Modeling Ranking Properties with In-Context Learning

Nilanjan Sinhababu*
Centre for Computational and Data Sciences
IIT Kharagpur, India
nilanjansb@kgpian.iitkgp.ac.in

Andrew Parry*
School of Computing Science
University of Glasgow
United Kingdom
a.parry.1@research.gla.ac.uk

Debasis Ganguly
School of Computing Science
University of Glasgow
United Kingdom
Debasis.Ganguly@glasgow.ac.uk

Pabitra Mitra
Department of Computer Science and Engineering
IIT Kharagpur, India
pabitra@cse.iitkgp.ac.in

Abstract

While standard IR models are primarily 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 for listwise LLM rerankers 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 common 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. Our experiment demonstrates significant improvements in auxiliary objectives, with up to 23% in topical diversity and 20.6% fairness gains across different tasks, while maintaining relevance across benchmarks.

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 their primary objective of providing relevant content to a user, an IR system may have other auxiliary objectives, such as maximizing fairness for protected groups or topical diversity of documents (Carbonell and Goldstein, 1998). Different from relevance, which is a 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 *ranking*, requiring effective modeling of *inter-document* interactions.

Prior work on modeling ranking properties, such as diversity (Carbonell and Goldstein, 1998) 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 supervised approaches is that they are trained on a large quantity 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 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 list-wise 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 engineering" is brittle (Ishibashi et al., 2023; Habba et al., 2025), re-

*These authors contributed equally to this work.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 49256–49271
October 24-29, 2026 ©2026 Association for Computational Linguistics

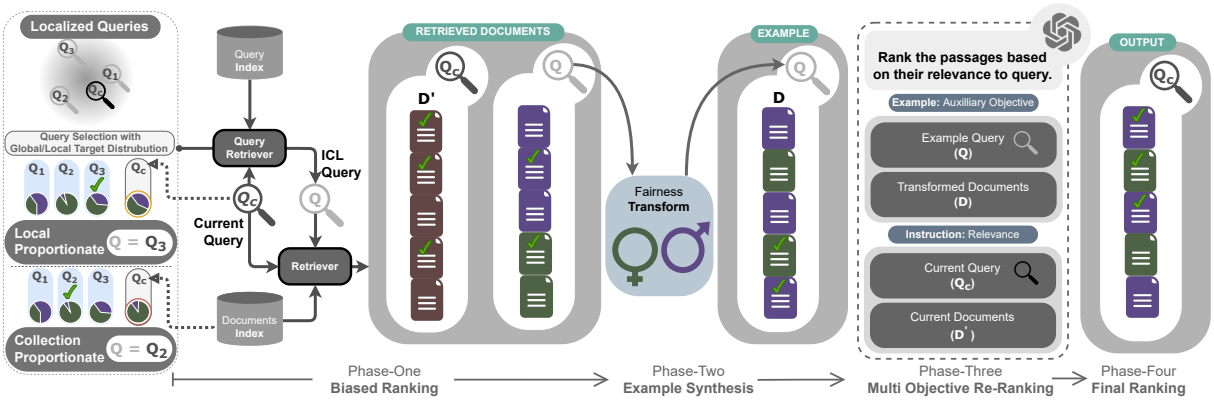


Figure 1: Proposed ICL reranking method uses an example that constitutes a localized query along with its top-retrieved results arranged to satisfy a desired ranking property, such as relevance, fairness, diversity, etc. Phases Two, Three, and the selection of localized queries with local or collection proportionate, constitute our contributions.

quiring iterative refinement as objectives evolve. For instance, consider a legal researcher retrieving prior legal case files to prepare counterarguments; a prompt such as "show diverse perspectives" may yield results that are apparently superficially varied results (e.g., lexical differences) rather than substantively balanced viewpoints (e.g., pro/con arguments on AI surveillance).

