Modeling Ranking Properties with In-Context Learning

Anonymous ACL submission

Abstract

While standard IR models are mainly designed to optimize relevance, real-world search often needs to balance additional objectives such as diversity and fairness. These objectives depend on inter-document interactions and are commonly addressed using post-hoc heuristics or supervised learning methods, which require task-specific training for each ranking scenario and dataset. In this work, we propose an in-context learning (ICL) approach that eliminates the need for such training. Instead, our method relies on a small number of example rankings that demonstrate the desired trade-offs between objectives for past queries similar to the current input. We evaluate our approach on four IR test collections to investigate multiple auxiliary objectives: group fairness (TREC Fairness), polarity diversity (Touché), and topical diversity (TREC Deep Learning 2019/2020). We empirically validate that our method enables control over ranking behavior through demonstration engineering, allowing nuanced behavioral adjustments without explicit optimization.

1 Introduction

Modern transformer-based language models are effective for ad-hoc ranking tasks (Karpukhin et al., 2020; Pradeep et al., 2023). By learning notions of relevance from sufficient training data, these approaches often outperform unsupervised rankers (Karpukhin et al., 2020; Formal et al., 2021). However, beyond the main objective of providing relevant content to a user, an IR system may have other auxiliary objectives, such as maximizing exposure fairness or topical diversity of documents (Carbonell and Goldstein, 2017). Different from relevance, which is an individual property of a document itself, these additional objectives, such as diversity (Clarke et al., 2008) or fair representation (Craswell et al., 2008), are instead properties of a top-retrieved list of documents, requiring effective modeling of *inter-document* interactions.

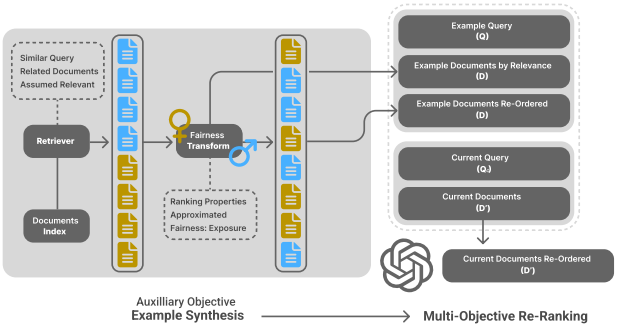

Figure 1: Proposed ICL method for reranking a set of top-retrieved documents. An example constitutes a localized query along with its top-retrieved arranged to satisfy a desired ranking property, such as relevance, fairness, diversity, etc.

Prior work on modeling ranking properties, such as diversity (Carbonell and Goldstein, 2017) involved interventions often in the form of ad hoc heuristic-based transformations of rankings (Agrawal et al., 2009). More recently, supervised approaches have been applied for learning neural interventions (MacAvaney et al., 2021), and incorporating inter-document interactions (Sun et al., 2023). A limitation of these supervised approaches is that they need to be trained on an adequate number of labeled examples for each different learning objective (Schlatt et al., 2024).

To alleviate this limitation, we explore the use of instruction-tuned generative language models (LLMs) for this task – the advantage being these models can potentially act as *universal ranking controllers* adjusting their behavior through prompt examples alone (Brown et al., 2020). While LLMs have been shown to be effective for zero-shot ranking tasks facilitated by interpreting natural language instructions (Sun et al., 2023; Pradeep et al., 2023), modeling listwise objectives is more challenging because of: firstly, the difficulty of expressing them in natural language, and secondly, optimizing a prompt instruction for each different auxiliary ranking objective. Such "prompt engi-

neering" is brittle (Ishibashi et al., 2023; Habba et al., 2025), requiring iterative refinement as objectives evolve. In contrast, we propose to use in-context learning (ICL) (Wei et al., 2022; Lu et al., 2024), where static instructions are paired with in-context examples of a desired ranking behaviour specific to a task. For instance, a single demonstration interleaving pro and con arguments could implicitly teach an LLM to diversify viewpoints, even with a task-agnostic instruction like "order by relevance". This approach circumvents the need for task-specific prompts while accommodating composite or dynamic objectives (Sinhababu et al., 2024). As shown in Figure 1, our method reranks candidate documents using LLMs conditioned on: (1) *localized (on-topic) query examples* (Sinhababu et al., 2024) from a large query log without requiring relevance assessments (Nguyen et al., 2016; Reimer et al., 2023) and (2) encoding ranking properties via *list-wise demonstrations*, e.g., diversity, fairness etc. Our main contributions are as follows:

- **A ICL-based approach that is shown to control desirable ranking properties without any supervised list-wise training.** We establish that demonstrations can change behaviour in a causal manner. Crucially, ablations confirm demonstrations as the causal factor: Inverting examples significantly degrades performance, affirming their role in adapting LLM behavior. These results provide evidence for demonstration-based model adaptation as a generalizable paradigm for dynamic ranking control.
- **Empirical validation of significantly improving several auxiliary objectives without sacrificing relevance.** Experiments on TREC DL (diversity) (Nguyen et al., 2016; Craswell et al., 2020b) and TREC Fairness 2022/Touché (fairness) (Ekstrand et al., 2022; Bondarenko et al., 2020) validate our approach. Unlike prior work that sacrifices relevance for auxiliary objectives (Zehlike and Castillo, 2020), our method maintains relevance while significantly improving diversity and fairness.

We provide our source code to facilitate future research and the reproducibility of our work[1].

2 Related Work

Ranking Models. Traditional term-weighting ranking models (Robertson et al., 1994) relied on exact lexical matches – a constraint later alleviated by neural approaches that leverage contextualized language representations for semantic soft matching (Karpukhin et al., 2020; Formal et al., 2021). Transformer-based architectures, such as cross-encoders (which jointly process query-document pairs) and bi-encoders (which map queries and documents to separate embeddings), emerged as dominant paradigms (Khattab and Zaharia, 2020; Schlatt et al., 2024). However, these models typically require task-specific fine-tuning via backpropagation to accommodate new objectives beyond generic relevance, limiting their adaptability. Our work diverges by eliminating gradient-based updates entirely, like other prior zero-shot neural ranking approaches (Li et al., 2023b; Sinhababu et al., 2024), enabling a flexible framework where rankings can dynamically satisfy diverse user or system-defined criteria, such as fairness or diversity, without requiring retraining.

Fairness. Prior work has emphasized the importance of balancing relevance with equitable exposure of document groups defined by attributes such as demographic origin, political stance, or gender (Zehlike et al., 2017; Morik et al., 2020). Biased exposure in rankings risks perpetuating systemic inequities, such as amplifying majority viewpoints while suppressing marginalized perspectives (Craswell et al., 2008; Ekstrand et al., 2019). Post-hoc fairness methods, including reranking algorithms (Morik et al., 2020; Biega et al., 2018) and fairness-aware loss functions (Singh and Joachims, 2018), explicitly redistribute exposure across groups but often degrade relevance (Pleiss et al., 2017; Corbett-Davies et al., 2017). Model-based approaches face challenges in defining target exposure distributions and incorporating them as an objective into a ranking loss (Heuss et al., 2022; Jänich et al., 2024). Furthermore, supervised methods struggle with biased training data (Zehlike and Castillo, 2020) and dependence on sensitive group labels, which are often incomplete or unavailable (Zehlike et al., 2017; Jänich et al., 2024). These limitations underscore the need for adaptable, training-free fairness mechanisms.

