DiAL : Diversity Aware Listwise Ranking for Query Auto-Complete

Sonali Singh
Amazon
ssonl@amazon.com

Sachin Farfade
Amazon
sfarfade@amazon.com

Prakash Mandayam Comar
Amazon
prakasc@amazon.com

Abstract

Query Auto-Complete (QAC) is an essential search feature that suggests users with a list of potential search keyword completions as they type, enabling them to complete their queries faster. While the QAC systems in eCommerce stores generally use the Learning to Rank (LTR) approach optimized based on customer feedback, it struggles to provide diverse suggestions, leading to repetitive queries and limited navigational suggestions related to product categories, attributes, and brands. This paper proposes a novel DiAL framework that explicitly optimizes for diversity alongside customer feedback signals. It achieves this by leveraging a smooth approximation of the diversity-based metric (αNDCG) as a listwise loss function and modifying it to balance relevance and diversity. The proposed approach yield an improvement of 8.5% in mean reciprocal rank (MRR) and 22.8% in αNDCG compared to the pairwise ranking approach on an eCommerce dataset, while meeting the ultra-low latency constraints of real time QAC systems. In an online experiment, the diversity-aware listwise QAC model resulted in a 0.48% lift in revenue. Furthermore, we replicated the proposed approach on a publicly available search log, demonstrating improvements in both diversity and relevance of the suggested queries.

1 Introduction

Query Auto-Complete is a valuable tool in eCommerce that helps customers articulate their query by suggesting relevant completions saving time as well as improving overall search relevance. The QAC problem is usually formulated as a two-step process of matching and ranking. Matching entails retrieving the list of most popular completions (MPC) (Bar-Yossef and Kraus, 2011) based on the characters entered by the user in the search box (or prefix). This is followed by re-ranking of the retrieved keywords by using LTR to finally select top ranked keywords to be displayed to the end user (Cai et al., 2016a). A simple and effective solution to QAC is to suggest the popular queries for a given prefix that reflect the customer choice. However, most popular suggestions have a lot of redundancies retrieving similar search results, thus wasting a precious opportunity to shape the customer search experience. The standard inference flow for QAC is presented in Fig. 1, demonstrating outcomes obtained from an LTR model focused exclusively on relevance, as well as outcomes when the LTR model is configured to concurrently optimize for both relevance and diversity. This redundancy in suggestions can be attributed to two reasons: 1) using the observed click rate as a label for training the ML model causes popular queries to be shown at the top which accumulates more clicks, creating a feedback loop 2) choosing top K queries by scoring each query individually for a given prefix, without considering the context of other queries. To mitigate this issue, it is crucial to diversify the QAC suggestions, similar to the approach taken in web search and retrieval, where researchers have utilized various diversity-based evaluation metrics such as ERR-IA (Chapelle et al.,

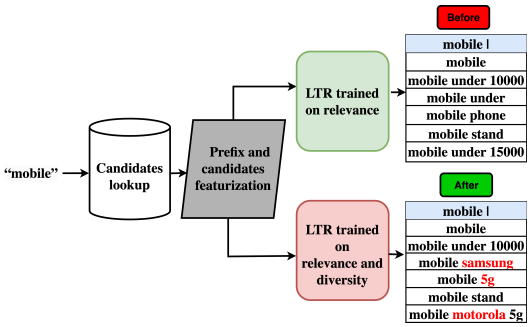

Figure 1: QAC inference flow for the prefix 'mobile' employing two LTR models: 'Before' showing results when trained solely on relevance, and 'After' presenting results after implementing diversification in India market.

2009), αNDCG (Clarke et al., 2008), and greedy optimization methods like Maximal Marginal Relevance (MMR) (Carbonell and Goldstein, 1998). In QAC, we define diversity as the maximum number of distinct topics within the candidate query suggestions presented to the user. The determination of topics depends on the QAC domain and can be tailored to specific business needs. For instance, in the eCommerce domain, we establish topics for QAC suggestions by considering the navigational usefulness of a query. We consider a query to be navigational if it contains product attribute tokens (words) that help narrow down the search results.

Learning to Rank (LTR) is a widely adopted approach for modeling QAC recommendations, typically implemented through pairwise ranking techniques (Fiorini and Lu, 2018; Park and Chiba, 2017; Cai and de Rijke, 2016). In these methods, given a pair of suggestions, the model learns to assign a higher score to the more frequently clicked suggestion (query) compared to the less clicked one. Addressing diversity in QAC suggestions is commonly handled as a post-processing step. First, suggestions relevant to the prefix (as reflected by the ML model score) are selected, and then post-processed to obtain diverse suggestions concerning the navigational topic (Cai et al., 2016b; Slivkins et al., 2010; Feng et al., 2018). However, this approach is disadvantageous as the trade-off between relevance and diversity is determined by heuristics, involving approximations to resolve ties. Furthermore, these greedy selection techniques involving one-by-one comparisons often fail to meet real-time diversity requirements for ranking.

To address existing limitations, we propose DiAL, a listwise ranking method with a tailored scoring function to simultaneously optimize relevance and diversity. Notably, we employ a smooth version of the diversity-based metric (αNDCG) as the loss function, where rank is approximated using scores of queries in the list. We modify this loss to suit QAC constraints, balancing relevance and diversity. Uniquely, we propose a novel diversifying strategy for QAC by mining navigational entities and further utilizing these entities with hierarchical intents in the loss. The overall score-and-sort strategy with a diversity-aware loss, deployable under real-time QAC constraints, has not been studied before. Therefore, we list the main contributions of our work below:

-We introduce a listwise ranking approach with a modified diversity-aware loss function to generate diverse and relevant QAC suggestions in real-time for eCommerce applications.
- We identify and incorporate different intents and graded relevances specific to eCommerce QAC within the listwise loss function.
- Through offline and online evaluations on eCommerce and public search log data, our listwise diversity-aware ranking approach outperformed pairwise baselines (Yuan and Kuang, 2021; Singh et al., 2023), improving both relevance and diversity in QAC recommendations.
To the best of our knowledge, this is the first effort to diversify QAC using a direct score-and-sort approach, emphasizing the novelty of our work.

