Rankers, Judges, and Assistants: Towards Understanding the Interplay of LLMs in Information Retrieval Evaluation

Krisztian Balog
Google DeepMind
Stavanger, Norway
krisztianb@google.com

Donald Metzler
Google DeepMind
Mountain View, USA
metzler@google.com

Zhen Qin
Google DeepMind
Mountain View, USA
zhenqin@google.com

Abstract

Large language models (LLMs) are increasingly integral to information retrieval (IR), powering ranking, evaluation, and AI-assisted content creation. This widespread adoption necessitates a critical examination of potential biases arising from the interplay between these LLM-based components. This paper synthesizes existing research and presents novel experiment designs that explore how LLM-based rankers and assistants influence LLM-based judges. We provide the first empirical evidence of LLM judges exhibiting significant bias towards LLM-based rankers. Furthermore, we observe limitations in LLM judges' ability to discern subtle system performance differences. Contrary to some previous findings, our preliminary study does not find evidence of bias against AI-generated content. These results highlight the need for a more holistic view of the LLM-driven information ecosystem. To this end, we offer initial guidelines and a research agenda to ensure the reliable use of LLMs in IR evaluation.

CCS Concepts

• **Information systems → Information retrieval**.

Keywords

Large language models, ranking, evaluation

1 Introduction

Due to their remarkable capabilities, large language models (LLMs) are fundamentally reshaping the field of information retrieval (IR), becoming integral to core ranking algorithms and the automation of evaluation processes. Beyond their role in core IR processes, LLMs are also powering AI assistants that are rapidly changing how users generate content, from writing emails and articles to creating code and translating content between languages. As the reliance on LLMs is expected to deepen given their potential, it is increasingly crucial to maintain a balanced perspective by assessing and acknowledging

the potential risks alongside the undeniable benefits. Could this heavy reliance on LLMs across content creation, retrieval, ranking, evaluation, etc., inadvertently introduce or amplify biases within these systems?

Recent research has begun to explore some of these emerging issues. For example, studies have shown that LLMs can exhibit biases in their output, favoring LLM-generated content over human-generated ones [10], and perpetuating biases present in their training data [15, 21, 33]. Furthermore, LLM-based rating systems have been found to be susceptible to manipulation [2], may not accurately reflect human preferences [28], and demonstrate self-inconsistency [50]. Additionally, the phenomenon of "model collapse" has also been observed, where LLMs trained on synthetic data generated by other LLMs can lead to a degradation of quality and diversity in generated content [47].

Within the IR research community, the use of LLMs for assessment is a subject of ongoing debate [14], with opinions ranging from complete rejection of LLMs for relevance assessment [48] to the assertion that they can fully replace human judgments [55]. Investigations have thus far focused on the agreement of LLM-generated ratings with human assessments [14, 54, 55] and the potential for LLMs to introduce biases in search results [10]. However, a comprehensive analysis of the implications of LLMs across the entire information ecosystem, from content creation with AI assistance to LLM-based reranking and LLM-based judges for evaluation, remains a critical gap in the current literature.

This paper aims to advance our understanding of these issues by synthesizing prior research and, crucially, providing novel empirical evidence. We specifically focus on the novel challenge of understanding the effect LLM-based rankers and AI-powered content creation have on an LLM-based judge's ability to accurately assess relevance. Prior work has separately noted the *potential* interaction between LLM-based rankers and judges [14, 30, 42, 54] (an interaction that has yet to be empirically investigated), while other initial work has explored the relationship between AI-powered content creation and rankers [10]. However, we argue that the complex interplay between each of these roles must be considered holistically to fully understand the potential implications of widespread adoption of LLM-based judges. We present initial results demonstrating the importance of this interconnected perspective, showcasing how the use of LLMs across the information lifecycle can influence the accuracy and potential biases of LLM judges.

We start by considering the case of LLMs being used as both rankers and judges and present the first empirical demonstration of a significant bias of LLM judges towards LLM-based rankers. Novel to our approach is the examination of LLM judge performance via the use of oracle rankers, allowing for a controlled assessment of
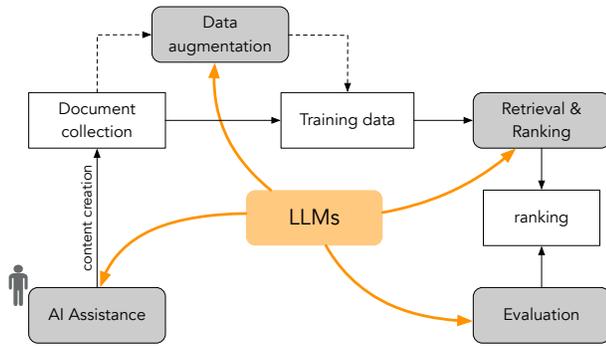
Figure 1: LLM usage in modern information access systems.

LLM judge behavior and discriminative ability. Using the TREC 2019 and 2020 Deep Learning track datasets, we conduct experiments that also compare different-sized LLM judges within the same model family. Our results reveal several key findings: (1) LLM judges are more lenient in their relevance assessments than human judges, confirming previous observations [55]; (2) LLM judges exhibit a significant bias towards LLM-based rankers, a phenomenon previously only hypothesized; and (3) LLM judges demonstrate limited ability to discern subtle, yet statistically significant, performance differences between systems. Additionally, we conduct a preliminary study into whether LLM judges demonstrate biases when they encounter AI-generated content. Contrary to some previously published findings [25, 27, 37], our experiments do *not* provide evidence of this bias, suggesting that deeper, more rigorous empirical evaluations are required to better understand this phenomenon.

What emerges from these targeted studies is a better picture of how different interactions between LLM-based components give rise to different behaviors within LLM-based judges. Taken together, our findings yield one of the most holistic views of this problem space, provide unique insights into best practices for leveraging LLMs as judges, and motivate a rich set of future research questions that will need to be answered to understand the complexities of these interactions even better.