Diversity. Different from fairness, diversity in IR addresses ambiguous queries, reducing redundancy among retrieved results (Sen et al., 2022). Diverse search results are useful in representing multiple user intents or query subtopics (Carbonell and Goldstein, 2017; Clarke et al., 2008). Similar

[1] https://anonymous.4open.science/r/GPT_ranker-7099

to the fairness criteria, diversity also seeks to ensure balanced representation across various groups. However, different from fairness, the groups in diversity-based IR models correspond to facets or interpretations of a query (Ganguly et al., 2013).

Carbonell and Goldstein (2017) proposed maximal marginal relevance (MMR), which is a greedy algorithm that balances between the two objectives of: i) maximizing the similarity of a document with a query, and ii) favoring documents that are dissimilar to those already ranked higher. More recent methods infer latent query intents to synthesize diverse result sets across query intent clusters (Agrawal et al., 2009), or employ generative models such as IntenT5 to produce a variety of plausible query interpretations (MacAvaney et al., 2021). However, a limitation of these approaches is that they depend on post-hoc aggregation of intent-specific sub-rankings, which not only increases computational complexity but also often leads to a decrease in precision at top ranks (Wang and Zhu, 2009). In contrast, our method integrates diversity objectives directly into the ranking process by leveraging in-context examples, thereby avoiding the limitations of post-processing heuristics.

Generative Rankers. Generative rankers use autoregressive language models (LMs) to predict document permutations (Sun et al., 2023), bypassing traditional embedding-based or feature-centric paradigms. Recent advances in instruction-following LMs have enabled list-wise ranking, where models generate entire document orderings conditioned on a query (Sun et al., 2023; Ma et al., 2024). Unlike point-wise methods that score documents independently, list-wise approaches can condition relevance on inter-document dependencies, potentially capturing subtler interactions, e.g., topic coverage, redundancy (Schlatt et al., 2024). Sun et al. (2023) first demonstrated the effectiveness of list-wise ranking in a zero-shot setting, later adapted to smaller models via knowledge distillation (Pradeep et al., 2023). We extend this paradigm by integrating multi-objective control through in-context learning.

In-Context Learning (ICL). ICL enables task adaptation by conditioning models on demonstration examples without parameter updates. Beyond classification and question answering (Li et al., 2023a; Xu et al., 2024), ICL allows for better out-of-distribution generalisation in pairwise (Sinhababu et al., 2024) and list-wise ranking (Li et al., 2023b). Success hinges on selecting informative examples that encode task semantics and context (Nie et al., 2022; Sinhababu et al., 2024), with similarity-based example retrieval improving generalization (Xie et al., 2024). Different from existing ICL rankers that focus on relevance alone (Sinhababu et al., 2024), we propose reflecting other objectives, such as fairness, diversity, etc. into example rankings, thus allowing models to infer target criteria without explicit group labels, post-hoc adjustments, or multi-objective supervision.

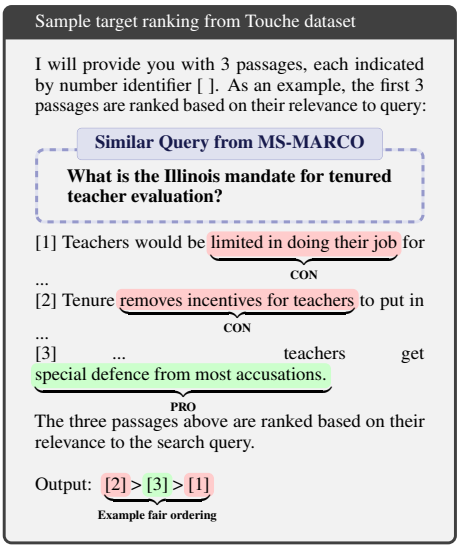

Figure 2: ICL Example for a Touche query. For this example, the target objective is to achieve a uniform distribution of the pro and the con arguments retrieved from the Touche collection. This figure shows the MS MARCO (train set) query - Q - which is the most similar to the current input query - Q^c. This figure shows how the documents retrieved for Q from the Touche collection are reranked to balance the pro:con ratio. This reranked list is added to the prompt as the example output.

3 ICL for Multi-Objective Ranking

Similar Queries from a Training Set. Given a test query Q_c (Figure 4 of Appendix D), the first step in our proposed workflow is to use a relatively large repository of existing queries $\mathcal{Q}$ to retrieve a set of k most *similar queries* - $\mathcal{N}_k(Q_c)$. For our experiments, we use the MS MARCO training set as our query reference set $\mathcal{Q}$ without labelled examples. We use BM25 as the initial retrieval model to obtain $\mathcal{N}_k(Q_c)$, and we set $k = 1$, i.e., we utilize only the top-retrieved query for subsequent steps; we call this query Q.

We then construct an **example ranking** to be used as additional context to condition the generative output of an LLM. To obtain this ICL example ranking, the top-retrieved query (i.e., Q) is used

to retrieve a set of top-m ranked documents from a target collection, say $\mathcal{T}$. We denote this top-retrieved list as $\theta(Q)_m = \{D_1, \ldots, D_m\}$, where θ is a retrieval model. The next step is to induce an ordering on the set of top-m documents $\theta(Q)_m$, i.e., $\theta(Q)_m \mapsto \langle D_{\pi(1)}, \ldots, D_{\pi(m)} \rangle$, where $\langle \cdot \rangle$ represents a sequence, and π denotes a permutation function over sets of m elements.

In the simplest case, the permutation function π corresponds only to the primary objective of relevance, i.e., maximizing the likelihood of positioning a relevant document ahead of a non-relevant one in the sequence. In practice, such information is available for a training set of queries (e.g., the MSMARCO train set), and prior work has shown that the inclusion of these examples improves pairwise ranking preferences (Sinhababu et al., 2024) - a special case of list-wise setting with the list size being 2. In addition to relevance, this permutation function can be designed to correspond to another auxiliary task thus allowing provision for a more general use-case, which we discuss next.

Target Distribution based Ranking. The auxiliary objective takes the general form of a categorical distribution involving k categories. More formally, $\forall D \in \theta(Q)_m$, let $A(D) \in \mathbb{Z}_k$ denote the value of the attribute A for document D. For instance, A may refer to the gender of an entity within a document, in which case, $k = 2$ and $A(D) \in \{$'Male', 'Female'$\}$.

For an IR task with an auxiliary objective involving an attribute A, a target distribution of these metadata values over the set of relevant documents is specified as a part of the input. For a query Q, we denote this distribution as $\tau(\mathcal{R}(Q)) \in \mathbb{R}^k$ (where k is the number of possible values). The i^{th} component of this vector is given by

$$\tau(\mathcal{R}(Q)_i) = \frac{\sum_{D \in \mathcal{R}(Q)} \mathbb{I}(A(D) = i)}{|\mathcal{R}(Q)|}, \quad (1)$$

where $(1 \leq i \leq k)$, which, in other words, represents the relative proportion of relevant documents for each metadata value ($\mathbb{I}$ denotes the indicator function).

Example 1 *Let us assume that for a query "architect" - out of 10 relevant documents in a collection, 6 are about male architects and the rest are about females. The target distribution for this example with '$A \equiv Gender$' is thus the 2d vector* $(0.6, 0.4)$ *as* $\sum_{D \in \mathcal{R}(Q)} \mathbb{I}(Gender(D) = M)/|\mathcal{R}(Q)| = 6/10$.

Grouping by metadata values. The aim is now to rerank the top-m documents in a way such that the aggregate of the relative proportion of the metadata values for different cutoffs $1 < m' < m$ is close to the target distribution. Relying on the retrieval similarity values as estimated probabilities of relevance, we apply a relatively simple approach to approximate this desired permutation π.

