Scaling Search Relevance: Augmenting App Store Ranking with LLM-Generated Judgments

Evangelia Christakopoulou
Apple
Cupertino, CA, USA
echristakopoulou@apple.com

Vivekkumar Patel
Apple
Cupertino, CA, USA
vpatel22@apple.com

Hemanth Velaga
Apple
Seattle, WA, USA
h_velaga@apple.com

Sandip Gaikwad
Apple
Cupertino, CA, USA
sandip_gaikwad@apple.com

Sean Suchter
Apple
Cupertino, CA, USA
ssuchter@apple.com

Venkat Sundaranatha
Apple
Cupertino, CA, USA
vsundaranatha@apple.com

Abstract

Large-scale commercial search systems optimize for relevance to drive successful sessions that help users find what they are looking for. To maximize relevance, we leverage two complementary objectives: behavioral relevance (results users tend to click or download) and textual relevance (a result's semantic fit to the query). A persistent challenge is the scarcity of expert-provided textual relevance labels relative to abundant behavioral relevance labels. We first address this by systematically evaluating LLM configurations, finding that a specialized, fine-tuned model significantly outperforms a much larger pre-trained one in providing highly relevant labels. Using this model as a force multiplier for human annotation, we generate millions of textual relevance labels to overcome the data scarcity. We show that augmenting our production ranker with these textual relevance labels leads to a significant outward shift of the Pareto frontier: offline NDCG improves for behavioral relevance while simultaneously increasing for textual relevance. These offline gains were validated by a worldwide A/B test on the App Store ranker, which demonstrated a statistically significant **+0.24%** increase in conversion rate, with the most substantial performance gains occurring in tail queries, where the new textual relevance labels provide a robust signal in the absence of reliable behavioral relevance labels.

ACM Reference Format:
Evangelia Christakopoulou, Vivekkumar Patel, Hemanth Velaga, Sandip Gaikwad, Sean Suchter, and Venkat Sundaranatha. 2026. Scaling Search Relevance: Augmenting App Store Ranking with LLM-Generated Judgments. In *Proceedings of Preprint*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Search ranking systems for large-scale digital marketplaces like the App Store are critical for app discovery and user satisfaction. The effectiveness of these systems is paramount to ensuring users can easily discover and engage with the vast array of applications available. These systems need to optimize for both textual relevance (the semantic fit of a result, as judged by experts) and behavioral relevance (the likelihood of a user clicking or downloading a result) to ensure good user experience. While behavioral relevance labels are abundant, textual relevance labels generated by human judges are much rarer. This creates a fundamental problem: high-quality textual relevance labels are scarce and expensive to produce, creating a scalability bottleneck and leaving the textual relevance objective under-powered in multi-objective training.

To address this label scarcity, recent work has explored LLMs as scalable offline generators of training data [3] and as automated judges [33]. In this work, we validate this "LLM-as-a-Judge" paradigm at industrial scale. We fine-tune an in-house LLM on existing judgments from human judges and use it to generate millions of new high-quality textual relevance labels across multiple storefronts and languages. We then use these new labels as an additional input to train the production ranker, effectively overcoming traditional annotation bottlenecks. This methodology does more than improve a single metric; it shifts outwards the behavioral-textual relevance Pareto frontier, enabling the ranker to achieve superior performance.

Our contributions are four-fold:

(1) We deploy large-scale experiments with multiple configurations through pretrained and fine-tuned in-house large language models, to determine the best way to get the highest quality relevance labels for the specific task of relevance annotation.
(2) We generate millions of pointwise textual relevance labels across multiple languages, vastly expanding our training data.
(3) We show that augmenting the training data of our ranker with millions of these LLM-generated labels leads to a significant outward shift of the behavioral-textual relevance Pareto frontier, strictly dominating the production model in offline NDCG metrics.
(4) We validate these gains through both offline evaluation and online A/B testing, confirming the effectiveness of our approach in a real-world commercial environment. Notably, the results reveal that the LLM-augmented model provides the greatest conversion rate improvements for tail queries, effectively bridging the relevance gap in areas where reliable behavioral data is most scarce.