2 Related Work

The study of diversification in QAC has not been extensively explored. (Cai et al., 2016b) conducted seminal research on diversifying QAC through a greedy query selection approach, suggesting the next query based on query popularity, aspects of the query already in the list, and previous search sessions. Subtopics or aspects are extracted from clicked document URLs for a given query. (Singh et al., 2023) proposed improving quality in QAC using multi-objective ranking by boosting navigational queries using pairwise ranking. While their approach improves the ranking of navigational queries over low-quality or non-navigational queries, it does not explicitly diversify topics or subtopics within navigational queries. Our work draws inspiration from the related area of diversifying web search results, exploring two paths based on whether subtopics are already known (explicit diversification) or not (implicit diversification). For explicit diversification, studies like (Santos et al., 2015; Dang and Croft, 2013; Hu et al., 2015; Sarwar et al., 2020) have been conducted. For implicit diversification, researchers such as (Carbonell and Goldstein, 1998; Sanner et al., 2011; Raiber and Kurland, 2013; Yu et al., 2018; Yan et al., 2021; Yu, 2022) have made contributions. (Yan et al., 2021; Yu, 2022) employed distributed embeddings to uncover latent subtopics and used an approximate diversification metric as a loss to enhance search diversity. Jointly training all queries in the list is critical for training a diversity-aware loss. As such, we adapt the listwise LTR framework (Cao et al., 2007) to score the queries and incorporate a query interaction layer similar to the Document Interaction Network (DIN) (Pasumarthi et al.,

2020) to produce higher-order features for queries in the list. (Qin et al., 2021) provided essential benchmarks and investigated various architectures and loss functions for LTR. We show how listwise LTR with DIN is superior to pairwise LTR with feed-forward layers (Yuan and Kuang, 2021) for modeling click-based relevance in QAC. Following (Bruch et al., 2019b), who suggested using the optimization metric as a loss function for similar or better results over standard LTR loss functions, we train a diversification-aware loss that performs direct metric optimization using a smooth variation of αNDCG.

3 Diversified Auto-Complete

We present the diversity-aware listwise ranking for Auto-Complete (DiAL) framework that models diversity alongside relevance in QAC. DiAL applies listwise ranking with a diversity-aware loss, detailed in this section.

3.1 Diversity aware listwise loss

An approach to defining a loss function in LTR is to directly approximate the evaluation metric, such as NDCG (Normalized Discounted Cumulative Gain), as the loss function, resulting in improved performance on the metric of interest. For diversified ranking, αNDCG is an important metric to evaluate diversity. However, it is not differentiable, and techniques to approximate it have been proposed in several works. The αNDCG metric is defined as follows: Let k be the total number of intents or topics for which the diversity of a list of n ranked keywords associated with prefix p needs to be computed. Each keyword can cover 0 to k intents. Let y_{ij} be keyword-intent labels, which will be 1 if the i^{th} keyword in the ranked list contains the j^{th} intent and 0 otherwise. Let r_i be the rank of the i^{th} item in the list. Then, αDCG is given as:

$$\alpha\text{DCG} = \sum_{i=1}^{n} \sum_{j=1}^{k} \frac{y_{ij}(1-\alpha)^{w_{ji}}}{\log_2(1+r_i)} \quad (1)$$

Here, α is a parameter for penalizing redundancy of intents, and $w_{ji} = \sum_{m:r_m<r_i} y_{mj}$ indicates how many times the j^{th} intent was covered in all keywords ranked above the i^{th} keyword. αNDCG is a normalized version of αDCG, and its approximate differentiable version is used as the loss adopting the approach in (Yan et al., 2021) explained in Appendix A.1.

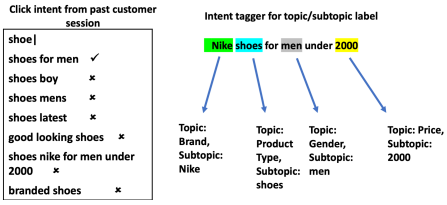

Figure 2: Example to obtain label for click based intent from past search data on the left and example to mine topic/subtopic labels for a query using intent tagger tool on the right.

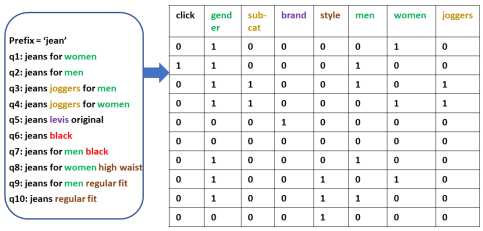

Figure 3: On the left, there is a list of prefix queries, and on the right, there is the corresponding query relevance matrix used to evaluate diversity loss and performance in the context of eCommerce data.

3.2 Intent for Auto-Complete Diversity

Utilizing historical clicked suggestions derived from anonymized search logs of an eCommerce platform and extracting 'navigational' utility-based intents from keywords, we categorize intent into 30 topics and subtopics:

Click intent: The primary intent derived from user-anonymized session logs, where the selected/clicked keyword for a prefix is labeled 1, and the rejected keywords are labeled 0 (Fig. 2).

Non-superfluous intent: Queries must be precise without redundant words like 'best', 'stylish', or 'good-looking'. We assign a label of 1 to denote queries with no redundant words, identified by matching with a predefined list of redundancies.