In summary, the primary contributions of this paper are: (1) a review of how LLMs are currently used in IR, bringing attention to the interconnected roles they play, and synthesizing the current understanding of their interactions; (2) experiments that highlight how interactions between LLMs might result in inaccurate or biased assessments of retrieval effectiveness; (3) a preliminary set of guidelines for using LLMs in IR evaluation; and (4) a research agenda aimed at sparking further discussion and research along this emerging direction.

2 Background

This section overviews the main uses of LLMs in information access, illustrated in Fig. 1, providing context for subsequent analysis of the interplay between some of these uses.

LLMs as Rankers. In modern large-scale IR systems, a multistage retrieve-then-rerank pipeline has become a prominent approach, wherein an initial retrieval stage, often based on lexical

matching or embedding-based methods, is followed by one or multiple reranking stages, utilizing more sophisticated models to refine the results. This reranking stage frequently employs LLMs, either fine-tuned for the task of ranking [34, 35, 39, 64] or via prompting in a pointwise [12, 20, 44], pairwise [40], or listwise [29, 51] fashion. Dai et al. [10] present results suggesting an inherent bias in neural retrieval models toward LLM-generated texts. This *source bias* may stem from shared Transformer-based architectures and pretraining approaches, and can lead to "semantic shortcuts" during matching. Neural IR models are also shown to be vulnerable to adversarial attacks, such as keyword stuffing and content injection [38, 52].

LLMs as Judges. Early LLMs, such as BERT, have been utilized for measuring the distributional similarity of texts [60, 61] and for evaluating specific tasks via fine-tuning, including machine translation [63], text summarization [26], and question answering [31]. The arrival of generative LLMs, such as ChatGPT, have enabled various data labeling and annotation tasks [17]. The use of LLMs as surrogates for humans for evaluation, often referred to as "LLM-as-a-Judge" [62], now extends across virtually all natural language processing tasks, including text summarization and dialog response generation [18]. However, recent research increasingly demonstrates their limitations, such as favoring longer responses (*length bias*) [13, 58] or content generated by similar models (*self bias*) [27, 59]. Our interest is specifically in the use of LLMs for relevance assessments in IR. MacAvaney and Soldaini [30] is among the first to employ LLMs for automatic relevance labeling. They specifically focus on a setting where a single known relevant document per query is available for evaluation and explore several one-shot approaches. Faggioli et al. [14] present a spectrum of human-machine collaboration for producing relevance assessments, from AI assistance to fully automated judgments. They conduct a preliminary assessment of LLMs' capabilities of relevance judgments on two TREC collections and report a fair agreement between human assessors and LLMs. Thomas et al. [54] experiment with various prompt templates to improve quality and observe better agreement with the official TREC labels than Faggioli et al. [14]. These improvements are attributed to both prompt design and the use of a more capable LLM. Thomas et al. [54] further share experiences on using LLMs for relevance assessment at Microsoft Bing, where LLMs have reportedly been used, in conjunction with expert human labelers, since late 2022. Upadhyay et al. [56] reproduce results from Thomas et al. [54], verifying their claims, and create an open-source implementation (UMBRELA). Most recently, LLMs are leveraged in the TREC 2024 Retrieval Augmented Generation (RAG) track for automatic relevance assessment [55]. Relative system rankings are found to correlate with those obtained using human judgments, even if human assessors apply stricter relevance criteria than LLMs [55]. The authors also experiment with various LLM-assisted labeling processes, such as using UMBRELA to pre-filter the pools or to suggest relevance labels that human judges can then post-edit, but find that those solutions "do not appear to have obvious tangible benefits over fully automatic processes" [55]. Clarke and Dietz [6] raise concerns about the claims made by Upadhyay et al. [55] and highlight how LLM-based judgments fail to demonstrate strong alignment with manual judgments for top-performing systems. They further present evidence that when evaluation is performed

through a publicly known automatic process, such as UMBRELA, it can be subject to manipulation. Chen et al. [5] show that when performing relevance assessments in batches, the relevance levels of earlier documents in a batch influences the relevance judgments of subsequent documents, and that some LLMs are more affected by this so-called *threshold priming effect* than others. Alaofi et al. [2] compare various open-source and proprietary LLMs in labeling passages for relevance. They demonstrate that most LLMs exhibit some degree of susceptibility to judging non-relevant documents as relevant if query words are inserted at random positions, simulating a keyword stuffing SEO strategy. Rahmani et al. [43] present a large-scale synthetic passage ranking collection, SnyDL, by extending the TREC 2019-2023 Deep Learning collections via LLM-generated labels, and observe a high agreement on system ordering.

LLMs as Assistants. There is a wide array of AI tools available to aid people with content creation. Focusing only on textual content, the spectrum ranges from basic grammar and spell checkers to advanced tools that generate full articles. Studies indicate that by late 2024, LLM assistance is detectable in a significant portion of various text domains, with estimates reaching up to 18% of financial consumer complaints and 24% in corporate press releases [22]. The use of powerful LLMs can lead to situations where it is unclear whether the content is primarily human-created with AI assistance or the other way around.

LLMs for Data Augmentation. While not considered for this role in the current paper, LLMs are also used for data augmentation. For example, Dai et al. [11] use few-shot prompting to generate synthetic queries, while Bonifacio et al. [3] consider query generation in a full unsupervised setting. Soudani et al. [49] present a survey on synthetic dialogue data generation in open-domain, task-oriented, and information seeking dialogue systems. The use of LLM-generated data brings forth new challenges in bias and unfairness, potentially affecting the reliability of IR systems [9].