Preprint, USA

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

The rest of the paper presents the related work, details our approach and presents a comprehensive analysis of our offline and online results.

2 Related work

LLMs as Rankers. Large language models (LLMs) have demonstrated success as zero-shot and few-shot rankers using pointwise, pairwise, and listwise prompting strategies [1, 5, 13, 21–23, 27, 34, 35]. Our work differs from these approaches as they mainly focus on deploying LLMs as real-time (re)rankers; instead, we use an LLM as an offline label generator to create large-scale textual relevance labels for a production ranker.

LLMs as Judges. The "LLM-as-a-Judge" paradigm utilizes LLMs as scalable offline annotators to mitigate the scarcity of labels from human judges [2, 6, 8, 9, 11, 16, 17, 25, 29, 32]. Our paper distinguishes itself by using a fine-tuned, in-house LLM not just for evaluation, but as a way to generate millions of textual relevance labels used directly as training data for our production ranker. In contrast to approaches like RRADistill [7], which focuses on distilling the semantic understanding of large LLMs into smaller language models with specialized architectural modifications, our work utilizes pointwise labels to train the production ranker without modifying its internal architecture.

Multi-Objective Learning to Rank (LTR). Multi-objective learning to rank (LTR) [15] is a principled way to balance diverse objectives, such as different forms of relevance or fairness [4, 10, 18–20, 28]. Our work operates within this framework, but rather than introducing a new objective, we focus on strengthening the existing textual relevance objective by dramatically expanding its label coverage via an LLM. This differs from the approach of Liu et al. [14] who also combine content and behavioral signals but do so by applying a sigmoid transformation to the LLM-generated label, whereas we utilize a data-mixing approach within our multi-objective framework.

Preference Alignment and Prompting for Ranking Objectives. A growing body of research examines how to align LLM preferences with ranking-oriented objectives [24, 26, 30, 31]. The common goal of these methods is to improve the LLM's ability to act as a ranker itself. While our work is inspired by these alignment techniques in our prompt and model design, our goal is fundamentally different. We do not seek to deploy the LLM as a ranker, but to obtain relevance labels that align with the existing human-rated rubric and can be used for training the ranker.

Positioning of Our Work. Our contribution connects these strands in a large-scale production system. We demonstrate that using a fine-tuned LLM as an offline judge to augment the training data for a multi-objective ranker is a scalable and effective industrial strategy. We show this approach shifts outwards the behavioral-textual relevance Pareto frontier, leading to measurable improvements in both offline metrics and online A/B tests.

3 Proposed approach

Our approach systematically augments the training data for our production ranker. It consists of two main stages:

(1) generating millions of high-quality, textual relevance labels, and
(2) integrating these new labels into our multi-objective ranker training pipeline.

3.1 LLM relevance labels generation

> Imagine you are an App Store evaluator. You are given a user search term and an app returned for this query, with app metadata 1, app metadata 2, app metadata 3.
>
> The goal is to choose one of the following labels for the app given the query: label_1, label_2, label_3, label_4, label_5.
>
> Description of relevance levels is as follows: …
>
> **Strict guideline:** Your response should only be the label and nothing else.
>
> **Example 1:** query: query_1; app: app_1 with associated app metadata: app_metadata_1, app_metadata_2, app_metadata_3; label: label_1.
> **Example 2:** query: query_2; app: app_2 with associated app metadata: app_metadata_1, app_metadata_2, app_metadata_3; label: label_5.
>
> Now generate the label for this: query: query_target; app: app_target with associated app metadata: app_metadata_1, app_metadata_2, app_metadata_3.

Figure 1: Example few-shot prompt for query (query_target) and returned app (app_target).

To address the scarcity of textual relevance labels from human judges, we adopt an LLM-as-a-Judge methodology where a model acts as a scalable offline annotator. We leverage historical search logs aggregated across queries, containing $n \times m$ candidate query-app pairs. For each pair, our goal is to generate a pointwise textual relevance label.

Our framework supports both large-scale pretrained and specialized fine-tuned in-house models. While fine-tuning on existing judgments from human judges allows a model to better internalize our specific ranking rubric, using high-capacity pretrained models remains a viable path. By performing this generation offline, we create a force multiplier for human annotation without the latency constraints of real-time LLM reranking.

