reject it by hitting back-space or by continuing to type more characters [5]. Ghosting helps reduce the number of keystrokes required to formulate a query.
- *Increased query acceptance:* Ghosting helps improve the chance of offered suggestions being accepted by customers. Offered suggestions are optimized for past search impact and are therefore better suited to lead customers to products they seek to purchase.

Figure 1: Default QAC experience (left) and QAC with ghosting (right) for prefix "wireless bl"

2 BEHAVIORAL GHOSTING WITH CONTEXT

In ghosting, one suggestion is recommended to the customer by auto-completing it inline and highlighting the suggested text. Let p be the prefix typed in by the customer, and $C = \{q_1, q_2 \cdots q_n\}$ be the set of candidates offered as suggestions for p. The model aims to identify Q, the most relevant candidate to ghost on. It uses search context to ensure that only queries that are relevant to the current shopping session are ghosted on.

Behavioral Relevance: Optimizing for query behavior ensures that we recommend queries that have successfully led customers to the products that they desired to purchase. Historic query behavior data can be used to determine a query's performance in the future. Q is identified as follows:

$$Q \leftarrow \arg\max_{q_i \in C} behavior(q_i|p) \quad (1)$$

Session Context: The model uses a constraint on the lexical similarity between Q and the last query in the current session r to ensure contextual relevance. Queries Q and r are vectorized as v_Q and v_r and contain tri-character, uni- and bi-grams extracted from

SIGIR '19, July 21–25, 2019, Paris, France

ACM ISBN 978-1-4503-6172-9/19/07.
https://doi.org/10.1145/3331184.3331432

Table 1: A/B test results show percentage improvement of the test group over control. The best model has a larger positive impact on acceptance, cart-add rate and net sales and larger negative impact on spell-correction rate and average prefix length. Statistically significant results are marked with "*" (t-test, p-value < 0.05). The session-context model shows an improvement on all metrics.

Ghosting model	acceptance	spell-correction rate	average prefix length	cart-add rate	net sales
Frequency	**+9.01***	**-9.52***	**-7.51***	-0.03	-0.15
Session context	+6.18*	-4.42*	**-7.12***	**+0.04***	**+0.14***

the respective strings. We compute cosine similarity between the vectorized queries and the decision to ghost (or not) is made based on:

$$D = \begin{cases} 1, & cos(Q,r) = \frac{(v_Q \cdot v_r)}{||v_Q||*||v_r||} \geq h \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

h is the threshold used to ensure that ghosting happens only on queries that are lexically close to r. h is tuned to ensure high precision, since auto-completing an undesirable query may result in a poor customer experience.

3 EVALUATION

We ran A/B tests in production on a subset of Amazon's customers who use the QAC feature. Session-context based ghosting was run for three weeks on over 140 million search sessions. Users were randomly split into two groups: *control* and *test*. The control group was offered the default auto-complete experience, i.e., a ranked list of relevant suggestions, driven by the behavioral model. The test group was presented with the same auto-complete experience along with ghosting on the top, most relevant suggestion, if the condition was met.

We compare the session-context model with a frequency-based approach, which includes a ghosting condition on query frequency. As in Equation 2 the threshold for frequency was tuned to ensure precision.

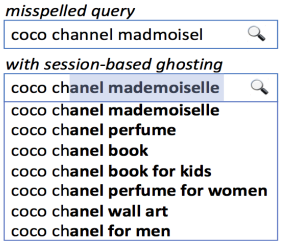

Figure 2: Ghosting on "coco chanel mademoiselle" for prefix "coco ch" due to lexical similarity with previous misspelled search query.

Ghosting helps increase the acceptance of offered suggestions and reduces the spelling-correction rate (results in Table 1). This validates our hypothesis that ghosting helps customers search with fewer misspelled queries thus reducing the need for spelling correction. From the search logs we noticed that a customer searching for a Chanel perfume misspelled the query as "coco channell madmoisel". During their second search attempt, the session-context model identified a lexical similarity between the recently misspelled query and the correct query. This resulted in ghosting on the correctly spelled query, thus helping the customer discover the right query to search on (Figure 2).

Both ghosting approaches show a drop in the average prefix length, i.e., the mean number of characters typed during search. Clicks on search results increased by 0.044% for contextual ghosting. The model also had a positive impact on cart-add rate and net sales. While ghosting with frequency helped improve acceptance of offered suggestions and decreased the rate of misspelled searches, it did not have a positive impact on business metrics. The frequency model likely accrued a lot of acceptances due to its novelty effect [4].

4 CONCLUDING REMARKS

Ghosting helps improve the quality of searches by pre-empting misspellings and helping customers get to desired products with fewer keystrokes. The proposed approach uses query behavior along with the current shopping session to determine whether to ghost. Our work demonstrates the overall impact of this feature on Amazon search and has been tested on live customer traffic.

In this approach we ghost on queries that are relevant to the most recent search. However, customer intent may evolve during the course of a session. For instance, in one session a customer searched for "running shoes for men" and then switched to searching for "dress shirts for men". Auto-completing inline on a query like "dress shoes for men" may not be useful to a customer looking to purchase shirts. We plan to tune the model to identify shifts in search intent in real-time, to determine what aspects of session context would be most meaningful.

5 ACKNOWLEDGMENTS

We would like to thank members (past and present) of the auto-complete and relevance teams at Amazon Search, including Anoop, Saj, Cathy, Hoshun, Andy, Rohit, Daria, Erick, Anup, François, Paul, Micah, Parth and Priya for their support with the testing of the feature and valuable feedback on the paper.

REFERENCES

[1] Ziv Bar-Yossef and Naama Kraus. 2011. Context-sensitive Query Auto-completion. In *Proceedings of the 20th International Conference on World Wide Web (WWW '11)*. ACM, New York, NY, USA, 107–116. https://doi.org/10.1145/1963405.1963424

[2] Qing Chen, Mu Li, and Ming Zhou. 2007. Improving Query Spelling Correction Using Web Search Results. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Association for Computational Linguistics, Prague, Czech Republic, 181–189.

[3] Tobias Flach, Nandita Dukkipati, Andreas Terzis, Barath Raghavan, Neal Cardwell, Yuchung Cheng, Ankur Jain, Shuai Hao, Ethan Katz-Bassett, and Ramesh Govindan. 2013. Reducing Web Latency: The Virtue of Gentle Aggression. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM (SIGCOMM '13)*. ACM, New York, NY, USA, 159–170. https://doi.org/10.1145/2486001.2486014

[4] Georgi Georgiev. 2018. Representative samples and generalizability of A/B testing results. http://blog.analytics-toolkit.com/2018/representative-samples-generalizability-a-b-testing-results/.

[5] Dan Marantz. 2013. *A look at autosuggest https://blogs.bing.com/search/2013/02/20/a-look-at-autosuggest.*