In contrast, we propose to use in-context learning (ICL) (Lu et al., 2024), where static instructions are paired with in-context examples of a desired ranking behavior specific to a task. For instance, a single demonstration interleaving positive (pro) and negative (con) arguments could implicitly teach an LLM to diversify viewpoints, even with a task-agnostic instruction like "order by relevance". Our approach circumvents the need for task-specific prompts while accommodating composite or dynamic objectives by combining the primary objective (relevance) with a single auxiliary objective (diversity or fairness). As shown in Figure 1, our method reranks candidate documents using LLMs conditioned on: (1) *localized (on-topic) query examples* from a large query log without requiring relevance assessments and (2) encoding ranking properties via *list-wise demonstrations*, e.g., diversity and fairness. Thus, our contributions are two-fold:

An ICL-based approach that is shown to control desirable ranking properties without any supervised list-wise training. Supervised methods rely on massive amounts of task-specific relevance training data for each new objective (Schlatt et al., 2024; Zehlike and Castillo, 2020). In contrast, our method requires no relevance judgments for the demonstration queries and requires no training. We also demonstrate that our method is flexible in handling metadata group information from the collection. For the fairness task, our collections provide explicit group metadata, which our method directly leverages. However, for the diversity task, we also demonstrate that such metadata can be derived from document clusters, even when the collection lacks explicit grouping metadata.

We establish that demonstrations can consistently change behavior across multiple objectives. Crucially, ablations confirm that demonstration construction is a key factor in effectiveness: Inverting examples significantly degrades performance, affirming its 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 our approach, significantly improving several auxiliary objectives while maintaining ranking effectiveness. Experiments on TREC DL (diversity) (Bajaj et al., 2018; 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 ranking effectiveness for auxiliary objectives (Zehlike and Castillo, 2020), our method maintains effectiveness while significantly improving diversity and fairness. Empirical validation demonstrates that our non-parametric approach consistently outperforms heavily engineered, task-specific supervised methods (such as PAO, Set-Encoder, and DELTR) and heuristic post-hoc interventions (such as MMR and FA*IR) across four different test collections.

We provide our source code to facilitate future

research and the reproducibility of our work[1].

2 Related Work

In this section, we review prior work on fairness and diversity in information retrieval, as well as recent advances in generative reranking and in-context learning that motivate our approach.

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; Jaenich 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; Jaenich et al., 2024).

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, 1998; Clarke et al., 2008). Similar 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 (1998) 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).

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). 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).

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 generalization in pairwise (Sinhababu et al., 2024) and list-wise ranking (Li et al., 2023b). Selecting informative examples that encode task semantics and context improves generalization (Nie et al., 2022; Sinhababu et al., 2024).

3 ICL for Multi-Objective Ranking

In this section, we present a non-parametric framework for controlling list-wise ranking behavior in large language models (LLMs) using in-context learning (ICL). Our approach conditions a generative list-wise ranker on a small number of constructed example rankings, such that desired ranking properties are conveyed implicitly through demonstrated permutations rather than explicit instructions or learned parameters.

Motivation. Prior work has shown that ICL can effectively influence pairwise reranking (Sinhababu et al., 2024). However, many practical retrieval

[1] https://github.com/nilanjansb/ICL_ranker

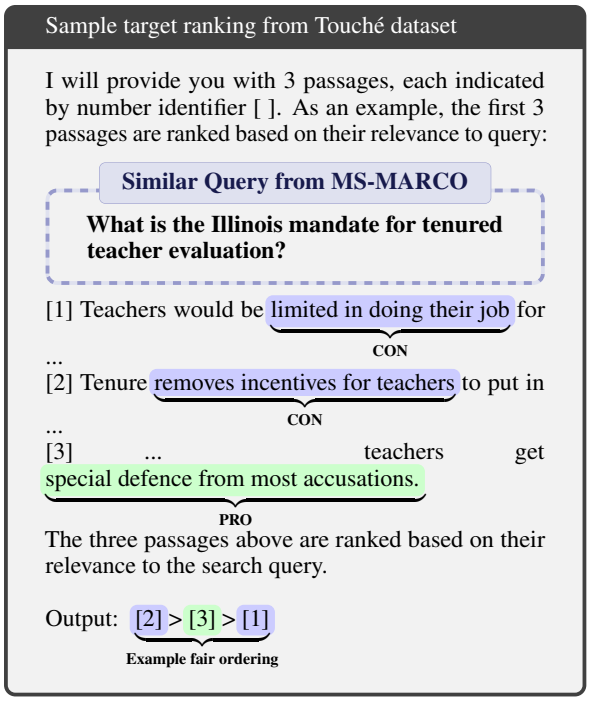


Figure 2: ICL Example for a Touché query. In this example, the objective is to achieve a uniform distribution of the pro and con arguments retrieved from the Touché collection. Here, the reference query - Q is lexically similar to the current input query - Q^c. Documents retrieved for Q are reordered to balance the pro/con ratio as per the target CP or LP. This list is added to the prompt as the example output. The 'PRO' and 'CON' tags are not provided to the model, but are included in this diagram for better understanding.

objectives, such as fairness and diversity, are inherently *list-wise*: they depend on interactions among documents and on the distribution of attributes across a ranked list, rather than on independent document scores.