Presence of topics: The intent labels are procured by the presence or absence of the top topics present in the query, such as 'product type', 'gender', and 'age' for shoes, or 'processor type' and 'screen size' for laptops. We use an internal intent tagger tool to identify topic boundaries (Fig. 2) in each query and pick the top 10 most frequent topics from the prefix keywords list.

Presence of subtopics: Subtopic labels are assigned within the topic, such as gender type ('men' or 'women') and screen size ('14 inch' or '15 inch'). We choose the 18 most frequent subtopics from each prefix keywords list.

Cutoff of topics and subtopics was determined by

measuring the frequency of their occurrence, beyond which they were considered unpopular for diversification purposes. We construct a prefix query relevance matrix (Fig. 3) with these 30 intent labels for each prefix and keywords list pair. While these binary intent labels can be directly plugged into the loss function, we add two additional parameters in equation 1 to discount binary relevance and account for varying levels of importance of various intents, the hierarchical relationship between topics and subtopics, and the potential skew caused by long keywords covering multiple topics given as:

$$\alpha\text{DCG} = \sum_{i=1}^{n} \sum_{j=1}^{k} \frac{rel_j * y_{ij} (1-\alpha)^{w_{ji}}}{tok_i * \log_2(1+r_i)} \quad (2)$$

where rel_j (a hyper-parameter) is the relevance associated with the j^{th} intent, and tok_i is the total number of tokens in the i^{th} keyword. To mitigate trade-offs between diversity and relevance during training, we adjusted rel_j through grid search to achieve a flat or better relevance rate with improved diversity on the validation dataset.

3.3 Query Interaction Network

Neural LTR architectures, such as DeepPLTR, have feed-forward layers and compute a score for each keyword independently. This architecture with feed-forward layers fails to capture listwise interaction among keywords mapped to the same prefix as explained in (Qin et al., 2021). Similar to document interaction networks, we use listwise context embedding using self attention layers and further use the latent cross concept for higher order feature interactions, as illustrated in Fig. 4. Suppose a list of n keywords where each keyword feature has dimension d is given, let $X \in \mathbb{R}^{n \times d}$ denote features of the list, thereafter these features are projected using query, key, and value projection matrices to obtain final query, key and value denoted as : Q=XW_q, K=XW_k, V=XW_v. These projection matrices are trainable and $\in \mathbb{R}^{d \times z}$ where z is the size of the attention head.

$$A(X) = Softmax\left(\frac{QK^T}{\sqrt{z}}\right) V \quad (3)$$

Utilizing matrices Q, K, and V, we derive $A(X)$, which is subsequently concatenated from multiple heads and projected back to the original head dimension z through the application of the projection matrix W_o, resulting in the output

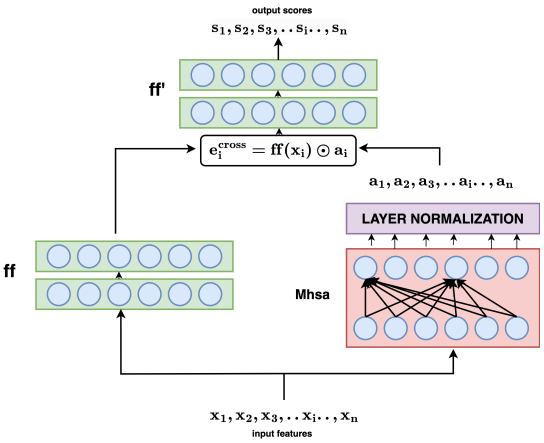

Figure 4: Query Interaction Network, where x_i is the input feature for the i^{th} keyword in the list, a_i is the attention-based embedding from the query interaction (Mhsa) layer, and s_i is the final score for the i^{th} keyword.

$Mhsa(X)$ as depicted in equations 3 and 4.

$$Mhsa(X) = Concat\left(A(X)_{h1}, A(X)_{h2}, \ldots, A(X)_{hh}\right) W_o \quad (4)$$

$$e_i^{cross} = Mhsa(X)_i \odot ff(x_i) \quad and \quad s_i = ff'(e_i^{cross}) \quad (5)$$

Latent cross features, denoted as e_i^{cross} for the i^{th} keyword, are acquired through element-wise multiplication of the embeddings from the Mhsa layers and those obtained from the feed-forward network ff. The final scores, represented by s_i for the i^{th} keyword, are obtained after passing these latent cross features through an additional layer of the feed-forward network ff'.

4 Experiment and Results

Table 1: Comparison of models based on optimization strategies and real-time deployability.

Model	Diversity	Relevance	Listwise	Real-time
DeepPLTR	X	✓	X	✓
moDPLTR	✓	✓	X	✓
LQIN	X	✓	✓	✓
DiALAllRank	✓	✓	✓	✓
DiALAttDin	✓	✓	✓	✓
DiALQIN	✓	✓	✓	✓

To compare the outcomes of the DiAL framework, we use DeepPLTR (Yuan and Kuang, 2021) and moDPLTR (Singh et al., 2023) as baseline methodologies. DeepPLTR optimizes for query relevance, while moDPLTR enhances the diversity of attributes in eCommerce queries by prioritizing queries with a greater number of attributes (or navigational tokens) through pairwise ranking.

The reported results encompass evaluations conducted on internal eCommerce data. To ensure reproducibility, we additionally provide results obtained on the publicly available AOL search logs (Pass et al., 2006) in Appendix A.5. Due to confidentiality, we report relative numbers for the eCommerce dataset, while providing absolute numbers for the AOL dataset.