First, we partition the sequence of top-m documents (sorted in decreasing order by the retrieval scores as obtained with the retrieval model θ) into k different sequences:

$$\begin{aligned} \theta(Q)_m &= \cup_{i=1}^{k} \theta(Q)_m^{(i)} \\ \text{s.t. } \forall\, D \in \theta(Q)_m^{(i)} &\quad A(D) = i, \end{aligned} \quad (2)$$

where each $\theta(Q)_m^{(i)}$ denotes a subsequence of $\theta(Q)_m$ comprised of documents with a specific category value. See Example 2 for an illustration of how this step works.

Example 2 *For the query of Example 1, assume that 3 male and 2 female documents constitute the top-5 list:* $\langle M, M, F, M, F \rangle$*, where, for simplicity, we only show the attribute values instead of the cluttered notation* $A(D_1) = M$*. In this case, Equation 2 leads to partitioning the documents into two lists* $\theta(Q)_m^{(M)} = \langle D_1, D_2, D_4 \rangle$ *and* $\theta(Q)_m^{(F)} = \langle D_3, D_5 \rangle$.

Auxiliary Objective based Rank Induction. As a next step, we apply a greedy algorithm - somewhat similar in characteristic to maximum margin relevance (MMR) (Xia et al., 2015). However, different from the MMR diversity objective, the objective here is to maximize alignment with a target distribution of metadata values. More specifically, we consider only the yet unselected top documents from each group as candidates, and select the one that induces the distribution closest to the target distribution. Assuming that p documents are already selected in the reranked list, selection of the $(p+1)^{\text{th}}$ document depends on k different choices - one from each group. Let s_i denote the index of the document last selected from the i^{th} list, in which case the candidate documents available for selection during the $(p+1)^{\text{th}}$ iteration are: $\mathcal{C}_{p+1} = \{D_{s_1}, \ldots, D_{s_k}\}$. From these s_k alternatives, we select the document that leads to a distribution of top-$(p+1)$ documents that is closest to the target distribution, an example is illustrated

in Figure 2. More formally,

$$D_{p+1} = \underset{D \in \mathcal{C}_{p+1}}{\arg\min}\, \mathrm{KL}(\tau(\mathcal{R}(Q)), \tau(\langle D_1, \ldots, D_p \rangle \cup D), \quad (3)$$

where $\mathcal{C}_{p+1} = \{D_{s_1}, \ldots, D_{s_k}\}$ is the set of candidate documents available for selection during the $(p+1)^{\text{th}}$ iteration, and $\mathrm{KL}(X, Y)$ represents the KL divergence between two distributions X and Y, with $\tau(\mathcal{R}(Q))$ being the target distribution as defined in Equation 1. After making a selection (say from the j^{th} list, i.e., $D_{p+1} = D_{s_j}$), we increment the corresponding index by 1 (i.e., $s_j \leftarrow s_j + 1$) to point to the next candidate available for selection. See Example 3 to see an illustration of how this greedy selection algorithm works on the data shown in Examples 1 and 2.

Example 3 *After selecting the first document D_1, the two choices available for the second selection (shown underlined) are: $\langle D_1, \underline{D_2}, D_4 \rangle$ and $\langle \underline{D_3}, D_5 \rangle$. Selecting D_2 means that the distribution over the top-2 documents ($\langle D_1, D_2 \rangle$) is (2, 0), whereas selecting D_3 (a female document) yields the distribution (1, 1). Since the latter is closer to the target distribution of (0.6, 0.4), we select D_3 as per Equation 3. After incrementing the selection index, the candidates available for the next step are: $\langle D_1, \underline{D_2}, D_4 \rangle$ and $\langle D_3, \underline{D_5} \rangle$. Following the same argument, applying the selection function of Equation 3 two more times, we obtain the desired ranking of top-5 documents that are most similar to the target distribution, which in this case is: $\langle D_1, D_3, D_2, D_5, D_4 \rangle$.*

The target distribution-driven reranked documents obtained by an iterative application of the greedy selection function of Equation 3 then act as the ICL examples shown in the prompt of Figure 4.

4 Evaluation

We now provide empirical evidence for our approach, structured around four research questions.

4.1 Research Questions and Setup

Our first research question explores the benefits of ICL examples, i.e., **(RQ-1)**: *Is our proposed ICL-based list-wise ranker consistently effective across a range of different tasks involving different types of attributes and target distributions*?

Our second research question contrasts our approach with the direct prompting of a language model to rank by multiple objectives, as opposed to implicitly providing two objectives through examples, i.e., **(RQ-2)**: *How do ICL examples compare to directly instructing a model in terms of auxiliary objective effectiveness*?

In supervised learning, the domain and distribution of inputs should generally match our test instances where possible (Gutmann and Hyvärinen, 2010). Learning-to-rank literature indicates that input rankings should match the first stage at test time (Macdonald et al., 2013). We look to validate to what degree this statement holds for in-context learning. Explicitly, **(RQ-3)**: *What are the effects of ICL example ranking strategies that are adversarial to a target distribution-based auxiliary objective*?

Objectives and Datasets. Our investigation is conducted over two auxiliary objectives: diversity and group fairness. For diversity, we aim to retrieve relevant but topically diverse content to satisfy multiple potential information needs under ambiguity. To operationalize this, we adopt the experimental framework of Schlatt et al. (2024) using the MS-MARCO passage corpus (Nguyen et al., 2016) and TREC Deep Learning 2019–2020 (Craswell et al., 2020b,a) test collections. In contrast to fairness, where group labels in relevance judgments inform the target distribution, diversity assumes a uniform distribution over latent topics, derived via clustering over retrieved documents. Additional details on the clustering procedure are given in Appendix A.2, and full dataset descriptions are provided in Appendix A.1. We also evaluate group fairness in the single-ranking setting using TREC Fairness 2022 (Ekstrand et al., 2022) and Touche (Bondarenko et al., 2020), both of which include explicit group attributes (gender and stance, respectively). In the Touche setting, we reformulate the task to seek balanced representation of PRO and CON arguments. The motivation in both cases is the promotion of equitable and unbiased outputs in re-ranking models.

To enable test-time control of retrieval behavior, we retrieve similar queries from the MSMARCO training set (approx. 8×10^5 entries), which serve as anchors for constructing contextual examples. Unlike Sinhababu et al. (2024), our approach assumes no relevance labels during example selection, but following the findings of Sinhababu et al. (2024), we fix $k = 1$ as gains beyond this value were found to be minimal, so we leave further parameter ablations to future work in which a single

example may be insufficient. Additional implementation details, including similarity metrics and retrieval configurations, can be found in Appendix A.

4.2 Baselines

Prompt-based Auxiliary Objective (PAO). This is a 0-shot list-wise ranking baseline that uses explicit instructions to elicit fairness or diversity from the model. The prompt for fairness is: "*Rank the passages based on their fairness, ensuring that ranked results do not discriminate against certain individuals, groups, or entities*", whereas for diversity, it is: "*Rank the passages based on their topical diversity, ensuring that ranked results contribute to different topics uniformly*".

Baselines for Diversity. For the diversity-ranking task, we employ the following baselines.

- **Max Margin Relevance (MMR)** (Carbonell and Goldstein, 2017): Combines relevance and diversity via a linear combination. Following Zhu et al. (2014), the mixture weight was tuned with 5-fold cross-validation.
- **Set-Encoder (SEN)** (Schlatt et al., 2024): A cross-encoder model trained with diversity-aware loss. We use `set-encoder-large`[2], fine-tuned on MS MARCO.

Baselines for Fairness. For the fair-ranking task, we employ the following.