3 Critical Issues with LLMs as Judges

While LLMs offer promising capabilities for automated evaluation in IR, a growing body of research highlights potential limitations and raises critical concerns about their widespread adoption as judges. This section synthesizes findings from prior work, identifying key challenges that warrant further investigation. We categorize these challenges into two broad areas: the quality of LLM judgments (Section 3.1) and the vulnerability of LLM judges to bias and manipulation (Section 3.2). Within these areas, we discuss specific issues related to validity, discriminative power, reliability, reproducibility, susceptibility to manipulation, and systemic biases. These issues, if unaddressed, could undermine the integrity of IR evaluation and potentially lead to misleading conclusions about system performance. This section discusses these critical issues, while Section 4 presents initial experiments designed to provide empirically-driven insight into each of the issues and Section 5 touches upon the fundamental issue of whether, and how, LLM judges should be used in practice.

3.1 Quality of Judgments

The fundamental question underlying the use of LLMs as judges is whether their judgments accurately reflect "true" relevance and

effectively differentiate between systems of varying quality. We break down this question of quality along two sub-dimensions: *validity and discriminative power* and *reliability and reproducibility*.

3.1.1 Validity and Discriminative Power. For LLMs to serve as effective judges, their assessments must align with human judgments of relevance. Existing research measures this in two ways: (1) agreement on individual document-query relevance labels and (2) agreement on the relative ranking of a set of systems.

- *Agreement on Individual Relevance Judgments:* Several studies demonstrated that it is possible to use LLMs for relevance assessment and obtain performance comparable to TREC judges [14, 55] and notably better than crowd judges [54]. At the same time, it has also been observed that LLMs are more lenient when labeling a document relevant [2, 55], which leads to inflated evaluation scores. This leniency can lead to inflated evaluation scores, potentially masking subtle differences between systems.
- *Agreement on System Rankings:* A common approach to meta-evaluating LLM judges is to compare the relative ranking of retrieval systems based on LLM assessments with the ranking based on human-generated relevance judgments. This typically involves calculating the correlation between the two rankings, often using systems submitted to TREC tracks [14, 30, 55]. While high correlation is often interpreted as evidence of LLM judge validity, this approach has significant limitations.

Issue #1: Discriminative Ability and the Limits of Correlation
Even though several studies report on strong leaderboard correlation between human and LLM judgments, Clarke and Dietz [6] argue that Kendall's τ is "less informative for assessing progress at the top of the leaderboard" and demonstrate that LLM-based assessments fail to reliably identify the best-performing systems. Further, Alaofi et al. [2] show that correlation-based meta-evaluation hides interesting failure patterns. A crucial, often overlooked, aspect is the disconnect between typical TREC evaluation setups and the needs of many practical IR scenarios. TREC evaluations often involve dozens of systems with widely varying approaches and performance levels. In contrast, practitioners often need to compare a small number of high-performing (state-of-the-art) systems or distinguish between subtle variations of a single system (e.g., in ablation studies). It remains an open question whether LLM judges possess the necessary sensitivity to reliably detect small but meaningful performance differences in such scenarios. Indeed, achieving high correlation is inherently easier with a larger and more diverse set of systems; simply including more systems with varying performance levels can artificially inflate correlation, even if the LLM judge struggles to differentiate between the top contenders. It thus remains an open question: *Can LLM judges reliably distinguish between high-performing systems with small, but meaningful, performance differences?*

3.1.2 Reliability and Reproducibility. Beyond validity, a critical concern for LLM-based evaluation is the *reliability* and *reproducibility* of the judgments. Even if an LLM demonstrates a reasonable level of agreement with human judgments on average, its utility as a judge is undermined if its assessments are highly sensitive to seemingly minor variations in setup or input. Indeed, existing

research demonstrates that LLM judgments can be significantly influenced by factors such as the choice of LLM [2, 5, 14], the specific wording and structure of the prompt [2, 54], and even the order in which documents are judged [5]. This variability raises concerns about the reliability of results obtained with a single LLM.

Issue #2: The Impact of Model Choice A recurring theme in the literature is that more powerful LLMs (typically larger models with more parameters and trained on larger datasets) tend to exhibit better performance and consistency as judges [2]. This raises a crucial, but largely unexplored, question: *To what extent would the conclusions of a study change if a more (or less) powerful LLM were used as the judge?* This sensitivity to model choice has not been systematically investigated, particularly in the context of comparing high-performing systems where subtle differences matter.

3.2 Vulnerability to Bias and Manipulation

Beyond the inherent quality of judgments, a separate set of concerns revolves around the potential for LLMs to be biased or manipulated, thereby impacting evaluation outcomes.

3.2.1 Vulnerability to Manipulation. A significant concern with the adoption of LLMs as judges is their potential vulnerability to adversarial manipulation. Initial research suggests that LLM judges might be vulnerable to keyword stuffing and other SEO strategies [2]. More broadly, knowledge of the (characteristics of the) LLM judge opens up ways to manipulate benchmarking results. This could lead to situations where a system achieves much higher scores under automatic evaluation with the LLM judge than under manual assessment [6]. This "eval hacking" undermines the purpose of evaluation, which is to accurately assess the *true* utility of a system for users.

Issue #3: Understanding and Mitigating Vulnerabilities of LLM Judges While initial studies demonstrate the *possibility* of manipulating LLM judges, the *extent* of this vulnerability across different LLMs, attack strategies, and IR tasks remains largely unknown. *What specific vulnerabilities do LLM judges exhibit, and how do these vulnerabilities vary across different models and evaluation settings?* Furthermore, *How can we design evaluation protocols that are robust to manipulation, ensuring that LLM-based evaluation remains a reliable and trustworthy measure of system performance?* This is a crucial area for future research.