In list-wise LLM ranking, a query together with a set of retrieved documents is provided as input, and the model generates a permutation of that set (Sun et al., 2023; Liu et al., 2024). This formulation offers a natural opportunity for behavioral control, but expressing complex list-wise objectives directly as natural-language instructions is brittle and often underspecified (Habba et al., 2025). In contrast, ICL allows ranking behavior to be shaped through examples that instantiate the desired trade-offs.

Crucially, this control mechanism is *non-parametric*: it does not require training or fine-tuning a model for each auxiliary objective. Instead, control is achieved by modifying only the example rankings supplied at inference time, enabling flexible adaptation to different ranking criteria without additional supervision.

Localised Example Selection. The effectiveness of ICL in ad-hoc ranking has been shown to depend on selecting demonstrations that are topically aligned with the test query while remaining free of relevance labels (Sinhababu et al., 2024). Given a test query Q_c, 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* - $\mathcal{N}_k(Q_c)$ (see Figure 6 of Appendix D). For our experiments, we use the MS MARCO training set as our query reference set $\mathcal{Q}$ *without* labeled examples. We use BM25 over the MS MARCO training queries and a dense index of BERT vectors to obtain $\mathcal{N}_k(Q_c)$. We refer to the lexical and semantic example selections as **LEX** and **SEM**, respectively. From the retrieved queries, the query with the ranked list attribute distribution closest to the target distribution is selected (one demonstration query), shown as "Query Retriever" in Figure 1. We refer to this selected demonstration query as Q. Details on the exploration of other higher values of k are provided in the Appendix A.4.

Implicit multi-objective control. Our method encodes auxiliary objectives entirely through the *structure* of the example ranking. The prompt instruction remains fixed ("rank by relevance"), while the example permutation exhibits a controlled distribution over document attributes. The example is relevance-ordered within attribute groups, but its composition approximates a target distribution τ.

As a result, the model is never explicitly instructed to optimize for fairness or diversity. Instead, it conditions on a relevance-consistent demonstration that implicitly specifies how relevance should be traded off against other ranking properties when generating a permutation for the test query Q_c.

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. This example ranking is constructed offline and serves solely as contextual input to the LLM at inference time.

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 MS MARCO train set), and prior work has shown that the inclusion of these examples improves ranking preferences (Sinhababu et al., 2024). 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 is specified as a target distribution over categorical document attributes, rather than as an explicit optimization instruction. The example ranking manifests this distribution while preserving within-group relevance order, thereby inducing a list-wise preference structure under a single relevance prompt. 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 $A(D) \in \{\text{'Male'}, \text{'Female'}\}$. Conversely, for topical diversity tasks, $A(D)$ maps to k distinct aspects (where $k > 2$), representing a multi-group distribution across diverse query interpretations.

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 represents the relative proportion of relevant documents for each metadata value ($\mathbb{I}$ denotes the indicator function).

Our approach is flexible with respect to any number of k. In fairness tasks, group selection is typically binary (since the chosen attributes are generally limited in such contexts), whereas in diversity tasks, multiple groups are considered.

Operational source of τ. In realistic deployment settings, relevance judgments are unavailable at test time; hence $\tau(\mathcal{R}(Q))$ of Equation 1 must be instantiated without using test relevance labels. Throughout our experiments, we therefore obtain τ from signals available *prior* to re-ranking: (i) for fairness, we use benchmark-provided document attributes or corpus-level priors; (ii) for diversity, we assume a uniform target over automatically induced aspect clusters (see Appendix A.2). This avoids train–test leakage while keeping the method non-parametric. Since these attributes are part of the collection, we refer to them as Collection Proportionate (**CP**). It ensures that the example ranking matches the distribution of the collection, preventing a biased reranking. In addition to global priors, we also define a target distribution that accommodates query-specific constraints. Specifically, we define Local Proportionate (**LP**), which dynamically derives the target distribution from the attribute distribution within the top-retrieved candidate list of the test query.