To ensure the generated labels align with our human judges, we construct prompts using the same query and app metadata available to them. An example prompt is shown in Figure 1. Our approach explores both zero-shot configurations and few-shot prompting, which incorporates previously rated examples from human judges to guide the model's judgment. We also use the judgments they have provided as labels for finetuning. These generated labels are represented as either string or numeric values and they correspond to ordinal relevance levels matching the rubric used by the human judges. They are then integrated into our multi-objective ranker to strengthen the textual relevance objective.

3.2 Multi-objective ranker training

Our production ranker is designed to optimize relevance. This is achieved within a multi-objective framework to jointly optimize for textual and behavioral relevance.

To achieve this, we construct a unique training dataset from two distinct label sources: downloads and clicks from aggregated App Store search logs (for behavioral relevance) and explicit textual

relevance judgments (from human judges and the LLM-generated relevance labels described in Section 3.1). Crucially, the same query-app pair, represented by an identical feature vector, can appear in the training data multiple times—once with a behavioral relevance label and once with a textual relevance label.

Our ranker training is a practical application of Multi-Objective Optimization (MOO) designed to find optimal solutions along the Pareto frontier. We employ a common MOO technique known as scalarization, where we combine the two objectives into a single objective function by creating a weighted mix of the training data. The strict separation of label sources during comparison is crucial, as it ensures the gradients for each objective are computed independently.

This approach allows us to control the relative influence of each objective by treating the data mixing ratio as a tunable hyperparameter. By sampling or limiting rows from each data source, we can construct datasets with any desired mix (e.g., $90-10$, $70-30$, $50-50$), enabling us to systematically train different models that correspond to different points on the Pareto frontier.

4 Offline Experimental Evaluation

We conduct a two-stage offline evaluation to validate our approach: first, we assess the quality of the LLM-generated labels, and second, we measure their impact on the production ranker.

4.1 Dataset

Our evaluation utilizes two primary data sources:

(1) a large-scale dataset of millions of query-app pairs from historical App Store session logs aggregated across queries over a large time window, and
(2) a much smaller dataset of historical textual relevance judgments on query-app pairs from our human judges.

The second, smaller dataset of human judgments is split into training and validation sets, used to fine-tune and evaluate the LLM judge respectively. We then use the LLM judge to perform inference on the first, large-scale aggregated query-app log data to generate new textual relevance labels.

Then, we construct the data for training the ranker by combining the two primary data sources: historical aggregated logs (for behavioral relevance) and labels from human judges (for textual relevance), but also the new LLM-generated labels (for improved textual relevance). We also split this dataset into training and validation for the ranker.

4.2 Offline results on the LLM relevance labels

4.2.1 Experimental Setup. We compared three LLM configurations:

(1) **Pretrained 3B**: A pretrained 3-billion parameter in-house model.
(2) **Pretrained 30B**: A pretrained 30-billion parameter in-house model.
(3) **Finetuned 3B**: The 3B model, fine-tuned on the training set of human judgments dataset described in Section 4.1.

We experimented with various prompt configurations (zero-shot vs. few-shot, string vs. numeric labels) using query and app metadata. To determine the best configuration, we evaluated each model's ability to predict the relevance label assigned by human judges on a held-out validation set. We report precision, recall, and F1 scores across all relevance classes.

Table 1: Offline evaluation of LLM configurations. The finetuned model (`FT-3B`) most closely reproduces the labels from the human judges. Pretrained models are labeled `PT`.

Model	Precision	Recall	F1
PT-3B	0.300	0.309	0.287
PT-30B	0.424	0.402	0.382
FT-3B	**0.802**	**0.798**	**0.800**

4.2.2 Results. Table 1 shows the results for the best performing model configurations on the validation set, measuring how closely each model reproduces the labels provided by the human judges across all levels.