3.2.2 Systematic Biases. A core challenge in using LLMs for both ranking and evaluation lies in the fundamental similarity of the two tasks: both involve estimating the relevance of a document to a given query. Several studies note the potential for significant systemic biases when LLMs are employed in both roles [14, 30, 42, 54]. In their summary of the LLM4IR workshop, Rahmani et al. [42] state "if we were to use an LLM both as an assessor and as a ranker, we could expect such a model to be favoured over other evaluated models." Faggioli et al. [14] similarly caution that "if the model is used to judge relevance both for annotation and for retrieval, its evaluation would be overinflated, possibly with perfect performance." If both ranking and automatic evaluation are predisposed towards certain types of results, it becomes difficult to identify *truly*

relevant results. This can lead to the suppression of diverse perspectives and the promotion of homogenous content. Novel ranking approaches that deviate from the LLM's inherent understanding of relevance might be unfairly penalized during the assessment phase. This phenomenon shares similarities with "reward hacking" observed in reinforcement learning, where agents exploit loopholes in the reward function to achieve high scores without genuinely solving the underlying task [4]. A particularly concerning form of this bias is *circularity*, where retrieval models are trained on LLM-generated labels [6, 14, 42]. This creates a self-reinforcing loop, where the ranker learns to produce outputs that the LLM judge deems relevant, further amplifying any existing biases.

Issue #4: Interrelated Systemic Biases in LLM-Based Evaluation While the potential for systemic biases in LLM-based IR evaluation is acknowledged, the specific interactions and magnitudes of these biases remain largely unquantified. We identify three interrelated potential biases:

- *Bias Towards LLM-Based Rankers*: LLM judges might favor the output of systems that also employ LLMs for ranking. While intuitively plausible, this bias needs to be systematically investigated and quantified, independent of the content being retrieved.
- *Bias Towards LLM-Generated Text*: LLM judges might exhibit an inherent preference for text generated by LLMs, regardless of the ranking system that retrieved it. This could be due to factors like stylistic similarities, reduced noise, or other characteristics of LLM-generated text. Indeed, studies have observed that LLMs exhibit bias favoring texts generated by the same underlying model [27, 37]. However, there is a significant lack of studies systematically quantifying the extent to which LLM judges favor LLM-generated text in the specific context of IR evaluation.
- *Combined Bias (LLM Ranker + LLM-Generated Text)*: The most complex scenario involves the potential interaction of the two biases above. Dai et al. [10] show that neural retrievers prefer LLM-generated content, but their analysis relies on human judgments, not LLM judges. Does an LLM judge exhibit an even stronger preference for LLM-generated text when it is retrieved by an LLM-based ranker? This synergistic effect, if present, could significantly distort evaluation outcomes.

It thus remains a set of open questions: *To what extent do LLM judges exhibit biases towards (1) LLM-based rankers, (2) LLM-generated text, and (3) the combination of the two? How do these biases interact, and what is their combined impact on IR evaluation?*

4 Experiments

To empirically demonstrate some of the challenges identified in Section 3, we present a series of targeted experiments aimed at investigating the discriminative ability of LLM judgments (Issue #1), the impact of model choice (Issue #2), and systematic biases (Issue #4). Note that, our goal is to provide illustrative evidence of these issues, rather than a comprehensive or exhaustive analysis.

4.1 Experiment Design

We study the classic ad hoc retrieval task where a ranked list of documents are returned in response to a user query. We follow a standard retrieve-then-rerank paradigm, where an initial set of

potentially relevant documents is identified by a fast and efficient first stage retriever, which are then subsequently reranked by a computationally more intensive but more accurate model. Our focus lies specifically in this reranking stage, noting that LLMs may also be used for retrieval [53].

We employ a set of rankers built upon progressively more capable LLMs. This allows us to observe how their performance changes as the underlying LLM technology advances and whether LLM judges indeed exhibit bias toward LLM-based rankers. In a novel methodological approach, we also incorporate "oracle" rankings as reference points of comparison. These oracle rankings leverage ground truth human relevance labels to represent a hypothetical perfect ranking system as well as controlled degradations from this ideal. By intentionally degrading the perfect rankings, we create a spectrum of performance levels against which we can compare our LLM-based rankers as well as test the sensitivity of LLM judges.

For the judging side, we explore the sensitivity of evaluation by employing specific variations of LLM judges within a single model family—a relatively unexplored dimension in prior work. By using these specific variations of LLM judges, we aim to assess the consistency and reliability of LLM-based evaluation and to understand how the choice of LLM judge might influence the predicted effectiveness of different rankers. Crucially, we compare the judgments provided by these LLM judges against human assessments, which we consider as *the* ground truth for relevance.

To further explore the implications of LLM integration across the information lifecycle, we also examine the impact of LLM-assisted content creation on retrieval and evaluation. Specifically, we investigate how AI assistance in document authoring may influence relevance scores assigned by LLM-based rankers and judges.

4.2 Experimental Setup

We utilize the TREC Deep Learning (DL) 2019 and 2020 datasets [7, 8], chosen due to their extensive use in prior research in this area. Both use the MS MARCO v1 passage corpus, which contains 8.8 million passages. We adopt the convention of referring to passages as "documents," even if the unit of retrieval are passages in our experiments. The two datasets contain 43 and 54 queries, respectively, with human relevance annotations by TREC assessors.

Following [40, 51], all comparisons are based on the reranking of the top 100 passages retrieved by BM25 [23]. To ensure a fair comparison between human and LLM judges, we filter results that have not been judged by TREC assessors (instead of treating them as non-relevant). For simplicity, we report only on NDCG@10, which is the official evaluation metric of the DL track.

4.2.1 LLM Judges. For automatic assessment, we use two model generations of a powerful commercial LLM, Gemini, in two sizes within each generation: v1 Nano, v1 Pro, v1.5 Flash, and v1.5 Pro. We use the best applicable prompt[1] in [54] based on the open source implementation UMBRELA [56], with judgments performed on a 4-point scale. We set top-p =1 and the temperature to 0.