Grouping by metadata values. We aim to rerank $\theta(Q)_m$ so that prefixes approximate τ. Treating retrieval scores as *monotonic surrogates* for relevance[2], we partition $\theta(Q)_m$ into k sequences $\{\theta(Q)_m^{(i)}\}_{i=1}^k$ by attribute value $A(D) = i$, preserving the within-group retrieval order. We do not expose labels to the ranker at runtime, either in demonstrations or test instances.

Auxiliary Objective based Rank Induction. To instantiate the target distribution while minimally perturbing relevance, we construct the example ranking using a greedy prefix-based selection procedure. Let p denote the current prefix length and let $\mathcal{C}_{p+1}$ contain the next unselected candidate (head) from each group $\theta(Q)_m^{(i)}$. We greedily select the $(p{+}1)$-th document to minimize the divergence between the prefix histogram and the target:

$$D_{p+1} = \arg\min_{D \in \mathcal{C}_{p+1}} \mathrm{KL}\big(\tau(\mathcal{R}(Q)), \tau(\langle D_1, \ldots, D_p \rangle \cup D)\big). \quad (2)$$

After selection from group j, we advance that group's pointer and repeat until m items are chosen. The procedure defined in Equation 2 mirrors classical exposure-control heuristics, but is used here solely to synthesize a demonstration rather than to directly rerank test documents. Demonstrations are

[2]Our approach inherits biases of the first-stage ranker, since example construction assumes monotonic relevance ordering within groups.

then appended before the test instance to influence ranking behaviour.

4 Evaluation

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

RQ-1: Can example rankings improve effectiveness in a zero-shot list-wise ranking task?

RQ-2: How do ICL examples compare to directly instructing an LLM in terms of auxiliary objective effectiveness?

RQ-3: What are the effects of ICL example ranking strategies that are adversarial to a target distribution-based auxiliary objective?

4.1 Experimental Setup

Objectives and Datasets. Our methodology is fundamentally task-agnostic and designed to generalize to any list-wise reranking objective. However, our current investigation focuses on two auxiliary objectives: diversity and group fairness. To improve 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 MSMARCO passage corpus (Bajaj et al., 2018) and TREC Deep Learning (**DL-2019** and **DL-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 **Fair-2022** (Ekstrand et al., 2022) and **Touché** (Bondarenko et al., 2020), both of which include explicit group attributes.

Our method allows a query log from any collection to serve as an anchor set. To demonstrate this flexibility, we use a query reference set from the MSMARCO training set (approximately 8×10^5 entries) to construct demonstrations for the in-domain test sets DL-2019 and DL-2020. We then use the same query reference set, without any modification, to construct demonstrations for entirely distinct, out-of-domain test collections such as Fair-2022 and Touché.

Measures. To evaluate ranking quality, we report Normalized Discounted Cumulative Gain (**nDCG**) for relevance as is standard on chosen benchmarks (Craswell et al., 2020b,a; Bondarenko et al., 2020; Ekstrand et al., 2022). For measuring fairness, we apply Attention-Weighted Rank Fairness (**AWRF**) and nDCG-AWRF combination (**M1**), as is standard in group fairness (Ekstrand et al., 2022). We report α**nDCG** following Clarke et al. (2008) with $\alpha = 1$ to measure diversity (see Section 4.3). All metrics are reported at a cutoff of 10.

Multiple Comparison Correction. To test the statistical significance of our results, we apply the Holm-Bonferroni correction to account for multiple hypothesis testing. Specifically, we define separate comparison families based on the reference baseline. We compare against the first-stage retrieval baseline (BM25/ColBERT) and against the 0-shot baseline. The correction is applied jointly within each family across all evaluated tasks, datasets, retrievers, and metrics.

Configurations. Our ICL methods are labeled to denote the explicit components of our demonstration engineering setup, comprising of **Diverse** or **Fair** objectives, **LEX** or **SEM** similarity, and **CP** or **LP**-based target distribution. For example, a pipeline defined as Diverse-LEX-CP indicates an auxiliary objective of topical diversity, using a lexical similarity-based retrieval mechanism to select a localized query from the query log, and mapping the target distribution (τ) using global collection metadata priors.

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 retrieval models. The second-stage ranker is our proposed ICL-based method. We apply a sliding window-based re-ranking over the top 100 documents. The window size is fixed at 20 documents with a stride size of 10 (Sun et al., 2023; Pradeep et al., 2023).