We can see that the finetuned model (`FT-3B`) outperforms the pretrained models in all cases. This is notable, as it is expected to outperform the pretrained model with the same number of parameters, but it also outperforms the pretrained model with ten times more parameters, showcasing the value of finetuning. This result carries significant industrial implications: fine-tuning a smaller, more efficient model proved more effective than using its much larger, pretrained counterpart, offering a clear path to production with lower computational and operational costs.

While fine-tuning the 30B model remains a compelling direction for future work, our findings confirmed that the finetuned 3B model provided a sufficiently strong and cost-effective solution.

For brevity, we omit detailed results on prompt design, but note that few-shot prompts with string labels performed best. The labels generated by this optimal `FT-3B` configuration are used to augment the ranker's training data, which we evaluate next.

4.3 Offline Ranker Results

Following the generation of LLM-based relevance labels, we conducted a series of large-scale experiments to measure their impact on the performance of the production ranker. The primary goal was to see the effect on textual and behavioral relevance as we augmented the training data.

4.3.1 Experimental Setup. We compare two models:

(1) **`prod`**: The production model, trained using the behavioral relevance labels from the aggregated App Store logs and the limited set of textual relevance labels from human judges.
(2) **`llm-augmented`**: The `prod` model, with its training data further augmented by millions of LLM-generated textual relevance labels generated as described in Section 4.2.

We hold out a validation set to measure performance using the NDCG@k metric [12]. The relevance score for each item used in the NDCG calculation is defined differently for each objective: for textual relevance, it is the judgment provided from a human judge; for behavioral relevance, it is a score derived from aggregated user clicks and downloads. Note that the judgments used for computing the NDCG@k for the textual relevance are disjoint from those used

Table 2: Offline Ranker Performance (NDCG). Augmenting the training data with LLM-generated labels (`llm-augmented`) improves relevance metrics compared to the production model (`prod`).

	Relevance					
	Textual			Behavioral		
Model	NDCG@1	NDCG@3	NDCG@7	NDCG@1	NDCG@3	NDCG@7
`prod`	0.867	0.803	0.760	0.646	0.479	0.403
`llm-augmented`	**0.868**	**0.805**	**0.761**	**0.652**	**0.484**	**0.407**

for fine-tuning and validation of the LLM relevance labels; these are distinct, non-overlapping sets.

4.3.2 Results. The results of our offline evaluation are summarized in Table 2, in terms of behavioral and textual relevance NDCG at ranks 1, 3 and 7 of returned results.

These results confirm our central hypothesis that adding LLM-generated labels helps improve the textual relevance of the ranker. They additionally improve the behavioral relevance of the ranker, showcasing that the increase of the training data with LLM-generated labels not only increases the semantic relevance of the results to the query, but also increases the likelihood of users downloading or clicking the results. The results are consistent across all different values of k. This demonstrates a true Pareto improvement, shifting the Pareto frontier outwards, meaning the `prod` model's performance is strictly dominated.

Further experimentation also revealed that by varying the proportion of LLM-generated labels, we could then move along this new, superior frontier, rather than shifting the frontier further outward.

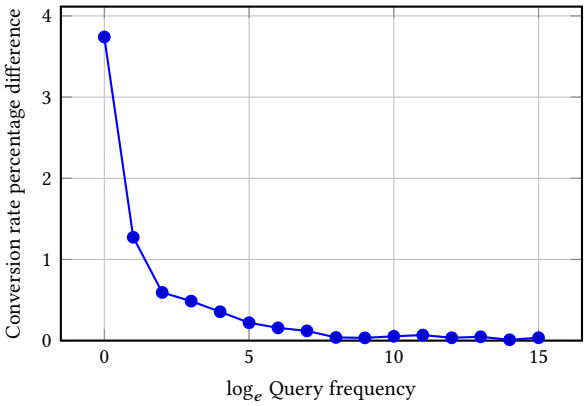

Figure 2: Conversion rate percentage difference of `llm-augmented` vs `prod` model across query frequency buckets. The lower-numbered buckets correspond to lower frequency - more tail queries, while the higher-numbered buckets correspond to higher frequency-more head queries.

5 A/B test results

Table 3: A/B test results of the `llm-augmented` model in terms of percentage improvement over the `prod` model with respect to conversion rate. The improvement shown is statistically significant.