4.2.2 LLM Rankers. We consider both supervised and unsupervised LLM-based rankers, in addition to a BM25 baseline:
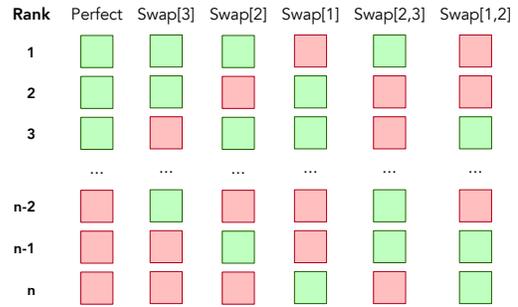

Figure 2: Illustration of oracle rankers, ordered by their expected performance, assuming that the top three results are highly relevant and the bottom three are non-relevant.

- **RankT5** [64] is a reranker that uses T5 [41] and listwise ranking loss during supervised fine-tuning. It is considered a state-of-the-art supervised LLM-based ranker.
- **RG** [20] is a pointwise prompting method based on Relevance Generation, where the prompt asks "Does the passage answer the query?" and the logit of "Yes" is used as the ranking score. We test RG with FLAN-T5-XXL and FLAN-UL2. Note that RG requires internal logits of output tokens and thus cannot be used with black-box LLMs such as Gemini.
- **PRP** [40] is a pairwise prompting approach that is effective and robust across LLMs with different sizes. Given a query and two passages, the prompt asks "Which of the two passages is more relevant to the query?" The winning rate is used as the ranking score for each passage.

4.2.3 Oracle Rankers. We generate oracle rankings using the ground truth TREC relevance assessments. To ensure a fair comparison with LLM rankers, we consider the same initial set of BM25-retrieved documents for reranking. Specifically, we consider the following oracle rankers, which are visualized in Fig. 2:

- **Perfect**: Reranks results according to the ground truth relevance labels. While not perfect overall, this represents the ideal ranking within the initially retrieved set.
- **Swap[i]**: Introduces controlled errors by swapping the top-i ranked result with the bottom-i result. Decreasing i (from 3 to 2 to 1) increases the deviation from the perfect ranking.
- **Swap[i,i+1]**: Swaps the ith and $(i+1)$th highest-ranked results with the ith and $(i+1)$th lowest-ranked results. This represents further degradation from the Swap[i] methods.

4.2.4 Measuring Alignment. Following prior work (cf. Section 3.1.1) we measure agreement with TREC judges (on all human-judged query-document pairs) in terms of Cohen's κ using both graded and binary relevance labels. Following Faggioli et al. [14], we create binary relevance labels by merging levels 0 and 1 (non-relevant) and levels 2 and 3 (relevant). Additionally, we report on relative system ordering in terms of Kendall's τ.

4.3 Results

Table 1 presents the results of the various reranking methods evaluated using both human and LLM judges. Selected methods are

---

[1]We use the prompt that considers multiple aspects (A), but not role (R) nor multiple judges (M); narrative (N) and description (D) are unavailable for TREC DL.

Table 1: Results (NDCG@10) on the TREC DL 2019 and 2020 collections using both human and LLM judges. The best LLM and Oracle reranking approaches per judge are boldfaced.

Method	LLM	TREC DL19					TREC DL20				
		Human judges	LLM judges				Human judges	LLM judges			
			v1 Nano	v1 Pro	v1.5 Flash	v1.5 Pro		v1 Nano	v1 Pro	v1.5 Flash	v1.5 Pro
Initial retrieval											
BM25	-	0.506	0.607	0.772	0.689	0.712	0.483	0.616	0.786	0.689	0.719
LLM reranking											
RankT5	T5 (3B)	0.731	**0.633**	0.907	0.911	0.916	0.696	0.621	0.924	0.888	0.899
RG	FLAN-T5-XXL (11B)	0.673	0.606	0.895	0.874	0.881	0.639	0.619	0.920	0.877	0.878
	FLAN-UL2 (20B)	0.689	0.595	0.896	0.884	0.887	0.667	0.611	0.922	0.880	0.885
PRP	FLAN-T5-XL (3B)	0.716	0.610	0.924	0.921	0.909	0.691	0.618	0.924	0.898	0.901
	FLAN-T5-XXL (11B)	0.712	0.620	0.918	0.922	0.926	0.712	0.615	0.938	0.905	0.912
	FLAN-UL2 (20B)	0.734	0.614	0.923	0.914	0.928	**0.718**	**0.622**	0.932	0.909	0.917
	Gemini v1.5 Flash	**0.747**	0.623	**0.937**	**0.961**	**0.947**	0.699	0.619	**0.952**	**0.937**	**0.933**
Oracle reranking											
Perfect	-	0.892	0.582	**0.896**	**0.876**	**0.864**	**0.871**	0.617	**0.872**	**0.828**	**0.824**
Swap[3]	-	0.824	0.589	0.868	0.835	0.827	0.795	0.611	0.842	0.795	0.796
Swap[2]	-	0.814	0.589	0.859	0.825	0.814	0.790	0.611	0.853	0.797	0.797
Swap[1]	-	0.803	0.578	0.870	0.836	0.836	0.764	**0.621**	0.832	0.778	0.776
Swap[2,3]	-	0.739	**0.596**	0.829	0.779	0.771	0.706	0.602	0.821	0.760	0.765
Swap[1,2]	-	0.713	0.585	0.831	0.782	0.783	0.672	0.615	0.810	0.743	0.746
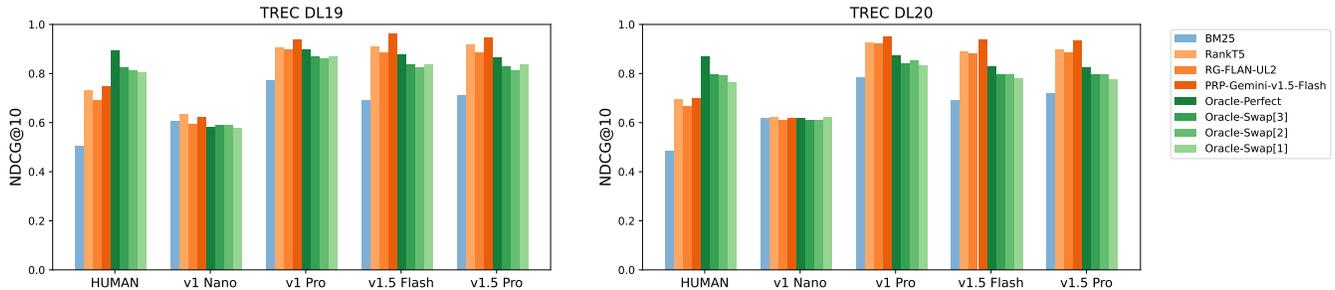
Figure 3: Visualization of the performance of selected rankers from Table 1.