We perform all experiments using `GPT-4o-mini` in the main experiments. We also test the effectiveness of our approach on different models, comparing the closed-source `GPT-4o-mini` with the open-weighted `Llama-3.1 (8B & 70B)` described in detail in Appendix B. Details on model configurations, hyper-parameters, and implementation are provided in Appendix A.

Ablations. Our ablation aims to establish that our augmentation of ICL target distributions indeed leads to improved effectiveness; thus, we ablate how we construct list-wise demonstrations. *Adversarial Examples* (**Adv**): By swapping the proportions, we seek to maximize the KL divergence from the target distribution (instead of minimizing this measure as per Equation 2), e.g., flipping a 3:2 gender ratio to 2:3. *Relevant Examples* (**Rel**): A transformation from a random order to an order by relevance determined by a first-stage system. *Static Examples* (**Static**): 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.

4.2 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*". The base model applied for PAO and our approach is identical (GPT is shown in our primary findings).

Baselines for Diversity. For the diversity-ranking task, we compare our approach to the following baselines.

Max Margin Relevance (MMR) (Carbonell and Goldstein, 1998): 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 (Schlatt et al., 2024): A cross-encoder trained with diversity-aware loss on MS-MARCO. We use `set-encoder-large`[3].

Baselines for Fairness. For the fair-ranking task, we compare our approach the following.

FA*IR[4] (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[5] (Zehlike and Castillo, 2020): A learning-to-rank model 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).

Table 1: Evaluating nDCG and αnDCG performance over TREC DL-2019 and 2020 for the Diversity objective. The maximum score in each category is shown in bold. Symbols $\star$ and † indicate statistical significance after Holm–Bonferroni correction (adjusted $p < 0.05$) for comparisons against the first-stage and 0-shot baselines, respectively.

			DL-2019		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	+ Set-Encoder	.5977	.5695	.5971	.6357
	5	+ 0-shot	.6971	.6761	.6826	.7039
ICL	*6*	+ Diverse-LEX-CP	**.7124***	.7135*	**.6844***	.7228*
	7	+ Diverse-SEM-CP	.7109*	**.7309***†	.6831*	**.7355***
	8	+ Diverse-LEX-LP	.6977*	.7098*	.6814*	.7141*
	9	+ Diverse-SEM-LP	.7089*	.7254*†	.6824*	.7223*
Baselines	*10*	ColBERT	.7205	.6583	.6864	.6385
	11	+ PAO	.6949	.6494	.6996	.6910
	12	+ MMR	.7173	.6567	.6873	.6405
	13	+ Set-Encoder	.7320	.6172	.7245	.6338
	14	+ 0-shot	**.7699**	.6850	.7498	.6843
ICL	*15*	+ Diverse-LEX-CP	.7601	.6891	**.7700***†	.7132*
	16	+ Diverse-SEM-CP	.7476	**.7012***	.7543*	**.7322***†
	17	+ Diverse-LEX-LP	.7582	.6827	.7638*	.7090*
	18	+ Diverse-SEM-LP	.7419	.6833	.7598*	.7104*

4.3 Findings

In this section, we present and discuss our 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 approaches significantly outperform the first stage and 0-shot baselines in topical diversity (αnDCG), with improvements of up to 23% (e.g., compare rows 1,5 with 6-9 and 10,14 with 15-18). We observe that semantic query retrieval (SEM) methods are more effective

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

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

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

Table 2: Evaluating AWRF, nDCG, and M1 on Touché-2020 (PRO/CON) and TREC Fair-2022 (M/F) for the fairness objective; other details are similar to Table 1.