Model details: For all pairwise baselines, we used a Siamese NN (Fig. 2 of (Singh et al., 2023)). All dense layers use ReLU activation with 128 nodes each. The architectural framework for Listwise Learning to Rank corresponds to the Query Interaction Network explained in Section 3.3. The multi-headed self-attention (Mhsa) layer is configured with 2 heads. This network comprises 6 stacked layers of self-attention. Both the head and tail feed-forward networks share similar structures, featuring three feed-forward layers with dimensions of 128, 256, and 512, respectively. Each connected feed-forward layer incorporates a rectified linear unit (RELU) activation function with batch normalization. All models were trained for a maximum of 15 epochs using the Adam optimizer with a starting learning rate of 3e-5. The best checkpoints were selected for evaluation based on performance on the validation dataset.

eCommerce dataset: Anonymized logs from an eCommerce store are used for generating training samples. A week of customer logged data is used to create a mapping of prefix to the top 100 clicked candidates for each prefix. This prefix-to-candidates map is then merged with session logs based on the prefix. We use 1 week of search logs for training and succeeding 3 days for testing. Each session log entry comprises the prefix, selected keyword, past searches, device type, and other relevant information. Prefixes with fewer than 10 candidates and sessions without a clicked QAC candidate are omitted. Subsequently, for each entry, the query relevance matrix is computed as explained in Section 3.2, by extracting topics and subtopics using an intent tagger tool and assigning binary relevance labels. The dataset is then randomly down-sampled to obtain 80k prefixes for training and 30k prefixes for testing. For assessment, each candidate in the 100 candidates list per prefix is scored and sorted to select the top 10. Features are computed across various time window to ensure robustness against concept drift.

eCommerce Data Features: Similar to DeepPLTR (Yuan and Kuang, 2021), keyword-based features, prefix to keyword-based features, and contextual features (user environment, device type, similarity with past searches) are used. The model's robustness to concept drift is facilitated by these aggregated features over multiple past time windows updated on a daily basis to dynamically capture any emerging drifts or trends. Navigational binary characteristics like the existence of certain categories are provided in combination with existing DeepPLTR features. Additionally, 100-dimensional keyword vectors obtained from training a word-to-vec model on QAC query logs and averaging the word embeddings are appended.

Evaluation Metric: We conduct evaluations using three prominent ranking metrics: Mean Reciprocal Rank (MRR) and Normalized Discounted Cumulative Gain (NDCG) for assessing click-based relevance, and αNDCG for evaluating diversity computed on the top 10 candidates sorted based on the prediction score per prefix.

Table 2: Relative lift in ranking metrics of Listwise LTR (LQIN), Diversified Listwise LTR (DiAL*), and moDPLTR compared to DeepPLTR on the eCommerce dataset.

Model	Network	MRR	NDCG	αNDCG
Navigational AC with pairwise loss				
moDPLTR	Feed-Forward	+2.48%	+1.03%	+2.24%
Listwise Softmax loss				
LQIN	QIN	+9.54%	+5.48%	+2.66%
Listwise Approx αNDCG loss				
DiALAllRank	AllRank	+1.98%	-0.35%	+21.3%
DiALAttDin	AttDin	+6.93%	+3.36%	+22.66%
DiALQIN	QIN	**+8.51%**	**+3.71%**	**+22.83%**

Table 3: Ranking and diversity metrics of DiALQIN with varying QIN parameters: attention heads (H) and encoder layers (L).

Attention heads (H)	Encoder Layers (L)	eCommerce Data MRR	NDCG	αNDCG
1	2	+4.75%	+2.12%	+21.66%
1	4	+7.32%	+3.53%	+21.33%
1	6	+6.93%	+3.62%	+22.67%
2	2	+7.32%	+3.71%	+23.3%
2	4	+5.74%	+2.30%	+22.3%
2	6	+8.51%	+3.71%	+22.83%

4.1 Results

Table 1 details the optimization settings across the evaluated models.

Performance on Losses: Table 2 showcases the performance of Listwise Learning to Rank (LQIN) and Diversified Listwise Learning to Rank (DiAL*) in comparison to the baselines DeepPLTR and moDPLTR on eCommerce dataset. LQIN is trained

with Softmax loss (Appendix A.4) using clicks as relevance, while DiAL* is trained with a diversity-aware loss on clicks and navigational topics. We observe a substantial increase in MRR and NDCG metrics for LQIN compared to DeepPLTR and moDPLTR, suggesting improved click-based relevance using the Softmax listwise loss compared to pairwise loss. Additionally, LQIN utilizes the QIN network, which captures listwise context. We notice a marginal improvement in diversity (αNDCG) using the Softmax loss, attributable to optimizing only click-based relevance while ignoring other navigational utilities. The DiALQIN model, utilizing a diversification metric as the loss, leads to considerable improvements across all three ranking metrics. The improvement in click-based relevance is comparable to LQIN and notable over baselines. The minor decrease in MRR and NDCG compared to the Softmax loss is due to the diversified loss optimizing over multiple relevances instead of solely click-based relevance. Significantly, we observe a substantial improvement in αNDCG compared to moDPLTR, indicating that using an explicit diversification metric as the loss allows for more precise and targeted optimization of diversity compared to other diversity loss variants.

DiAL Performance with varying Encoder Architectures: Further, we emphasize the performance while utilizing different architectures of the Attention network for capturing listwise context using diversity-aware loss. QIN denotes the architecture as described in Section 3.3. AllRank denotes the encoder architecture in (Pobrotyn et al., 2020) which is comparable to the initial encoder architecture presented in (Vaswani et al., 2017) lacking position encoding. AttDin is an attention driven document interaction network from (Pasumarthi et al., 2020) where in lieu of latent cross, the listwise embeddings from the attention layer are concatenated with embeddings from the feed-forward layer. We identify the QIN network (DiALQIN) performs the best among the three architectures with diversification loss.