- **FA*IR**[3] (Zehlike et al., 2017): A post-processing method to enforce fair exposure. The two hyper-parameters of FA*IR, namely a) α: the proportion of protected candidates, and b) p: the significance level, were tuned via 10-fold cross-validation over $\alpha \in [0.01, 0.13]$ and $p \in [0.4, 0.85]$ with step sizes of 0.01. The optimal values were found to be $(\alpha, p) = (0.1, 0.2)$.
- **DELTR** (Zehlike and Castillo, 2020): A learning-to-rank model[4] trained on fair rankings created using FA*IR. We use our example sets to simulate fair supervision. We set a high value on the trade-off parameter γ=1, to ensure that it promotes fairness without impacting the overall ranking utility (Zehlike and Castillo, 2020).

Measures. To evaluate ranking quality, we report nDCG@10 for relevance, AWRF and nDCG-AWRF combination (M1) for fairness as per Ekstrand et al. (2022), and αnDCG@10 with $\alpha = 1$ and for diversity as per Clarke et al. (2008). All these metrics are reported at a cutoff of 10.

Models. We re-rank the top 100 documents using a two-stage pipeline. We apply BM25 (Robertson et al., 1994) and ColBERT (Khattab and Zaharia, 2020) as initial rankers. The second-stage ranker is our proposed ICL-based one. As re-ranking 100 passages directly degrades effectiveness (Schlatt et al., 2025), we apply a sliding window-based reranking over the top 100 documents. The window size was set to 20, and the stride size to 10 as commonly applied in literature (Sun et al., 2023; Pradeep et al., 2023). As the underlying LLM for reranking with ICL, we experimented with the larger closed-source GPT-4o-mini and smaller open-weighted Llama-3.1 (7/70B). Details on model configurations, hyperparameters, and implementation are provided in Appendix A.

Ablations. To validate that the target distribution plays an important role in our proposed ICL-based reranking, we experiment with the following mechanisms of other reranking objectives.

- **Adversarial Examples**: By swapping the proportions, we seek to maximize the KL divergence from the target distribution (instead of minimizing it as per Equation 3), e.g., flipping a 3:2 gender ratio to 2:3.
- **Uniform Examples**: Construct examples that enforce uniform distribution across clusters or attributes, ignoring relevance information.
- **Relevant Examples**: Solely make a transformation from a random order to an order by relevance determined by a first-stage system.
- **Static Examples**: Replace similar query examples with a fixed example ranking used for all test queries, which implies no shared topicality. The purpose is to explore whether topicality is important for effective ICL example rankings.

4.3 Findings

Examples allow for the modeling of multiple objectives. Our experiments demonstrate that in-context learning (ICL) with task-guided examples enables effective optimization of auxiliary objectives while maintaining relevance (**RQ-1**). For diversity modeling (Table 1), our approach significantly outperforms the 0-shot baseline in topical diversity (αnDCG), with improvements of up to 19% (e.g., compare rows 5 and 13 with those of 7

[2] https://github.com/webis-de/set-encoder

[3] https://github.com/fair-search/fairsearch-fair-python

[4] https://github.com/fair-search/fairsearch-deltr-python

Table 1: Evaluating nDCG and αnDCG performance over TREC DL-2019 and 2020 for the Diversity objective. The maximum score in each category is represented in bold font. Symbols $\star$ and † indicate the statistical significance of our proposed model with the first-stage and 0-shot baselines, respectively (paired t-test with $p = 0.05$).

			TREC DL-2019		TREC DL-2020	
Type		Pipeline	nDCG	αnDCG	nDCG	αnDCG
Baselines	*1*	BM25	.4795	.4569	.4936	.4895
	2	+ PAO	.6817	.6844	.6349	.6670
	3	+ MMR	.4786	.4559	.4922	.4899
	4	+ 0-shot(Llama)	.5977	.5695	.5971	.6357
	5	+ 0-shot(GPT)	.6971	.6761	.6826	.7039
ICL	*6*	+ Diverse(Llama)	.6144	.5968	.5983	.6062
	7	+ Diverse(GPT)	**.7124***	**.7135***†	**.6844***	**.7228***
Baselines	*8*	ColBERT	.7205	.6583	.6864	.6385
	9	+ PAO	.6949	.6494	.6996	.6910
	10	+ MMR	.7173	.6567	.6873	.6405
	11	+ SEN	.7320	.6172	.7245	.6338
	12	+ 0-shot(Llama)	.7363	.6595	.7044	.6622
	13	+ 0-shot(GPT)	**.7699**	.6850	.7498	.6843
ICL	*14*	+ Diverse(Llama)	.7116	.6527	.6976	.6443
	15	+ Diverse(GPT)	.7601	**.6891***	**.7700**	**.7132***†

Table 2: Evaluating AWRF, nDCG, and M1 over Touche-2020 (PRO,CON) and TREC Fair-2022 (M,F) for the Fairness objective, other details are analogous those of Table 1.

			Touche-2020			Fair-2022		
Type		Pipeline	nDCG	AWRF	M1	nDCG	AWRF	M1
Baselines	*1*	BM25	.2530	.4811	.1851	.4974	.4901	.2975
	2	+ PAO	.2258	.5218	.1589	.5667	.5312	.3332
	3	+ FA*IR	.2452	.4620	.1660	.3735	**.7215**	**.3989**
	4	+ DELTR	.2486	.3212	.1190	.3786	.5593	.3220
	5	+ 0-shot(Llama)	.2388	.4821	.1748	.5658	.4895	.3228
	6	+ 0-shot(GPT)	.2590	.5377	.1936	.5688	.5494	.3428
ICL	*7*	+ Fair(Llama)	.2136	.5199	.1486	.5316	.5300	.3159
	8	+ Fair(GPT)	**.2608**	**.5800***†	**.2023***	**.5697**	.5697*	.3526
Baselines	*9*	ColBERT	**.2590**	.2994	.1462	.4854	.2068	.1204
	10	+ PAO	.2234	.2388	.0956	.6565	.1837	.1411
	11	+ FA*IR	.2500	**.3598**	**.1698**	.2111	**.5896**	**.3320**
	12	+ DELTR	.2518	.2216	.1078	.2128	.3370	.2215
	13	+ 0-shot(Llama)	.2344	.2588	.1169	.5870	.1247	.0917
	14	+ 0-shot(GPT)	.2496	.2197	.1027	.6487	.2056	.1466
ICL	*15*	+ Fair(Llama)	.2089	.2389	.0960	.5183	.2069	.1141
	16	+ Fair(GPT)	.2508	.2602†	.1216	**.6606**	.2272	.1628*

and 15). Notably, it surpasses both post-hoc diversification (MMR) and a supervised list-wise method (SEN) by 8-15% in αnDCG (Rows 3, 10-11), despite relying only on heuristic examples rather than explicit optimization. These results indicate that example-based task conditioning provides a sufficient learning signal for the model to acquire objective-specific ranking behaviors.

For the smaller re-ranker (LlaMA-8B), incorporating ICL examples yields an improvement in αnDCG over the 0-shot setting on DL-19 when using BM25 (Rows 4 vs. 6), though it still underperforms relative to all other baselines. When applied to a smaller model, diversity-oriented examples may inadvertently introduce less relevant or random items, negatively impacting relevance and diversity metrics. In contrast, larger models appear more capable of integrating both the examples' diversity cues and the instructions' relevance constraints, likely due to their greater representational capacity.