Model	Conversion rate
`prod`	-
`llm-augmented`	**+0.24%**

We validated our approach with a large-scale online A/B test that ran on worldwide traffic, comparing our `llm-augmented` model against the production model (`prod`).

As shown in Table 3, the `llm-augmented` model demonstrated a statistically significant **+0.24%** increase in our primary metric, conversion rate, defined as the proportion of search sessions with at least one app download. While this number may appear small, it is considered a significant improvement for a mature industrial ranker. Across storefronts worldwide, the `llm-augmented` model outperformed `prod` in conversion rate in 89% of the storefronts.

To understand the source of these gains, we analyzed the conversion rate lift across query frequency buckets, from low-frequency (tail) queries to high-frequency (head) queries. The results, shown in Figure 2, reveal that the most substantial improvements occur in the tail. This is because tail queries, by definition, lack sufficient user traffic to generate reliable behavioral relevance signals. Our `llm-augmented` model excels here because the newly added textual relevance labels provide a robust and accurate signal where the behavioral signal is sparse or absent, effectively closing the relevance gap.

6 Conclusion and future work

By systematically evaluating both large-scale pretrained and specialized fine-tuned in-house models, we demonstrate that LLMs can serve as a powerful force multiplier for human judges, effectively shifting the Pareto frontier in a production ranking pipeline. Our findings, validated through rigorous offline evaluation and online A/B testing, confirm the industrial scalability of the LLM-as-a-Judge paradigm and highlight that LLM-generated labels are most impactful for tail queries, where they provide a robust signal in the absence of reliable behavioral data. Our work provides a practical blueprint for other large-scale search systems to overcome relevance-label scarcity and systematically improve their ranking models by leveraging the power of modern LLMs as offline data generators.

In the future, we plan to experiment with additional fine-tuned models, as well as additional types of prompt creation, such as

pairwise and listwise configurations that will allow us to generate labels for pairs or lists of apps for a particular query.

Acknowledgments

We would like to thank Don Dini and Vivek Kanojiya for inspirational and helpful discussions in the early stages of this work. We are grateful to Ian Fischer and Aashir Gajjar for their engineering contributions and technical support.

References

[1] Abdelrahman Abdallah, Bhawna Piryani, Jamshid Mozafari, Mohammed Ali, and Adam Jatowt. 2025. How Good are LLM-based Rerankers? An Empirical Analysis of State-of-the-Art Reranking Models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 5693–5709. doi:10.18653/v1/2025.findings-emnlp.305

[2] Negar Arabzadeh and Charles L. A. Clarke. 2025. Benchmarking LLM-based Relevance Judgment Methods. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Padua, Italy) *(SIGIR '25)*. Association for Computing Machinery, New York, NY, USA, 3194–3204. doi:10.1145/3726302.3730305

[3] Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. 2022. InPars: Unsupervised Dataset Generation for Information Retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 2387–2392. doi:10.1145/3477495.3531863

[4] David Carmel, Elad Haramaty, Arnon Lazerson, and Liane Lewin-Eytan. 2020. Multi-objective ranking optimization for product search using stochastic label aggregation. In *Proceedings of The Web Conference 2020*. 373–383.

[5] Yiqun Chen, Qi Liu, Yi Zhang, Weiwei Sun, Xinyu Ma, Wei Yang, Daiting Shi, Jiaxin Mao, and Dawei Yin. 2025. TourRank: Utilizing Large Language Models for Documents Ranking with a Tournament-Inspired Strategy. In *Proceedings of the ACM on Web Conference 2025* (Sydney NSW, Australia) *(WWW '25)*. Association for Computing Machinery, New York, NY, USA, 1638–1652. doi:10.1145/3696410.3714863

[6] Cheng-Han Chiang and Hung-yi Lee. 2023. Can Large Language Models Be an Alternative to Human Evaluations?. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, Toronto, Canada, 15607–15631. doi:10.18653/v1/2023.acl-long.870