DiAL Sensitivity with Encoder Parameters: Table 3 presents the outcomes for various encoder configurations within the query interaction layer for eCommerce data. Based on this observation, we found that utilizing 2 attention heads with 6 encoder layers leads to the highest enhancement in both relevance and diversity metrics for the eCommerce data. Overall, our observations indicate that augmenting the number of attention heads while maintaining the same number of encoder layers leads to a more pronounced performance increase compared to maintaining the same number of heads while increasing the number of encoder layers.

DiAL sensitivity with rel_j: Specifically, rel_j was set to 2 for click-based intent, 1 for both non-superfluous intent and topic presence, and 0.5 for subtopic presence using grid-search on validation data. A large value of rel_j (e.g., 10) for click intent compared to smaller values (e.g., 1 and 0.5) for topics and subtopics resulted in behavior and metrics similar to pure relevance-based optimization, while a small value of rel_j (e.g., 1) for click intent and larger values (e.g., 10 or 5) for topics and subtopics led to high diversity but very low relevance.

Interpretability: We demonstrate the increased diversity using DiALQIN in Fig. 5 as compared to baseline DeepPLTR for two example prefixes, 'tops for' and 'induction c' in the eCommerce dataset. QAC using DeepPLTR has few attribute based or 'navigational' suggestions. We note only two completions with price-based ideas ('under 250') and size-based notion 'plus size' for the first example. The various recommendations offered by DiALQIN provide a more even set of recommendations from essential categories. For the prefix 'tops for', brand such as 'max', material such as 'net', size ('long') and price ('under 200') are displayed. Similarly, for 'induction c', brands like 'prestige' and power rating ('2000w') are shown. Fig. 6 illustrates the diversity improvement in DiALQIN as compared to moDPLTR. As moDPLTR upranks queries with navigational terms, we observe the presence of colours, material and gender for the prefix 'tops for'. However, since it doesn't maintain the context of all queries in the list while predicting scores, we may have repeated topics or subtopics in the results. This behaviour can be observed for the prefix 'tops for' where three suggestions can be seen from the same topic 'gender', i.e., 'women', 'ladies', and 'girls'. As DiALQIN penalizes queries belonging to the same topics and subtopics using diversity-aware listwise loss, we see more diversified results such as the presence of 5 topics (gender: 'women', material: 'net', brand: 'max', size: 'long' and price:'under 200') for prefix 'tops for'.

Limitations: The diversification approach only considers categories and attributes defined in the product catalog, limiting its ability to diversify generic user-typed phrases that fall outside these predefined categories and attributes.

DeepPLTR	DiALQIN
tops for women	tops for women
tops for women western wear	tops for women net
tops for women stylish latest under 250	tops for women max
tops for women churidar tops	tops for women long
tops for women plus size	tops for women jeans
tops for jeggings for women stylish	tops for women under 200

DeepPLTR	DiALQIN
induction cooktop	induction cooktop
induction cookware set	induction cooker
induction cooker	induction cookware set
induction cookware	induction cooktop prestige
induction cooker 3l	induction cooker 5litres+
induction cooker 2l	induction cooktop prestige 2000w

Figure 5: Comparing diversity-aware listwise ranking (DiALQIN) with pairwise ranking (DeepPLTR) for prefixes 'tops for' and 'induction c' from the eCommerce dataset.

moDPLTR	DiALQIN
tops for women	tops for women
tops for women net	tops for women net
tops for women cotton	tops for women max
tops for women green	tops for women long
tops for ladies	tops for women jeans
tops for girls	tops for women under 200

moDPLTR	DiALQIN
induction cooktop	induction cooktop
induction cooker 3l	induction cooker
induction cooker 2l	induction cookware set
induction cooktop 2000w	induction cooktop prestige
induction cookware	induction cooker 5litres+
induction cooker 1.5l	induction cooktop prestige 2000w

Figure 6: Comparing diversity-aware listwise ranking (DiALQIN) with navigational pairwise ranking (moDPLTR) for prefixes 'tops for' and 'induction c' from the eCommerce dataset.

4.2 Online deployment (A/B test)

We conducted an online A/B test by implementing DiALQIN on the search bar of a major eCommerce store. The model operated in real-time and provided comprehensive coverage for all prefixes during the test sessions.

Duration and Statistical Significance: Our A/B test lasted for more than a week, achieving statistical significance with 99% power, covering around 10 million customer search sessions.

Click-Through Rate (CTR): We observed a significant lift of 0.02% in CTR, indicating that customers positively received the diverse suggestions in QAC.

Revenue: Our A/B test yielded a significant revenue boost of 0.48%. Our tests in the same setting revealed that DiALQIN resulted in a significant revenue lift over moDPLTR.

Diversity: Addressing diversification in QAC resulted in a 10.6% increase in product diversity during the online test, along with a revenue boost, manifesting the downstream impact of diversified QAC suggestions.

Latency: DiALQIN served an average of 100k QAC requests per second without any notable latency issues. Although the latency for the listwise models (DiALQIN) was reported to be 15ms versus 2ms for the pairwise (moDPLTR) model, it remained well below the required limit without any noticeable impact on serving end consumers.

5 Conclusion

In this paper, we introduce diversified listwise LTR for the QAC task using a score-and-sort strategy. Unlike previous greedy approaches employed in QAC, the score-and-sort approach is quicker and more efficient in diversifying QAC suggestions. We demonstrate the importance of capturing listwise context in QAC ranking with Query Interaction Network and show improved click based relevance and relevance w.r.t different product attributes. Further, we diversify QAC on different dimensions (or intents) and modify the approx αNDCG loss to penalize longer queries and assign different weights to the intents based on their relative importance. This technique jointly boosted relevance and diversity with speedy inference.