Table 2 shows results for the auxiliary objective of group fairness. Our approach exceeds all baselines on Touche and is competitive with post-processing and supervised methods on Fair-2022. In experiments with the smaller model, we observe a similar outcome except for Touche under ColBERT, as seen from Table 2 (Rows 13 vs 15), suggesting that even smaller LLMs are effective at modeling an auxiliary objective. The larger model, however, significantly outperforms the smaller model across most scenarios.

Using the larger model (GPT-4o-mini), our method achieves approximately a 52% improvement in nDCG performance compared to DELTR and FA*IR, as shown by comparing Rows 3 and 8 in Table 2. However, this gain in effectiveness is accompanied by a reduction in fairness on the Fair-2022 dataset. Table 2 also reveals a sensitivity to the choice of first-stage retriever: BM25 yields better fairness outcomes than ColBERT across evaluation settings. Notably, under the ColBERT retriever, our approach shows diminished effectiveness on Touche-2022, as seen in the performance drop between Rows 9 and 16. This trend aligns with prior findings (Tang et al., 2024; Parry et al., 2024b), highlighting that weaker first-stage rankings tend to impair the effectiveness of list-wise re-ranking. Interestingly, under a diversity-oriented objective, we observe the opposite pattern: stronger first-stage rankings result in reduced diversity effectiveness. We analyze these interactions, particularly the role of positional bias, in detail in Appendix C.

Nevertheless, our approach with ICL outperforms the 0-shot setting. With BM25 as the first stage ranker, we outperform all but one baseline FA*IR on Fair-2022, which we parameter-tuned with a grid search. Under ColBERT, we observe that FA*IR is most effective across both datasets. However, our approach outperforms all other baselines but is at near-parity with 0-shot, suggesting a minimal change in the model's process. The FA*IR approach requires prior group information for ranking and poses a significant trade-off in relevance,

thereby limiting its practical applicability. In contrast, our approach addresses these issues with a simple yet effective solution.

In-context learning improves ranking performance over direct instructions. To address **RQ-2**, we examine the impact of prompt-based optimization on fairness and diversity objectives. As shown in Tables 1 and 2, our ICL approach is more effective than the Prompt-based Auxiliary Objective (PAO) baseline when applied to the larger model. However, this improvement does not extend to the smaller model, which fails to outperform PAO in most cases. These findings highlight the importance of model capacity in effectively leveraging prompt-based optimization techniques for complex ranking objectives.

Relative to the GPT 0-shot baseline, the PAO method exhibits a notable decline in nDCG, with the sole exception occurring under the ColBERT retriever on the Fair-2022 dataset. This reduction in ranking utility is likely attributable to the modified prompt, which explicitly instructs the model to enhance diversity–an objective combination that may be out-of-distribution for the model. This observation further underscores the advantages of demonstration-based adaptation. We observe marginal improvements in αnDCG with PAO on DL-2019 under BM25 and on DL-2020 under ColBERT (Table 1, Rows 2 and 9). In terms of AWRF, a positive gain is observed only for Touche when ColBERT is used as the first-stage retriever (Table 2, Row 10). We attribute the inconsistency in PAO performance to the lack of explicit contextual grounding regarding the fairness or diversity criteria being targeted, as opposed to the model's default interpretation of these objectives.

Example inversion largely degrades effectiveness. From Table 3, we see that examples modeled with relevance yield significant improvements in nDCG relative to both BM25 and ColBERT. This shows that ICL examples, in addition to modeling auxiliary objectives, can also effectively capture relevance. From Rows 4 and 10, we observe a consistent degradation in terms of αnDCG when applying a uniform example (the adversarial setting for the diversity task). This suggests that the target distribution induced ordering plays an important role in ranking.

Additionally, we observe that ICL examples exhibit minimal trade-off in terms of relevance, suggesting that fine-grained control can be exerted without compromising the core task except for static examples as observed from Rows 6 and 12 of Table 3. Static examples cause a substantial decline in the evaluation scores, occasionally falling below those of the base ranker. This validates that the locality of queries in ICL examples remains useful in a list-wise setting, as was observed by Sinhababu et al. (2024) in a pair-wise setting.

Table 3: Ablations (Adv, Rel, and Static) on the diversity and fairness tasks show that ICL examples with different objectives have a significant impact on the ranking. The "+Target" row corresponds to the target-distribution specific ICL re-ranking results, which have been presented in Tables 1 and 2. Suffixes *a* to *e* represent the statistical significance of "+Target" when compared to the first-stage, 0-shot, Adversarial (Adv), Relevant (Rel), and Static methods, respectively, computed via paired t-test with $p = 0.05$.

	DL-2019		DL-2020		Touche-2020		Fair-2022	
Method	nDCG	αnDCG	nDCG	αnDCG	nDCG	AWRF	nDCG	AWRF
BM25	.479	.457	.494	.489	.253	.481	.497	.490
+ 0-shot	.697	.676	.683	.704	.259	.537	.569	.549
+Target	**.712**^{a}	**.713**^{abe}	.684^{a}	**.729**^{ac}	**.261**	**.580**^{abcde}	.570	**.570**^{a}
+Adv	.696	.688	.686	.701	.260	.520	.572	.550
+Rel	.700	.685	**.692**	.712	.255	.524	**.581**	.548
+Static	.704	.680	.682	.709	.251	.480	.570	.550
ColBERT	.720	.658	.686	.638	**.259**	**.299**	.485	.207
+0-shot	**.770**	.685	.750	.684	.250	.220	.649	.206
+Target	.760	**.689**^{ac}	**.770**	**.713**^{abe}	.251	.260^{bcde}	**.661**	**.227**^{ce}
+Adv	**.770**	.686	.761	.706	.253	.197	.656	.185
+Rel	.764	.682	.762	.710	.245	.217	.646	.185
+Static	.760	.687	.755	.698	.241	.181	.648	.185

5 Conclusion and Future Work

We propose a novel approach to multi-objective ranking leveraging demonstrations that balance relevance and auxiliary objectives. Our experiments confirm that localized examples modeled for fairness and diversity improve the respective objectives significantly without compromising relevance. We validate each component of our approach, for instance, finding that effectiveness gains can be controlled with adversarial examples, degrading the fairness of downstream rankings. Additionally, our approach improves over directly instructing a model for each objective. Our approach demonstrates superior performance compared to task-specific post-hoc and supervised methods, both in evaluation metrics and practical applicability, while effectively mitigating the potential trade-offs and the need for task-specific modifications. Furthermore, our findings present encouraging evidence for demonstration-based model adaptation as a mechanism for controlling ranking behaviour beyond the objectives investigated in this work.

Ethics Statement. Though our work primarily focuses on augmenting core ranking tasks, one could, in principle, use our approach to induce more harmful behaviour within a model. As no explicit instruction change occurs, this may allow the bypassing of guardrails, as harmful behaviour could be demonstrated. Nevertheless, such approaches are common as are their mitigations, and our work does not explicitly facilitate such applications more broadly.

Limitations. We do not explore all possible avenues for demonstration-based multi-objective search in this work. Indeed, several parameter choices are motivated by prior work; however, due to our novel setting, it could be that under this new setting, effectiveness could be further improved. Our approach requires an existing query log, which in low information environments or low resource languages may present difficulties in adopting our approach. In future work, we look to rectify the need for a monolingual corpus.

References

Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. 2009. Diversifying search results. In *Proceedings of the Second International Conference on Web Search and Web Data Mining, WSDM 2009, Barcelona, Spain, February 9-11, 2009*, pages 5–14. ACM.

Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. 2018. Equity of attention: Amortizing individual fairness in rankings. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 405–414. ACM.