Table 2: Agreement between LLM and human judges (1) on individual relevance judgments (Cohen's κ) using both graded and binary labels and (2) on relative ordering of systems (Kendall's τ) considering all systems in Table 1 and Oracle rankers only.

LLM judge	Cohen's κ				Kendall's τ			
	Graded		Binary		All systems		Oracles-only	
	DL19	DL20	DL19	DL20	DL19	DL20	DL19	DL20
v1 Nano	-0.002	-0.011	0.007	-0.003	-0.253	0.011	-0.067	0.067
v1 Pro	0.139	0.144	0.337	0.273	**0.077**	0.121	**0.600**	**0.867**
v1.5 Flash	**0.268**	**0.230**	0.461	**0.370**	0.033	**0.143**	**0.600**	**0.867**
v1.5 Pro	0.204	0.192	**0.462**	0.359	**0.077**	**0.143**	**0.600**	**0.867**

shown in Fig. 3 for easier visual inspection. Additionally, Table 2 reports on agreement between human and LLM judges.

Choice of LLM *How well do LLM-based judgments align with human assessments when using different variations of LLM judges from the same model family?* Looking at the evaluation scores of various rankers, we observe generally good agreement among the three

largest models. In terms of agreement with human judges on individual relevance judgments (see Cohen's κ Table 2) the results are comparable to those reported in prior work for these datasets [56], with the newer v1.5 models performing clearly better than the v1 models. Interestingly, within this newer model generation, a larger model is not necessarily more capable, at least not according to this measure; v1.5 Flash shows much better agreement with humans

when a graded relevance scale is used than the v1.5 Pro. On the other hand, the smallest LLM (v1 Nano) is unable to provide useful judgments, as evidenced by the Cohen's κ values being close to 0. While this model may be capable in other tasks [16], our results clearly indicate its unsuitability for judging relevance in this specific context. Therefore, we exclude the v1 Nano judge from subsequent analyses and discussions referring to "LLM judges."

Another way to validate LLM judges is by measuring how well they agree on the relative ordering of systems with human judges; see Kendall's τ Table 2. In this regard, the newest and largest model (v1.5 Pro) is the most capable overall, but there is in fact little difference in performance among the three largest models. Thus, while newer model generations clearly perform better (v1 vs. v1.5), larger models with the same generation do not necessarily make more capable judges (v1.5 Flash vs. v1.5 Pro). We also note that differentiating between the entire pool of systems ("All systems") proves to be especially challenging; we will elaborate on this next.

Discriminative Ability *Can LLM judges reliably distinguish between high-performing systems with small, but meaningful, performance differences?* The Oracle rankers, with their controlled performance degradations, enable us to assess the discriminative power of LLMs in a setting free from potential biases introduced by LLM-based rankers. While the absolute score differences between some pairs of Oracle rankings may be small, all pairwise differences are statistically significant according to human judgments (paired t-test, $p < 0.05$). Therefore, a failure to observe a statistically significant difference, or, more critically, a reversal of the correct ordering, indicates that the LLM judge is not sufficiently sensitive. Table 2 (Oracle-only setting) reveals that accurately ordering the Oracle rankings is challenging for LLM judges, particularly on the DL19 dataset. This suggests limitations in their ability to discern subtle, yet statistically significant, performance differences. This limited discriminative ability is not confined to the Oracle setting; it also manifests when evaluating actual retrieval systems. For instance, the v1.5 Pro model, which performed best among the LLMs on the Oracle rankings, fails to identify statistically significant differences between certain pairs of systems (e.g., RankT5 vs. RG-FLAN-T5-XXL on DL19, $p < 0.001$ according to human evaluation). Conversely, it can also identify differences as statistically significant (e.g., PRP-FLAN-UL2 vs. PRP-Gemini-v1.5-Flash, $p < 0.05$ for both years) when human judgments show no significant difference.

Furthermore, the substantial difference in correlation between the "All systems" and "Oracles-only" results in Table 2 provides direct evidence of the concerns raised in Issue #1 (Section 3.1.1), namely, how easily correlation-based metrics can be manipulated by the choice of systems included in the evaluation.

Bias Towards LLM-based Rankers *Do LLM judges exhibit biases towards LLM-based rankers?* The results presented in Fig. 3 demonstrate a clear and substantial bias in favor of LLM-based rankers when evaluated by LLM judges. While prior work has hinted at the *potential* for such a bias, this study provides direct empirical evidence of its existence and magnitude. Human judgments consistently place the selected Oracle rankers shown in Fig. 3 above all LLM-based rankers. LLM judges, however, completely invert
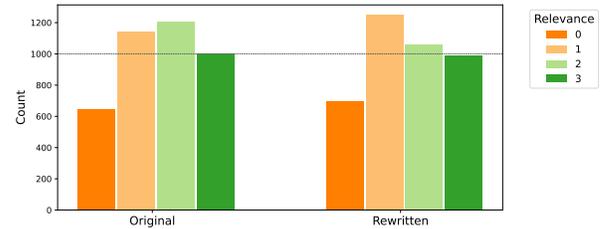

Figure 4: Relevance levels estimated by an LLM judge (Gemini v1.5 Pro) for Original vs. AI-assisted content (Rewrites, using Gemini v1.5 Flash). According to human assessors, the labels should be uniformly distributed across the four relevance classes, as indicated by the dashed horizontal line.

this order, ranking *all* LLM-based rankers as superior to these Oracle runs. This is not a subtle effect; the magnitude of the bias is sufficient to completely reverse the relative ranking of these two fundamentally different types of systems. The fact that the true performance of non-LLM-based systems is severely underestimated, highlighting a critical limitation of relying solely on LLM judges for evaluation, particularly when assessing fundamentally new or unconventional approaches.