References

2017. aol query log analysis. https://github.com/wasiahmad/aol_query_log_analysis/.

2022. sentence transformers. https://huggingface.co/sentence-transformers/multi-qa-MiniLM-L6-cos-v1.

Ziv Bar-Yossef and Naama Kraus. 2011. Context-sensitive query auto-completion. In *Proceedings of the 20th International Conference on World Wide Web (WWW)*, pages 107–116.

Sebastian Bruch, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2019a. An analysis of the softmax cross entropy loss for learning-to-rank with binary relevance. In *Proceedings of the 2019 ACM SIGIR international conference on theory of information retrieval*, pages 75–78.

Sebastian Bruch, Masrour Zoghi, Michael Bendersky, and Marc Najork. 2019b. Revisiting approximate metric optimization in the age of deep neural networks. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 1241–1244.

Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81.

Fei Cai and Maarten de Rijke. 2016. Learning from homologous queries and semantically related terms for query auto completion. *Information Processing & Management*, 52(4):628–643.

Fei Cai, Maarten De Rijke, et al. 2016a. A survey of query auto completion in information retrieval. *Foundations and Trends® in Information Retrieval*, 10(4):273–363.

Fei Cai, Ridho Reinanda, and Maarten De Rijke. 2016b. Diversifying query auto-completion. *ACM Transactions on Information Systems (TOIS)*, 34(4):1–33.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336.

Olivier Chapelle, Donald Metlzer, Ya Zhang, and Pierre Grinspan. 2009. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 621–630.

Charles LA Clarke, Maheedhar Kolla, Gordon V Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 659–666.

Van Dang and Bruce W Croft. 2013. Term level search result diversification. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 603–612.

Yue Feng, Jun Xu, Yanyan Lan, Jiafeng Guo, Wei Zeng, and Xueqi Cheng. 2018. From greedy selection to exploratory decision-making: Diverse ranking with policy-value networks. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 125–134.

Nicolas Fiorini and Zhiyong Lu. 2018. Personalized neural language models for real-world query auto completion. *arXiv preprint arXiv:1804.06439*.

Aditya Grover, Eric Wang, Aaron Zweig, and Stefano Ermon. 2019. Stochastic optimization of sorting networks via continuous relaxations. *arXiv preprint arXiv:1903.08850*.

Sha Hu, Zhicheng Dou, Xiaojie Wang, Tetsuya Sakai, and Ji-Rong Wen. 2015. Search result diversification based on hierarchical intents. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 63–72.

Gyuwan Kim. 2019. Subword language model for query auto-completion. *arXiv preprint arXiv:1909.00599*.

Dae Hoon Park and Rikio Chiba. 2017. A neural language model for query auto-completion. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1189–1192.

Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A picture of search. In *Proceedings of the 1st international conference on Scalable information systems*, pages 1–es.

Rama Kumar Pasumarthi, Honglei Zhuang, Xuanhui Wang, Michael Bendersky, and Marc Najork. 2020. Permutation equivariant document interaction network for neural learning to rank. In *Proceedings of the 2020 ACM SIGIR on international conference on theory of information retrieval*, pages 145–148.

Przemysław Pobrotyn, Tomasz Bartczak, Mikołaj Synowiec, Radosław Białobrzeski, and Jarosław Bojar. 2020. Context-aware learning to rank with self-attention. *arXiv preprint arXiv:2005.10084*.

Tao Qin, Tie-Yan Liu, and Hang Li. 2010. A general approximation framework for direct optimization of information retrieval measures. *Information retrieval*, 13:375–397.

Zhen Qin, Le Yan, Honglei Zhuang, Yi Tay, Rama Kumar Pasumarthi, Xuanhui Wang, Mike Bendersky, and Marc Najork. 2021. Are neural rankers still outperformed by gradient boosted decision trees?

Fiana Raiber and Oren Kurland. 2013. Ranking document clusters using markov random fields. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 333–342.

Scott Sanner, Shengbo Guo, Thore Graepel, Sadegh Kharazmi, and Sarvnaz Karimi. 2011. Diverse retrieval via greedy optimization of expected 1-call@ k in a latent subtopic relevance model. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1977–1980.

Rodrygo LT Santos, Craig Macdonald, Iadh Ounis, et al. 2015. Search result diversification. *Foundations and Trends® in Information Retrieval*, 9(1):1–90.

Sheikh Muhammad Sarwar, Raghavendra Addanki, Ali Montazeralghaem, Soumyabrata Pal, and James Allan. 2020. Search result diversification with guarantee of topic proportionality. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*, pages 53–60.

Sonali Singh, Sachin Farfade, and Prakash Mandayam Comar. 2023. Multi-objective ranking to boost navigational suggestions in ecommerce autocomplete.

Aleksandrs Slivkins, Filip Radlinski, and Sreenivas Gollapudi. 2010. Learning optimally diverse rankings over large document collections. In *Proc. of the 27th International Conference on Machine Learning (ICML 2010)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Le Yan, Zhen Qin, Rama Kumar Pasumarthi, Xuanhui Wang, and Michael Bendersky. 2021. Diversification-aware learning to rank using distributed representation. In *Proceedings of the Web Conference 2021*, pages 127–136.

Hai-Tao Yu. 2022. Optimize what you evaluate with: Search result diversification based on metric optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10399–10407.