Alexander Bondarenko, Maik Fröbe, Meriem Beloucif, Lukas Gienapp, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. 2020. Overview of touché 2020: Argument retrieval: Extended abstract. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association, CLEF 2020, Thessaloniki, Greece, September 22–25, 2020, Proceedings*, page 384–395, Berlin, Heidelberg. Springer-Verlag.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Jaime G. Carbonell and Jade Goldstein. 2017. The use of mmr, diversity-based reranking for reordering documents and producing summaries. *SIGIR Forum*, 51(2):209–210.

Charles L. A. 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, SIGIR 2008, Singapore, July 20-24, 2008*, pages 659–666. ACM.

Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. 2017. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pages 797–806. ACM.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2020a. Overview of the TREC 2020 deep learning track. In *Proceedings of the Twenty-Ninth Text REtrieval Conference, TREC 2020, Virtual Event [Gaithersburg, Maryland, USA], November 16-20, 2020*, volume 1266 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020b. Overview of the TREC 2019 deep learning track. *CoRR*, abs/2003.07820.

Nick Craswell, Onno Zoeter, Michael J. Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *Proceedings of the International Conference on Web Search and Web Data Mining, WSDM 2008, Palo Alto, California, USA, February 11-12, 2008*, pages 87–94. ACM.

Michael D. Ekstrand, Robin Burke, and Fernando Diaz. 2019. Fairness and discrimination in retrieval and recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pages 1403–1404. ACM.

Michael D. Ekstrand, Graham McDonald, Amifa Raj, and Isaac Johnson. 2022. Overview of the TREC 2022 fair ranking track. In *Proceedings of the Thirty-First Text REtrieval Conference, TREC 2022, online, November 15-19, 2022*, volume 500-338 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. SPLADE: sparse lexical and expansion model for first stage ranking. In *SIGIR '21: The 44th International ACM SIGIR Conference on*

Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, pages 2288–2292. ACM.

Debasis Ganguly, Manisha Ganguly, Johannes Leveling, and Gareth J. F. Jones. 2013. Topicvis: a GUI for topic-based feedback and navigation. In *The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13, Dublin, Ireland - July 28 - August 01, 2013*, pages 1103–1104. ACM.

Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pages 297–304. JMLR.org.

Eliya Habba, Ofir Arviv, Itay Itzhak, Yotam Perlitz, Elron Bandel, Leshem Choshen, Michal Shmueli-Scheuer, and Gabriel Stanovsky. 2025. DOVE: A large-scale multi-dimensional predictions dataset towards meaningful LLM evaluation. *CoRR*, abs/2503.01622.

Maria Heuss, Fatemeh Sarvi, and Maarten de Rijke. 2022. Fairness of exposure in light of incomplete exposure estimation. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 759–769. ACM.

Yoichi Ishibashi, Danushka Bollegala, Katsuhito Sudoh, and Satoshi Nakamura. 2023. Evaluating the robustness of discrete prompts. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 2365–2376. Association for Computational Linguistics.

Thomas Jänich, Graham McDonald, and Iadh Ounis. 2024. Fairness-aware exposure allocation via adaptive reranking. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 1504–1513. ACM.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 39–48. ACM.

Juhi Kulshrestha, Motahhare Eslami, Johnnatan Messias, Muhammad Bilal Zafar, Saptarshi Ghosh, Krishna P. Gummadi, and Karrie Karahalios. 2017. Quantifying search bias: Investigating sources of bias for political searches in social media. In *arXiv preprint arXiv:1704.01347*, volume abs/1704.01347.

Tianle Li, Xueguang Ma, Alex Zhuang, Yu Gu, Yu Su, and Wenhu Chen. 2023a. Few-shot in-context learning on knowledge base question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 6966–6980. Association for Computational Linguistics.

Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. 2023b. Unified demonstration retriever for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 4644–4668. Association for Computational Linguistics.

Sheng Lu, Irina Bigoulaeva, Rachneet Sachdeva, Harish Tayyar Madabushi, and Iryna Gurevych. 2024. Are emergent abilities in large language models just in-context learning? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 5098–5139. Association for Computational Linguistics.

Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2024. Fine-tuning llama for multi-stage text retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 2421–2425. ACM.

Sean MacAvaney, Craig Macdonald, Roderick Murray-Smith, and Iadh Ounis. 2021. Intent5: Search result diversification using causal language models. *CoRR*, abs/2108.04026.

Craig Macdonald, Rodrygo L. T. Santos, and Iadh Ounis. 2013. The whens and hows of learning to rank for web search. *Inf. Retr.*, 16(5):584–628.

Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. 2020. Controlling fairness and bias in dynamic learning-to-rank. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 429–438. ACM.

David Mueller, Mark Dredze, and Nicholas Andrews. 2024. Multi-task transfer matters during instruction-tuning. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and*

virtual meeting, August 11-16, 2024, pages 14880–14891. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Feng Nie, Meixi Chen, Zhirui Zhang, and Xu Cheng. 2022. Improving few-shot performance of language models via nearest neighbor calibration. *CoRR*, abs/2212.02216.

Andrew Parry, Sean MacAvaney, and Debasis Ganguly. 2024a. Exploiting positional bias for query-agnostic generative content in search. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 11030–11047. Association for Computational Linguistics.

Andrew Parry, Sean MacAvaney, and Debasis Ganguly. 2024b. Top-down partitioning for efficient list-wise ranking. *CoRR*, abs/2405.14589.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon M. Kleinberg, and Kilian Q. Weinberger. 2017. On fairness and calibration. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5680–5689.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. Rankzephyr: Effective and robust zero-shot listwise reranking is a breeze! *CoRR*, abs/2312.02724.

Jan Heinrich Reimer, Sebastian Schmidt, Maik Fröbe, Lukas Gienapp, Harrisen Scells, Benno Stein, Matthias Hagen, and Martin Potthast. 2023. The archive query log: Mining millions of search result pages of hundreds of search engines from 25 years of web archives. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 2848–2860. ACM.

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. In *Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, November 2-4, 1994*, volume 500-225 of *NIST Special Publication*, pages 109–126. National Institute of Standards and Technology (NIST).

Ferdinand Schlatt, Maik Fröbe, Harrisen Scells, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, Benno Stein, Martin Potthast, and Matthias Hagen. 2024. Set-encoder: Permutation-invariant inter-passage attention for listwise passage re-ranking with cross-encoders. *CoRR*, abs/2404.06912.

Ferdinand Schlatt, Maik Fröbe, Harrisen Scells, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, Benno Stein, Martin Potthast, and Matthias Hagen. 2025. Rank-distillm: Closing the effectiveness gap between cross-encoders and llms for passage re-ranking. In *Advances in Information Retrieval - 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6-10, 2025, Proceedings, Part III*, volume 15574 of *Lecture Notes in Computer Science*, pages 323–334. Springer.

Procheta Sen, Debasis Ganguly, and Gareth J. F. Jones. 2022. I know what you need: Investigating document retrieval effectiveness with partial session contexts. *ACM Trans. Inf. Syst.*, 40(3):53:1–53:30.

Ashudeep Singh and Thorsten Joachims. 2018. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, pages 2219–2228. ACM.

Nilanjan Sinhababu, Andrew Parry, Debasis Ganguly, Debasis Samanta, and Pabitra Mitra. 2024. Few-shot prompting for pairwise ranking: An effective non-parametric retrieval model. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 12363–12377. Association for Computational Linguistics.

Taylor Sorensen, Joshua Robinson, Christopher Michael Rytting, Alexander Glenn Shaw, Kyle Jeffrey Rogers, Alexia Pauline Delorey, Mahmoud Khalil, Nancy Fulda, and David Wingate. 2022. An information-theoretic approach to prompt engineering without ground truth labels. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 819–862. Association for Computational Linguistics.