Bias Towards LLM-generated Text *Do LLM judges exhibit biases towards LLM-generated text (independent of the ranking mechanism used to retrieve that text)?* We investigate this by comparing LLM judge assessments of original human-written documents and their AI-assisted counterparts. Using the MS MARCO dataset, which predates the widespread adoption of modern AI writing tools, we can reasonably assume that the original documents represent content created without significant AI assistance. To avoid bias potentially introduced by initial retrieval, we take a balanced sample: for each year (DL19 and DL20) we randomly sample 500 query-document pairs for each of the four relevance levels, resulting in a total of 4000 query-document pairs. We refer to this set as Original. We then employ our second most capable LLM (Gemini v1.5 Flash) to create an AI-rewritten version of each document in the Original set, following the methodology of Dai et al. [10].[2] This rewritten set is referred to as Rewritten. We shall assume that this rewriting process does not substantially alter the relevance of the documents to their corresponding queries, as verified by human assessors in [10]. Figure 4 presents the results using our most capable LLM (Gemini v1.5 Pro) as the judge. We can observe on the Original data that the LLM judge is lenient in its assessment of relevance, and specifically in labeling non-relevant documents as partially relevant. However, the judge does not appear to systematically inflate scores for the highest relevance level. Crucially, when comparing these results to the judgments on the Rewritten (LLM-generated) text, we do not observe a distributional shift towards higher relevance levels. In fact, the Rewritten documents show a slight increase in lower relevance labels. While these findings are specific to this particular combination of LLM rewriter and judge, they provide evidence against a general bias towards LLM-generated content, even when both models are from the same family.

---

[2]They use the straightforward prompt "Please rewrite the following text: human-written text" in a zero-shot setting.
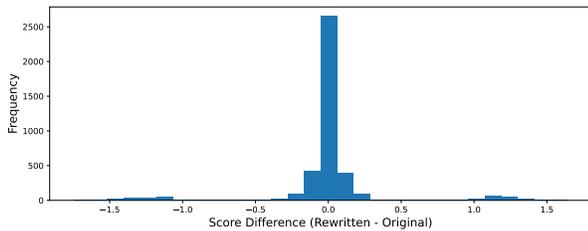
Figure 5: Distribution of score differences of a RankT5 ranker on LLM-rewritten vs. original text on a sample of 4000 query-document pairs.

It is important to note that the preceding analysis examines the *distributional* impact of LLM-generated text on relevance judgments. To further investigate potential biases in a ranking context, we conduct a second experiment. We take the rankings produced by the perfect Oracle method and re-evaluate them using our LLM judge (Gemini v1.5 Pro). However, instead of using the Original document content, we substitute the Rewritten versions. If the LLM judge exhibited a strong preference for LLM-generated text, we would expect to see a significant increase in the scores assigned to these rankings. However, according to our results, that is not the case. We find that the performance of the Perfect Oracle method, as assessed by the LLM judge, does not change significantly when using the Rewritten text instead of the Original text: we get an NDCG@10 of 0.868 vs. 0.883 on DL19 and 0.825 vs. 0.818 on DL20 for Rewritten vs. Original; none of these differences is statistically significant. This further reinforces the conclusion that, at least in this experimental setup, the LLM judge does not exhibit a strong bias towards LLM-generated content.

Combined Bias (LLM Ranker + LLM-Generated Text) *Do LLM judges exhibit biases towards LLM-generated text when using LLM-based rankers?* To address this question, we conduct an experiment combining an LLM-based ranker with LLM-generated text and an LLM judge. We utilize the same balanced sample of 4,000 query-document pairs (500 per relevance level for each of DL19 and DL20) used in the previous experiment, comprising both the Original and Rewritten sets. For ranking, we employ a pointwise approach using RankT5 with a Flan-T5-XXL model. We compare the scenarios where the RankT5 model scores (1) the Original query-document pairs and (2) the Rewritten query-document pairs. Both scorings are then evaluated using the same LLM judge (Gemini v1.5 Pro). We observe minimal differences in the LLM-assigned evaluation scores between the Original and Rewritten scenarios. Closer inspection of the RankT5 scores reveals that the rewriting process had a negligible impact on retrieval scores for the vast majority of query-document pairs. The few observed changes were symmetrically distributed, with increases and decreases in scores mirroring each other; see Fig. 5. This aligns with the previous experiment's findings, suggesting that neither the LLM judge nor the LLM ranker (in this specific configuration) exhibits a strong preference for the LLM-rewritten content. Consequently, the LLM judge produces very similar evaluation results in both cases. While these results do not demonstrate a combined bias in this specific experimental setup, the potential for synergistic effects between LLM rankers,

LLM-generated text, and LLM judges remains an open question requiring further, more comprehensive investigation.

5 The Role and Challenges of LLM Judges in IR

The feasibility of LLMs as automatic relevance assessors has been established and they have been rapidly adopted both in academia and in industry. The question, therefore, is not whether they *can* be used as judges, but rather *how* they should be used in a principled and effective manner. This requires a careful consideration of both the intended purpose of LLM judges and their inherent limitations.