[7] Nayoung Choi, Youngjune Lee, Gyu-Hwung Cho, Haeyu Jeong, Jungmin Kong, Saehun Kim, Keunchan Park, Sarah Cho, Inchang Jeong, Gyohee Nam, Sunghoon Han, Wonil Yang, and Jaeho Choi. 2024. RRADistill: Distilling LLMs' Passage Ranking Ability for Long-Tail Queries Document Re-Ranking on a Search Engine. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, Franck Dernoncourt, Daniel Preoţiuc-Pietro, and Anastasia Shimorina (Eds.). Association for Computational Linguistics, Miami, Florida, US, 627–641. doi:10.18653/v1/2024.emnlp-industry.46

[8] Guglielmo Faggioli, Laura Dietz, Charles L. A. Clarke, Gianluca Demartini, Matthias Hagen, Claudia Hauff, Noriko Kando, Evangelos Kanoulas, Martin Potthast, Benno Stein, and Henning Wachsmuth. 2023. Perspectives on Large Language Models for Relevance Judgment. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval* (Taipei, Taiwan) *(ICTIR '23)*. Association for Computing Machinery, New York, NY, USA, 39–50. doi:10.1145/3578337.3605136

[9] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Zhouchi Lin, Bowen Zhang, Lionel Ni, Wen Gao, Yuanzhuo Wang, and Jian Guo. 2026. A survey on LLM-as-a-judge. *The Innovation* (2026), 101253. doi:10.1016/j.xinn.2025.101253

[10] Zihan Hong, Yushi Wu, Zhiting Zhao, Shanshan Feng, Jianghong Ma, Jiao Liu, and Tianjun Wei. 2025. Multi-Objective Recommendation in the Era of Generative AI: A Survey of Recent Progress and Future Prospects. arXiv:2506.16893 [cs.IR] https://arxiv.org/abs/2506.16893

[11] Kasra Hosseini, Thomas Kober, Josip Krapac, Roland Vollgraf, Weiwei Cheng, and Ana Peleteiro Ramallo. 2025. Retrieve, Annotate, Evaluate, Repeat: Leveraging Multimodal LLMs for Large-Scale Product Retrieval Evaluation. In *Advances in Information Retrieval: 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6–10, 2025, Proceedings, Part I* (Lucca, Italy). Springer-Verlag, Berlin, Heidelberg, 149–163. doi:10.1007/978-3-031-88708-6_10

[12] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.* 20, 4 (Oct. 2002), 422–446. doi:10.1145/582415.582418

[13] Qi Liu, Haozhe Duan, Yiqun Chen, Quanfeng Lu, Weiwei Sun, and Jiaxin Mao. 2025. LLM4Ranking: An Easy-to-use Framework of Utilizing Large Language Models for Document Reranking. arXiv:2504.07439 [cs.IR] https://arxiv.org/abs/2504.07439

[14] Qi Liu, Atul Singh, Jingbo Liu, Cun Mu, and Zheng Yan. 2024. Towards More Relevant Product Search Ranking Via Large Language Models: An Empirical Study. arXiv:2409.17460 [cs.IR] https://arxiv.org/abs/2409.17460

[15] Tie-Yan Liu. 2009. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval* 3, 3 (2009), 225–331.

[16] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG Evaluation using Gpt-4 with Better Human Alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 2511–2522. doi:10.18653/v1/2023.emnlp-main.153

[17] Sean MacAvaney and Luca Soldaini. 2023. One-Shot Labeling for Automatic Relevance Estimation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 2230–2235. doi:10.1145/3539618.3592032

[18] Debabrata Mahapatra, Chaosheng Dong, Yetian Chen, and Michinari Momma. 2023. Multi-Label Learning to Rank through Multi-Objective Optimization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 4605–4616. doi:10.1145/3580305.3599870

[19] Debabrata Mahapatra, Chaosheng Dong, and Michinari Momma. 2023. Querywise Fair Learning to Rank through Multi-Objective Optimization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, New York, NY, USA, 1653–1664. doi:10.1145/3580305.3599482

[20] Yue Meng, Cheng Guo, Yi Cao, Tong Liu, and Bo Zheng. 2025. A Generative Re-ranking Model for List-level Multi-objective Optimization at Taobao. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Padua, Italy) *(SIGIR '25)*. Association for Computing Machinery, New York, NY, USA, 4213–4218. doi:10.1145/3726302.3731935