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, EMNLP 2023, Singapore, December 6-10, 2023*, pages 14918–14937. Association for Computational Linguistics.

Raphael Tang, Xinyu Crystina Zhang, Xueguang Ma, Jimmy Lin, and Ferhan Ture. 2024. Found in the middle: Permutation self-consistency improves listwise ranking in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 2327–2340. Association for Computational Linguistics.

Jun Wang and Jianhan Zhu. 2009. Portfolio theory of information retrieval. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009*, pages 115–122. ACM.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Trans. Mach. Learn. Res.*, 2022.

Long Xia, Jun Xu, Yanyan Lan, Jiafeng Guo, and Xueqi Cheng. 2015. Learning maximal marginal relevance model via directly optimizing diversity evaluation measures. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, pages 113–122. ACM.

Tingyu Xie, Qi Li, Yan Zhang, Zuozhu Liu, and Hongwei Wang. 2024. Self-improving for zero-shot named entity recognition with large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Short Papers, NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 583–593. Association for Computational Linguistics.

Hongling Xu, Qianlong Wang, Yice Zhang, Min Yang, Xi Zeng, Bing Qin, and Ruifeng Xu. 2024. Improving in-context learning with prediction feedback for sentiment analysis. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 3879–3890. Association for Computational Linguistics.

Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. 2017. Fa*ir: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*, pages 1569–1578. ACM.

Meike Zehlike and Carlos Castillo. 2020. Reducing disparate exposure in ranking: A learning to rank approach. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 2849–2855. ACM / IW3C2.

Yadong Zhu, Yanyan Lan, Jiafeng Guo, Xueqi Cheng, and Shuzi Niu. 2014. Learning for search result diversification. In *The 37th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '14, Gold Coast , QLD, Australia - July 06 - 11, 2014*, pages 293–302. ACM.

Table 4: Statistics of the datasets used in our experiments.

Task	Collection	$\lvert\mathcal{C}\rvert$	Queries	$\lvert\mathcal{Q}\rvert$	Name	Values
Fairness	TREC Fair	6.5M	Fair-2022	50	Gender	M, F
	ToucheV2	383K	Touche'20	49	Stance	PRO, CON
Diversity	MS MARCO	8.8M	DL'19	43	Topic	$\mathbb{Z}$
			DL'20	54		

A Additional Experimental Details

A.1 Dataset Description

The fairness task corresponds to that of the 'single ranking' task of TREC Fairness track (Ekstrand et al., 2022) on the 'eval' query set. The objective in the Touche task is to mitigate the bias towards a specific stance (Kulshrestha et al., 2017), whereas the objective in the ad-hoc search task on TREC DL topics is to maximize the topical diversity, where each topic maps to a cluster of documents. We illustrate the details of our chosen collections in Table 4.

A.2 Clustering for Diversity

To induce topic clusters for diversity evaluation, we apply hierarchical agglomerative clustering with complete linkage over Jaccard similarity between token sets. This is applied to the top 100 documents retrieved per query. Due to the nature of agglomerative clustering, the number of clusters varies with query specificity, resulting in a query-dependent target distribution.

A.3 Query Similarity Retrieval

Following Sinhababu et al. (2024), we retrieve similar queries from the MSMARCO training split using BM25 over query text. We retrieve the top-5 most similar queries and aggregate their retrieved documents to build example sets for in-context learning. No relevance judgments are used in this process.

A.4 LLM Configurations

- **Llama-3.1-8B-Instruct**: 8B decoder-only LLM[5] with 8K context length, sufficient for in-context example ranking. We use the "`text-generation`" pipeline with the standard `bfloat16` as it is the recommended way to conduct evaluations. We use the default parameters for the rest of the experiments: `do_sample=True`, `temperature=0.6` and `top_p=0.9`. Additionally, we set

[5] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

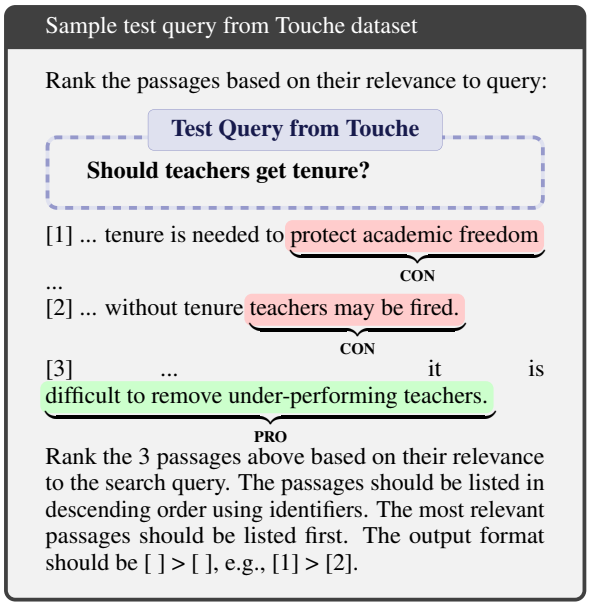

Figure 3: The figure shows a sample input query from the Touche dataset. The ICL example of a related query from MS MARCO and its example output (balancing both relevance and pro:con parity, as shown in Figure 2) is used to *control* the current query's reranking.

`seed=42` for all our experiments. We perform these experiments locally using a single NVIDIA A100 (40GB) GPU.

- **GPT-4o-mini**: Used as the primary re-ranking model[6]. During inference we set the following hyperparameters: `temparature=0`, `return_text=True` and `seed=42`. Contamination concerns are minimal since the model is not optimized for improved test scores but for behavioral modulation under auxiliary objectives.

- **Llama-3.1-70B-Instruct**: 70B decoder-only LLM[7] with 8K context length to test behavior of targeted ICL with larger model size. We use an API service due to hardware limitations of using the full precision model locally. We use the exact same parameters as detailed in `Llama-3.1-8B-Instruct`.

We refrain from using rank instruction-tuned models, as these models tend to exhibit greater sensitivity to prompt formatting and catastrophic forgetting of general task knowledge in ICL setups (Mueller et al., 2024).

[6] https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

[7] https://deepinfra.com/meta-llama/Meta-Llama-3.1-70B-Instruct

B Effect of LLM variations

We included test results with Llama-70B alongside Llama-8B and GPT-4o-mini to answer if and how our approach is dependent on LLM size. As observed from Table 5, we mark that GPT-4o-mini consistently outperformed all other models when considering the target task using ICL examples, with the only exception being TREC DL-2020. Llama-8B results show that our approach works even for small models, but with limited gains both in terms of relevance and auxiliary objective. In contrast to Llama-8B, we observe that Llama-70B is strong in the relevance task; however, it does not show significant improvements in auxiliary objectives with the ICL examples. This suggests that our approach provides limited gains when the model is already superior in terms of diverse rankings. Nevertheless, under settings such as fairness, which are generally detrimental to ranking effectiveness, we can further improve and maintain nDCG.

C Example ordering

We initially adopted random ordering for in-context examples due to its simplicity and computational efficiency. However, prior work highlights that the ordering of both examples and test documents can critically influence model behavior, introducing instability or performance degradation in tasks like ranking and generation (Sorensen et al., 2022; Parry et al., 2024a). To systematically evaluate this risk, we complement random ordering with examples ordered by first-stage ranker scores (e.g., BM25 or ColBERT relevance scores). This dual approach tests whether example ordering impacts model outputs analogously to test document ordering—specifically, whether ordered examples provide clearer task conditioning while random ordering acts as a regularizer. By comparing these configurations, we isolate the effect of document ordering on the model's ability to balance relevance and auxiliary objectives.