Type		Pipeline	Touché-2020 nDCG	Touché-2020 AWRF	Touché-2020 M1	Fair-2022 nDCG	Fair-2022 AWRF	Fair-2022 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	.2590	.5377	.1936	.5688	.5494	.3428
ICL	*6*	+ Fair-LEX-CP	**.2608**	**.5800***†	**.2023**	**.5697***	.5697*	.3526*
	7	+ Fair-SEM-CP	.2531	.5594*	.1960	.5637*	.5549*	.3485*
	8	+ Fair-LEX-LP	.2516	.5437*	.1949	.5600*	.5484*	.3416*
	9	+ Fair-SEM-LP	.2308	.5420*	.1912	.5596*	.5488*	.3418*
Baselines	*10*	ColBERT	**.2590**	.2994	.1462	.4854	.2068	.1204
	11	+ PAO	.2234	.2388	.0956	.6565	.1837	.1411
	12	+ FA*IR	.2500	**.3598**	**.1698**	.2111	**.5896**	**.3320**
	13	+ DELTR	.2518	.2216	.1078	.2128	.3370	.2215
	14	+ 0-shot	.2496	.2197	.1027	.6487	.2056	.1466
ICL	*15*	+ Fair-LEX-CP	.2508	.2602†	.1216	**.6606***	.2272	.1628*
	16	+ Fair-SEM-CP	.2503	.2316	.1137	.6527*	.2148	.1522*
	17	+ Fair-LEX-LP	.2576	.2318	.1142	.6502*	.2011	.1467
	18	+ Fair-SEM-LP	.2545	.2303	.1139	.6513*	.2117	.1516*

in-domain, aligning with earlier findings (Sinhababu et al., 2024). Notably, these surpass both post-hoc diversification (MMR) and a supervised list-wise method (Set-Encoder) by 8-15% in αnDCG (compare rows 3-4 with 6-9 and 12-13 with 15-18), 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.

Table 2 shows results for the auxiliary objective of group fairness. Our approach exceeds all baselines on Touché and is competitive with post-processing and supervised methods on Fair-2022. Our method achieves approximately a 52% improvement in nDCG performance compared to DELTR and FA*IR applied post-hoc to an LLM relevance ranking, as shown by comparing Rows 3-4,12-13 with 6-9,15-18 in Table 2. However, this gain in effectiveness is accompanied by a reduction in fairness on the Fair-2022 dataset. For this task, however, we observe that lexical query retrieval is more effective out-of-domain, consistent with existing findings (Sinhababu et al., 2024). 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 Touché. 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.

Table 3: Ablations showing effects on diversity and fairness tasks. The "+Target" row corresponds to the target-distribution specific ICL re-ranking results using LEX-CP. Suffixes *a* to *e* represent the statistical significance of "+Target" when compared to the first-stage, 0-shot, Adv, Rel, and Static methods, respectively, computed via Holm–Bonferroni correction (adjusted $p < 0.05$).

Method	DL-2019 nDCG	DL-2019 αnDCG	DL-2020 nDCG	DL-2020 αnDCG	Touché-2020 nDCG	Touché-2020 AWRF	Fair-2022 nDCG	Fair-2022 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

Overall, our methods using collection proportionate (CP) are consistently better than those using local proportionate (LP). This, however, is not a drawback; instead, it indicates reliance on the distribution of collection attributes in evaluation measures (such as AWRF). Additionally, this highlights the flexibility of our approach, which adapts to both global and local distributions to select localized queries for ICL, advancing over existing ICL approaches for context selection (Sinhababu et al., 2024; Xie et al., 2024; Li et al., 2023a).

Nevertheless, our approach with ICL outperforms the 0-shot setting. With BM25 as the first-stage ranker, our approach is more effective than all but one baseline, FA*IR with grid-search optimised parameters, on Fair-2022. Under ColBERT, 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-context learning improves ranking performance over direct instructions. We now examine the impact of prompt-based optimization on fairness and diversity objectives(**RQ-2**). As shown

in Tables 1 and 2, our ICL approach is more effective than the Prompt-based Auxiliary Objective (PAO) baseline. Relative to the 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. Marginal improvements are found in αnDCG with PAO on DL-2019 under BM25 and on DL-2020 under ColBERT (Table 1, Rows 2 and 11). In terms of AWRF, a positive gain is observed only for Touché when ColBERT is used as the first-stage retriever (Table 2, Row 11). 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 optimizing directly for relevance yield significant improvements in nDCG relative to both BM25 and ColBERT. This is a natural consequence of the original motivation for ICL demonstrations (Lu et al., 2024). From Rows 4 and 10, see 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 (**RQ-3**).

Additionally, see 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 lead to a substantial decline in effectiveness, occasionally reducing it below that of the base ranker. This validates that the locality of queries in ICL examples remains useful in a list-wise setting, as was found by Sinhababu et al. (2024) in a pair-wise setting.

Sensitivity Analysis of αnDCG. To assess the sensitivity of our results to the choice of α in αnDCG, we evaluate three settings, $\alpha \in \{0.5, 0.75, 1\}$. While $\alpha = 0.5$ is a commonly used convention, our primary setting of $\alpha = 1$ sets high emphasis on diversity in the relevance–diversity trade-off. This choice is motivated by our objective of evaluating the model's capacity to diversify, while we evaluate relevance separately using standard nDCG.