[21] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. RankVicuna: Zero-Shot Listwise Document Reranking with Open-Source Large Language Models. arXiv:2309.15088 [cs.IR] https://arxiv.org/abs/2309.15088

[22] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. RankZephyr: Effective and Robust Zero-Shot Listwise Reranking is a Breeze! *arXiv:2312.02724* (2023). https://arxiv.org/abs/2312.02724

[23] Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2024. Large Language Models are Effective Text Rankers with Pairwise Ranking Prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*, Kevin Duh, Helena Gomez, and Steven Bethard (Eds.). Association for Computational Linguistics, Mexico City, Mexico, 1504–1518. doi:10.18653/v1/2024.findings-naacl.97

[24] Rahul Raja, Arpita Vats, and Sudipta Roy. 2025. Aligning Prompts with Ranking Goals: A Technical Review of Prompt Engineering for LLM-Based Recommendations. *Preprints* (September 2025). doi:10.20944/preprints202509.1959.v1

[25] Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. 2024. ARES: An Automated Evaluation Framework for Retrieval-Augmented Generation Systems. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Kevin Duh, Helena Gomez, and Steven Bethard (Eds.). Association for Computational Linguistics, Mexico City, Mexico, 338–354. doi:10.18653/v1/2024.naacl-long.20

[26] Nilanjan Sinhababu, Andrew Parry, Debasis Ganguly, and Pabitra Mitra. 2025. Modeling Ranking Properties with In-Context Learning. arXiv:2505.17736 [cs.IR] https://arxiv.org/abs/2505.17736

[27] Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 14918–14937. doi:10.18653/v1/2023.emnlp-main.923

[28] Jie Tang, Huiji Gao, Liwei He, and Sanjeev Katariya. 2024. Multi-objective Learning to Rank by Model Distillation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Barcelona, Spain) *(KDD '24)*. Association for Computing Machinery, New York, NY, USA, 5783–5792. doi:10.1145/3637528.3671597

[29] Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2024. Large Language Models can Accurately Predict Searcher Preferences. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 1930–1940. doi:10.1145/3626772.3657707

[30] Kai Yuan, Anthony Zheng, Jia Hu, Divyanshu Sheth, Hemanth Velaga, Kylee Kim, Matteo Guarrera, Besim Avci, Jianhua Li, Xuetao Yin, Rajyashree Mukherjee, and Sean Suchter. 2026. Unifying Ranking and Generation in Query Auto-Completion via Retrieval-Augmented Generation and Multi-Objective Alignment. arXiv:2602.01023 [cs.IR] https://arxiv.org/abs/2602.01023

[31] Yang Zhao, Yixin Wang, and Mingzhang Yin. 2025. Permutative Preference Alignment from Listwise Ranking of Human Judgments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 310–334. doi:10.18653/v1/2025.emnlp-main.17

[32] Chujie Zheng, Jeffrey Wang, Shuqian Albee Zhang, Anand Kishore, and Siddharth Singh. 2024. Semantic Search Evaluation. arXiv:2410.21549 [cs.IR] https://arxiv.org/abs/2410.21549

[33] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (Eds.), Vol. 36. Curran Associates, Inc., 46595–46623. https://proceedings.neurips.cc/paper_files/paper/2023/file/91f18a1287b398d378ef22505bf41832-Paper-Datasets_and_Benchmarks.pdf

[34] Honglei Zhuang, Zhen Qin, Kai Hui, Junru Wu, Le Yan, Xuanhui Wang, and Michael Bendersky. 2024. Beyond Yes and No: Improving Zero-Shot LLM Rankers via Scoring Fine-Grained Relevance Labels. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, Kevin Duh, Helena Gomez, and Steven Bethard (Eds.). Association for Computational Linguistics, Mexico City, Mexico, 358–370. doi:10.18653/v1/2024.naacl-short.31

[35] Shengyao Zhuang, Honglei Zhuang, Bevan Koopman, and Guido Zuccon. 2024. A setwise approach for effective and highly efficient zero-shot ranking with large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 38–47.