We observe from Tables 6 and 7, how a randomly shuffled initial ordering of ICL examples compares to the ordering by the first stage. Intuitively, one can assume that demonstrations should closely match the exact setting in which the model is used. Our results demonstrate that stage-one ordering enhances nDCG but adversely impacts the auxiliary objective performance.

However, we generally observe that random shuffling is robust and contributes positively to

Table 5: A comparison to show behavior of different LLMs to targeted ICL examples using similar details as shown in Table 1 and 2. The best score across all the categories is bold, and the second-best scores are underlined.

		TREC DL-2019		TREC DL-2020		Touche-2020			Fair-2022		
Type	Pipeline	nDCG	α-nDCG	nDCG	α-nDCG	nDCG	AWRF	M1	nDCG	AWRF	M1
Baseline	BM25	.4795	.4569	.4936	.4895	.2530	.4811	.1851	.4974	.4901	.2975
	+ Llama-8B	.5977	.5695	.5971	.6357	.2388	.4821	.1748	.5658	.4895	.3228
	+ Llama-70B	.7026	.6441	.6944	**.7242**	.2400	.4994	.1691	.5742	.5060	.3278
	+ GPT-4o-mini	.6971	.6761	.6826	.7039	.2590	.5377	.1936	.5688	.5494	.3428
ICL	+ Llama-8B	.6144	.5968	.5983	.6062	.2136	.5199	.1486	.5316	.5300	.3159
	+ Llama-70B	.6975	.6344	.6906	.6780	**.2625**	.4981	.1687	.6146	.4910	.3428
	+ GPT-4o-mini	.7124	.7135	.6844	.7228	.2608	**.5800**	**.2023**	.5697	**.5697**	**.3526**
Baseline	ColBERT	.7205	.6583	.6864	.6385	.2590	.2994	.1462	.4854	.2068	.1204
	+ Llama-8B	.7363	.6595	.7044	.6622	.2344	.2588	.1169	.5870	.1247	.0917
	+ Llama-70B	**.7766**	.6788	.7471	.6786	.2347	.2606	.1053	.6580	.1850	.1313
	+ GPT-4o-mini	.7699	.6850	.7498	.6843	.2496	.2197	.1027	.6487	.2056	.1466
ICL	+ Llama-8B	.7116	.6527	.6976	.6443	.2089	.2389	.0960	.5183	.2069	.1141
	+ Llama-70B	.7621	.6188	.7563	.6789	.2406	.1995	.0909	.6436	.1822	.1449
	+ GPT-4o-mini	.7601	**.6891**	**.7700**	.7132	.2508	.2602	.1216	**.6606**	.2272	.1628

Table 6: Evaluating the effect of the initial ordering of example documents and ordering with the first stage over TREC DL-2019 and 2020.

	Example	TREC DL-2019		TREC DL-2020	
Pipeline	Ordering	nDCG	α-nDCG	nDCG	α-nDCG
BM25 + Diverse	Random	.7124	**.7135**	**.6844**	**.7228**
	BM25	**.7216**	.6882	.6823	.6999
ColBERT + Diverse	Random	.7601	.6891	**.7700**	**.7132**
	ColBERT	**.7784**	**.6991**	.7670	.7074

Table 7: Measuring document order sensitivity over Touche and TREC Fair-2022, other details are similar to the evaluation shown in Table 6.

	Example	Touche-2020			Fair-2022		
Pipeline	Ordering	nDCG	AWRF	M1	nDCG	AWRF	M1
BM25 + Fair	Random	.2608	**.5800**	.2023	.5697	**.5697**	.3526
	BM25	**.2856**	.5410	**.2180**	**.6029**	.5692	**.4013**
ColBERT + Fair	Random	**.2508**	**.2602**	**.1216**	**.6606**	**.2272**	**.1628**
	ColBERT	.2444	.2028	.1010	.6554	.2260	.1593

the auxiliary objectives. We attribute this to the fact that random ordering mitigates bias and enables the system to generalize effectively across diverse objectives while also enhancing the adaptability of our approach. While document order is a key factor in the robustness of supervised list-wise re-rankers (Pradeep et al., 2023), this appears to have a reduced negative effect on exemplar-based zero-shot list-wise ranking. With likely improvements of supervised rankers in the future, these same improvements may bolster the effectiveness of in-context learning methods.

D Prompt Template

Figure 4 shows the template for including the list-wise examples. The sample output labeled as 'Example ordering' (marked with green) in Figure 4 refers to an ordering - a permutation map of the input - found by maximizing a given objective related to the distribution of the metadata values of each document of the input list $\langle D_1, \ldots, D_k \rangle$ retrieved for the query Q which is similar to Q_c (the current input query). This permutation of a set of input documents retrieved for a similar query is the only mechanism to 'control' the output ranking for the query Q_c.

E Localized Queries used for ICL examples

Using BM25, we retrieve five queries for each test query from the MS MARCO train query set. These similar queries are used as the query for ICL examples. Sample examples of such similar queries for each test are shown in Figure 5.

Prompt header

You are RankGPT, an intelligent assistant that can rank passages based on their relevancy to the query.

Example target ranking for the top-most similar query

I will provide you with $\{m\}$ passages, each indicated by number identifier [ ]. As an example, the first $\{m\}$ passages are ranked based on their relevance to query:
$\{Q\}$
[1] $\{D_1\}$
[2] $\{D_2\}$
...
[k] $\{D_m\}$
The $\{m\}$ passages above are ranked based on their relevance to the search query.
Output: [4] > [1] > ... > [6]

Example ordering

Current (Input) Query

Rank the passages based on their relevance to query:
$\{Q^c\}$
[1] $\{D_1^c\}$
[2] $\{D_2^c\}$
...
[k] $\{D_m^c\}$
Rank the $\{m\}$ passages above based on their relevance to the search query. The passages should be listed in descending order using identifiers. The most relevant passages should be listed first. The output format should be [ ] > [ ], e.g., [1] > [2].

Figure 4: The prompt template used in our work with the header identical to that of (Sun et al., 2023). Different from Sun et al. (2023) our prompt allows provision to include a target ranking for a similar query. In the figure, Q^c denotes the current input query, and D_i^c denotes the document at position i of the input ranked list, which is to be re-ranked.

Localized Queries from TREC DL-2019

Test Query

Do goldfish grow?

Similar Query from MS-MARCO

1. How to grow goldfish faster?
2. How big do shubunkin goldfish grow?
3. How fast do baby goldfish grow?
4. How big can goldfish grow?
5. What is a goldfish?

Localized Queries from TREC DL-2020

Test Query

Why is pete rose banned from hall of fame?

Similar Query from MS-MARCO

1. Why is pete rose banned from mlb?
2. Where does pete rose do autographs?
3. How old is pete rose?
4. When was pete rose born?
5. How many catchers are in the hall of fame?

Localized Queries from Touche-2020

Test Query

Should teachers get tenure?

Similar Query from MS-MARCO

1. How many years does it take to get tenure as a teacher?
2. What is the illinois mandate for tenured teacher evaluation?
3. What are tenure protections?
4. What is tenure mean?
5. Greenspan tenure?

Localized Queries from Fair-2022

Test Query

Architecture

Similar Query from MS-MARCO

1. What do you do in architecture?
2. What is it architecture?
3. What is an architecture do?
4. What is architectural?
5. How roman architecture influenced modern architecture?

Figure 5: An example showing five localized queries that are retrieved for a test query in each test collection.