Table 4: Sensitivity to the α parameter on DL-2019 and DL-2020. We compare first-stage retrieval (BM25 and ColBERT), 0-shot, and ICL (Diverse-LEX-CP) across $\alpha \in \{0.5, 0.75, 1\}$. We use bold font to denote $\alpha = 1$, our primary setting.

		DL-2019			DL-2020		
Model	α	**1**	0.75	0.5	**1**	0.75	0.5
BM25		.457	.469	.480	.490	.491	.492
0-shot		.676	.689	.700	.704	.721	.746
ICL		.714	.726	.739	.723	.743	.757
ColBERT		.658	.681	.705	.639	.665	.671
0-shot		.685	.692	.697	.684	.695	.706
ICL		.689	.694	.698	.713	.718	.722

To ensure that our findings are not an artifact of the $\alpha = 1$ setting, we compare first-stage retrieval, 0-shot, and ICL across different α values. As shown in Table 4, ICL consistently outperforms the corresponding 0-shot setting for both datasets across different values of α. These consistent gains demonstrate that the effectiveness of our proposed ICL method is robust to the choice of α. We use $\alpha = 1$ as our primary setting because it places greater emphasis on the diversification capability that is central to our task.

Concluding Remarks. We propose an approach to multi-objective ranking that leverages demonstrations to balance relevance and auxiliary objectives. Our experiments confirm that examples modeled for fairness and diversity improve target objectives significantly without compromising relevance. We found that effectiveness gains can be controlled with adversarial examples, leading to biased downstream rankings. Additionally, our approach improves over directly instructing a model for each objective. Our approach demonstrates greater effectiveness compared to task-specific post-hoc and supervised methods, 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 behavior beyond the objectives investigated in this work. Our ICL-based approach for ranking topical diversity achieves gains of up to 23% in the αnDCG metric and a 20.6% improvement in the AWRF metric for fairness, while often preserving or improving nDCG.

5 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.

We demonstrate that a single example is generally sufficient for the collections in our experiments. We expect modern LLMs with large context windows to accommodate applications involving longer documents, such as legal or medical retrieval. However, we acknowledge that complex real-world applications may benefit from additional examples that may pose greater computational and representational challenges.

We represent gender as binary in accordance with the collection metadata; however, this simplification does not fully capture the complexities of gender in the real world.

Our approach requires an existing query log, which, in low-information environments or low-resource languages, may pose difficulties for adoption. 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.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. Ms marco: A human generated machine reading comprehension dataset. *Preprint*, arXiv:1611.09268.

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 *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 335–336, New York, NY, USA. Association for Computing Machinery.

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.

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 Jaenich, 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 '24, page 1504–1513, New York, NY, USA. Association for Computing Machinery.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick 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)*, pages 6769–6781, Online. 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 '20, page 39–48, New York, NY, USA. Association for Computing Machinery.

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.

Qi Liu, Bo Wang, Nan Wang, and Jiaxin Mao. 2024. Leveraging passage embeddings for efficient listwise reranking with large language models. *CoRR*, abs/2406.14848.

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)*, pages 5098–5139, Bangkok, Thailand. 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.

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.

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*, pages 11030–11047, Bangkok, Thailand. 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.

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.

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.

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.

Table 5: Statistics of the datasets used in our experiments.

Task	Collection	$\|\mathcal{C}\|$	Queries	$\|\mathcal{Q}\|$	Name	Values
Fairness	TREC Fair	6.5M	Fair-2022	50	Gender	M, F
	TouchéV2	383K	Touché'20	49	Stance	PRO, CON
Diversity	MS MARCO	8.8M	DL'19	43	Topic	$\mathbb{Z}$
			DL'20	54		

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.

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 Touché 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 5.

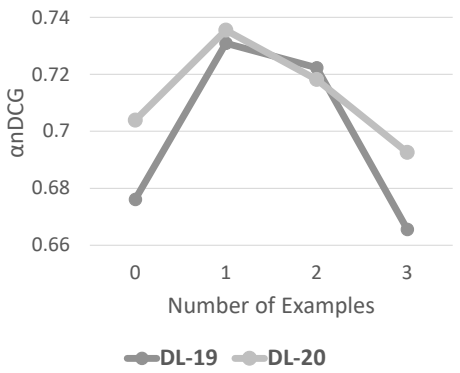
A.2 Clustering for Diversity

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.