Hai-Tao Yu, Adam Jatowt, Roi Blanco, Hideo Joho, Joemon M Jose, Long Chen, and Fajie Yuan. 2018. Revisiting the cluster-based paradigm for implicit search result diversification. *Information Processing & Management*, 54(4):507–528.

Kai Yuan and Da Kuang. 2021. Deep pairwise learning to rank for search autocomplete. *arXiv preprint arXiv:2108.04976*.

A Appendix

A.1 Approximate αNDCG

αNDCG is a normalized version of αDCG,

$$\alpha\text{NDCG} = \frac{\alpha\text{DCG}}{\alpha\text{DCG}_{ideal}} \tag{6}$$

In order to obtain the approximate differentiable version of the above metric, the rank of keyword r_i with score s_i and intent coverage w_{ji} is denoted as:

$$r_i = 1 + \sum_j I_{s_j > s_i} \quad and \quad w_{ji} = \sum_m y_{mj} I_{s_m > s_i} \tag{7}$$

In the above equation, the indicator functions are not differentiable, but can be smoothed by using the sigmoid function:

$$R_i = 1 + \sum_{j \neq i} sigmoid(s_j - s_i) \tag{8}$$

$$W_{ji} = \sum_{m \neq i} y_{mj} sigmoid(s_m - s_i) \tag{9}$$

We use these smooth approximations in equation 1 to obtain the final differentiable loss.

A.2 Diversity problem in QAC via. Examples.

Diversity issue in QAC is illustrated with an example in Fig. 7 where top recommendations for prefix 'jacket' and 'mobile' contain many non-informative suggestions that are almost synonyms of each other, resulting in search experience that are indistinguishable for various choices of suggestions presented to the users.

jacket	mobile
jacket for men	mobile
jacket for women	mobile under 10000
jacket	mobile under 1000+0
jacket for boys	mobile phone
jacket for men stylish latest	mobile stand
jacket for girls	mobile under 15000
jacket for women stylish latest	mobile under 20000
jacket for men stylish	mobile holder for car
jacket for men winter wear	mobile under 7000

Figure 7: Prefixes 'jacket' (left) and 'mobile' (right) in India marketplace. QAC suggestions lack specifics on color, material, style, size for 'jacket', and brand, configuration for 'mobile'.

A.3 Baselines

DeepPLTR: We adopt the DeepPLTR model (Yuan and Kuang, 2021) as the baseline QAC approach that ranks keywords with the goals of maximizing relevance and session revenue. It optimizes on a pairwise loss function that learns to rank an accepted (clicked or fully-typed) keyword higher than the rejected (non-clicked) keyword. For each prefix p and its list of n completed keywords, the accepted keyword (positive ✓) is sampled and paired with all rejected keywords (negative X). The positive and negative keywords are featurized and input to a Siamese Neural Network with pairwise cross-entropy loss. The loss term is weighted by $w^{revenue}$, which is the logarithm of the revenue obtained by clicking on the prefix in the session from which the evaluation data is sampled. This weighting biases the model towards revenue-generating prefixes. The pairwise loss for prefix p, completing to n keywords in list l is given as:

$$L_p = \frac{1}{(n-1)} \sum_{(k+,k-) \in l} w^{revenue} \left[log(1 + e^{(s_{k-} - s_{k+})}) * |\Delta_{(k+,k-)}| \right] \tag{10}$$

where s_k denotes the output scores for keyword k from the neural network, $|\Delta_{(k+,k-)}|$ denotes the difference in reciprocal rank of the positive and negative keywords. The term $|\Delta_{(k+,k-)}|$ ensures that the score of k_+ is larger than k_- if the gap in their relative ranks is higher.

moDPLTR: moDPLTR (Singh et al., 2023) augments the customer behavior (CB) objective optimized on clicks and revenue with a query quality (QQ) objective to uprank queries with navigational aspects. It is adopted as the diversity-aware baseline. A navigational score y_k^{rel} is assigned to each query based on the presence of navigational tokens like product, brand, and color. The QQ objective improves the correlation between the acceptance score f_k and navigational score y_k^{rel} in a batch. The loss is stated below, where L_p denotes the CB ob-

jective (same as equation 10) and the correlation term denotes the QQ objective.

$$L_{corr} = \lambda_1 L_p - \lambda_2 \left\{ Corr(\mathbf{f}_{k+}, \mathbf{y}^{rel}_{k+}) \right\} \quad (11)$$

To replicate this model for AOL search logs, we assign a navigational score proportional to the likelihood of topic presence in the query.

A.4 Listwise Learning to Rank

Learning to rank (LTR) algorithms are classified into pointwise, pairwise, and listwise based on the choice of loss functions. Pointwise loss assesses each item independently, pairwise samples pairs and learns to rank one higher than the other, while listwise takes the entire list as one instance and calculates loss. Pairwise LTR techniques like LambdaMart (Burges, 2010) are state-of-the-art for ranking keywords on single labels, such as clicks. However, for diversification, it becomes difficult to incorporate pairwise loss as all items in the list need to be diversified, and the contextual knowledge connected to the list of keywords with the same prefix is missing. Usually, diversity is added as a step after the initial ranking using pairwise loss, which is computationally expensive. This drives us to use Listwise LTR methods for jointly modeling relevance using clicks and diversity. Recent research has progressed significantly on Neural LTR losses, specially Listwise ranking losses, such as Softmax (Bruch et al., 2019a), ApproxNDCG (Qin et al., 2010), and NeuralSortNDCG (Grover et al., 2019). Softmax is known to be the simplest, yet robust for modeling listwise relevance. Let y_i be the relevance label associated with the i^{th} item in the list of n items and s_i be the corresponding neural score, then the Softmax loss for the list is given as:

$$L_{Softmax} = -\sum_{i=1}^{n} y_i \log \left(\frac{e^{s_i}}{\sum_{k=1}^{n} e^{s_k}} \right) \quad (12)$$