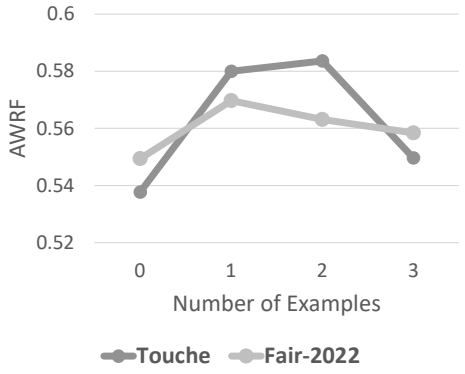
A.3 Query Similarity Retrieval

Following (Sinhababu et al., 2024), we retrieve similar queries from the MSMARCO training split using BM25 over query text. We retrieve the top-3 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 Number of Examples



(a) Analyzing the number of ICL examples for the Diversity task.



(b) Analyzing the number of ICL examples for the Fairness task.

Figure 3: Experimenting with the GPT-4o-mini LLM, we observe significant improvement in αnDCG and AWRF using 1-shot in all the experiments. However, a sudden degradation in the scores after 1-shot indicates possible ambiguity in the example or LLM context length limits.

We test by varying the number of examples (k) from 1 to 3 and found that $k = 1$ yields the best performance improvement as shown in Figure 3. With each increase in the value of k, the context requirement doubles, since each example contains a query and its set of 20 example documents (considering a window size of 20). We observe that $k > 1$ exceeds the context of small LLMs, resulting in degraded performance. For larger LLMs, we observe no significant improvement over $k = 1$ performance but with increased inference time, similar to what was reported in (Sinhababu et al., 2024).

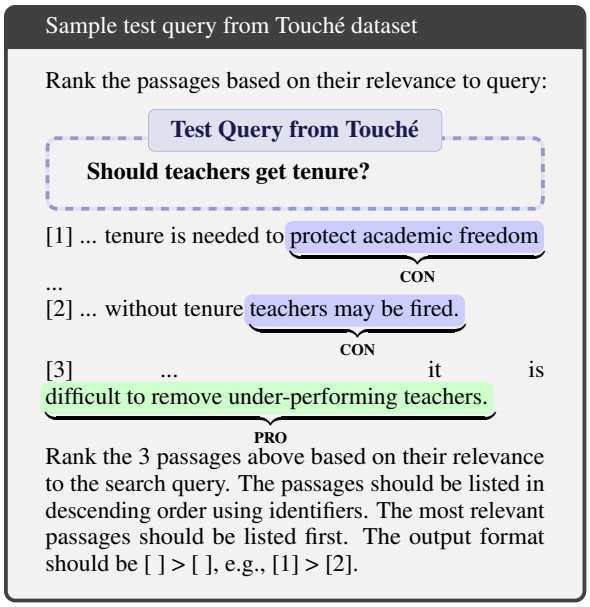


Figure 4: The figure shows a sample input query from the Touché dataset. The ICL example of a related query from MS MARCO and its example output (balancing both relevance and pro:con parity, as shown in Figure 2) is used to *control* the current query's reranking.

A.5 LLM Configurations

- **Llama-3.1-8B-Instruct**: 8B decoder-only LLM[6] 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 `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[7]. 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[8] 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).

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 6, 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.

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, 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 diversity cues of the examples and the relevance constraints of the instructions, likely due to their greater representational capacity.

In fairness experiments, we observe a similar outcome, except for Touché under ColBERT, as seen in Table 2, suggesting that even smaller LLMs are effective at modeling an auxiliary objective. The larger model, however, significantly outperforms the smaller model across the majority of settings.

C Example ordering

We initially adopted random ordering for in-context examples due to its simplicity and computational efficiency. However, prior work highlights that

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

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

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

Table 6: A comparison to show the 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		Touché-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 7: 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 8: Measuring document order sensitivity over Touché and TREC Fair-2022, other details are similar to the evaluation shown in Table 7.

	Example	Touché-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 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 7 and 8, 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 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 5 shows the template for including the list-wise examples. The sample output labeled as 'Example ordering' (marked with green) in Figure 5 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

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 5: 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.

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 6.

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