A.5 Details on reproducibility on external dataset

AOL search logs: We utilize publicly accessible query topic analysis data (git, 2017), derived from AOL search logs (Pass et al., 2006), as our dataset. The query topic analysis involves the examination of the top 1000 user search logs with the highest frequency of search queries. Each query within this dataset is annotated with a primary and secondary topic, along with associated scores. For instance, the query 'country inn & suites' is labeled with the best topic 'Hospitality' and the second-best topic 'Hotels_and_Motels'. The dataset encompasses a total of 318,023 queries. In the absence of a prefix for each query, we use a strategy similar to (Kim, 2019) to uniformly sample prefix for each query. Each searched query is treated as the clicked query, and a list of the top 100 clicked queries starting with the same prefix from these logs is appended as candidates. A total of 90k prefixes are randomly sampled for training, and 30k prefixes are allocated for testing purposes. We construct a query relevance matrix, comprising a total of 30 topics, by combining click-based relevance and aggregating the top 29 most frequently occurring topics per prefix-candidate list, utilized for both loss computation and evaluation purposes.

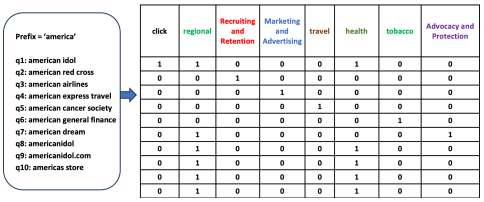

Figure 8: A list of prefix queries on the left and the corresponding query relevance matrix on the right, utilized for assessing diversity loss and conducting evaluations specific to AOL search logs data.

AOL dataset Features: We utilize a 384-dimensional embedding acquired from the Sentence Transformer model 'multi-qa-MiniLM-L6-cos-v1' (sen, 2022), tailored specifically for semantic search, as a feature for each candidate within the list. Furthermore, our feature set incorporates aggregated click information from the preceding logs, the cosine similarity of the query embedding with the two preceding queries within a 300-second timeframe, the ratio of prefix to query length, and a binary feature indicating the presence of recent past two searches. Additionally, we account for the temporal difference in timestamps between the ongoing search and the two prior searches.

Query Relevance Matrix: In the context of AOL search logs, we formulate a query relevance matrix, as illustrated in Fig. 8, employing pre-extracted topics from query topic analysis data conducted on AOL search logs, accessible at (git, 2017). We employ two distinct intents for diversity: click-based intent and topic presence, using identical hyper-parameters as those utilized in the eCom-

merce dataset. We adopt 30 intent labels, designating one for click-based intent and the remaining 29 for the presence of the top 29 most frequent topics within the prefix candidate list.

Table 4: Performance comparison of DeepPLTR, moDPLTR, Listwise Learning to Rank (LQIN), and Diversified Listwise Learning to Rank (DiAL*) models on the AOL dataset.

Model	Network	MRR	NDCG	αNDCG
Pairwise Cross-Entropy Loss				
DeepPLTR	Feed-Forward	0.384	0.438	0.578
Navigational AC with pairwise loss				
moDPLTR	Feed-Forward	0.381(-0.78%)	0.435(-0.69%)	0.585(+1.21%)
Listwise Softmax loss				
LQIN	QIN	0.417(+8.59%)	0.469(+7.07%)	0.601(+3.97%)
Listwise Approx αNDCG loss				
DiALAllRank	AllRank	0.404(+5.20%)	0.460(+5.02%)	0.662(+14.53%)
DiALAttDin	AttDin	0.403(+4.94%)	0.460(+5.02%)	0.676(+16.95%)
DiALQIN	QIN	**0.409(+6.51%)**	**0.463(+5.70%)**	**0.681(+17.82%)**

Table 5: Ranking and diversity metrics of DiALQIN with varying QIN parameters: attention heads (H) and encoder layers (L).

Attention	Encoder	AOL Search Logs		
heads (H)	Layers (L)	MRR	NDCG	αNDCG
1	2	0.394	0.453	0.631
1	4	0.397	0.457	0.635
1	6	0.398	0.457	0.643
2	2	0.405	0.462	0.680
2	4	0.409	0.463	0.681
2	6	0.408	0.460	0.680

Results: In Tables 4 and 5 we present results comparing diversity and relevance for various losses and encoder architectures. The trends noted are similar to eCommerce dataset reflecting that DiALQIN achieves significant improvement in relevance and diversity over pairwise baselines. We also exhibit improvement in query topics diversity for QAC using the AOL dataset in Fig. 9 using DiALQIN compared to baseline DeepPLTR for an example prefix 'i'. In the case of DeepPLTR approach, we observe that 4 out of 6 queries pertain to the same topic 'regional', encompassing a total of 3 topics ('regional', 'news', and 'software for engineering') within the top 6 results. In contrast, with DiALQIN, we observe a broader range of 5 topics ('regional', 'software for engineering', 'associations', 'abuse', and 'directories') in total.

DeepPLTR	DiALQIN
Internet[topic=regional]	Italy[topic=regional]
Idol[topic=regional]	Internet[topic=regional]
Iran[topic=regional]	Interest rates[topic=software for engineering]
Interesting websites[topic=news]	Internet explorer[topic=associations]
Indiana lotter[topic=regional]	Inspirational quotes[topic=abuse]
Interest rate on savings[topic=software for engineering]	Interesting facts[topic=dictionaries]

Figure 9: Comparing diversity-aware listwise ranking (DiALQIN) with pairwise ranking (DeepPLTR) for prefix 'i' from the AOL search logs dataset.