Clarifying the Purpose Most studies, albeit often implicitly, employ LLMs as judges with the aim of *replacing* human assessors. However, as we discuss below, fundamental limitations preclude this possibility. We argue (in line with [14, 30]) that a more appropriate and productive goal should be to enable more effective use of limited human assessor time and resources. This shift in perspective—from replacement to reducing human effort—is crucial for guiding the development and deployment of LLM judges.

Acknowledging Fundamental Limitations It is essential to recognize that both ranking and relevance assessment address the same problem: predicting the relevance of a document to a given query. This inherent overlap introduces fundamental limitations when using LLMs for both tasks. Clarke and Dietz [6] argue that "A true gold standard must originate from human assessments, as only humans can determine the relevance of information in a way that reflects real-world utility." We must also recognize that relevance itself carries an intrinsic uncertainty; it depends on the entire cognitive state of the person, which changes as they use the system [46]. Because of these inherent limitations, "LLM assessments may themselves represent a strong ranking method, rather than a valid evaluation metric" [6]. Recent work, such as [36], has begun to provide uncertainty measures with LLM relevance predictions.

What, then, is the Role of LLM Judges? Given the limitations outlined above, and recognizing that information access systems are ultimately built to serve *human* needs and provide utility to *users*, the highest-fidelity evaluation of these systems can only be achieved through online evaluation with real users. Offline evaluation, while valuable, remains an abstraction of the real-world task because it removes the user from the evaluation process. There is a genuine risk that findings from offline experiments, particularly those relying solely on LLM judges, may not translate to operational settings. It is crucial to recognize that the signal provided by LLM judges is inherently a noisy and potentially biased one, and therefore cannot be fully trusted as a direct proxy for utility. Nevertheless, this noisy signal can be a useful indicator, helping to identify which methods or system variants are promising enough to warrant the more resource-intensive process of human evaluation.

5.1 Guidelines for Employing LLMs as Judges

We outline several key considerations for employing LLMs in evaluation, aiming to foster a community-wide set of best practices and ensure methodological soundness. These are not exhaustive, but represent an important starting point.

- *Consistent Evaluation Across Systems.* All systems being compared within a single evaluation should be assessed using the same LLM judge configuration (model, prompt, settings). This ensures a fair and unbiased comparison, avoiding situations where some systems are evaluated with a more lenient or biased judge than others. Specifically, LLM judges should not be used selectively to fill "relevance holes" in existing human judgments [1].
- *Transparency and Reproducibility.* To enable reproducibility and facilitate comparisons across studies, researchers should clearly report the specific LLM used (model version), the exact prompt(s) employed, and any relevant settings or parameters.
- *Employing Multiple LLMs as Judges.* Using a combination of different LLM judges can help mitigate biases stemming from LLMs favoring responses from their own model family and improve robustness (see, e.g., [19]). Reporting the distribution of scores across different judges, as suggested by Rahmani et al. [42], can further enhance robustness.
- *Alignment with Human Preferences.* Ensuring alignment between human and LLM raters is a substantial effort that needs to be continuously monitored and refined [54]. Ideally, results reported on LLM judges should also include human validation of the results on a representative sample. Researchers should also exercise care when making research claims based on results from LLM judges.

5.2 Open Questions and Future Directions

The adoption of LLMs as judges in IR presents several open questions and necessitates further research to address the limitations and biases identified in previous sections.

- *Assessing and Improving LLM Judge Quality.* Our findings highlight the critical importance of LLM judge quality, revealing shortcomings in their discriminative ability and biases toward LLM-powered rankers. Developing robust methods for assessing and improving the quality of LLM judges is a crucial research direction for the IR community, potentially drawing motivations from horizontal autorater efforts [57].
- *Robustness Against Adversarial Attacks.* LLM judges, similar to LLM-based rankers, are susceptible to adversarial attacks, including keyword stuffing and content injection [2, 52]. Understanding these vulnerabilities and developing effective mechanism to enhance the robustness of LLM judges against them are critical areas for ensuring the practical applicability of LLM judges in real-world scenarios.
- *Human-in-the-Loop LLM Judges.* The potential for using LLMs to augment human assessors, e.g., for quality-control, has been suggested [14, 48]. While Upadhyay et al. [55] found that human-in-the-loop processes did not bring obvious tangible benefits, their study represents a preliminary investigation. Further research is needed to explore the potential of this approach more comprehensively.
- *From Passages to Longer Documents.* Most of the existing work focuses on paragraphs as the unit of retrieval, using either the TREC DL [2, 5, 14, 30] or RAG [55] benchmarks. There is much less work on ad hoc retrieval, with exceptions including TREC-8 [14] and Robust [54]. It is known that LLMs handle long context differently [24] and its implication in judging long documents need further investigation.

- *Alternative Judging Approaches.* All existing studies apply LLMs in a pointwise manner, but a pairwise or listwise setup would also be possible. LLM judge research can borrow ideas from LLM ranking research where pairwise and listwise approaches are extensively explored.
- *Domain-specific Solutions.* While most existing research focuses on general-purpose search, specific domains might require purpose-built solutions. Recent work, for example, has explored the use of LLM judges for e-commerce search [32, 45]. Extending this concept of domain specialization, the applicability and value of LLM judges in specialized domains requiring expert knowledge (e.g., medical or legal search) remain less clear. In such domains, LLM judges might offer potential cost savings and more in-depth domain-specific knowledge compared to non-expert human assessors, but they also introduce new challenges, including the need for high accuracy, the potential for serious consequences from errors, and the complexities of expert judgment. Future research should explore the use of LLM judges in these contexts, carefully considering the trade-offs between cost, efficiency, and the risks of inaccurate assessments.
- *Smaller, Purpose-Built Models.* The use of LLMs for judging at large scale raises concerns about computational cost and latency. A promising research direction is to explore the development of smaller models, designed specifically for the judging task. These purpose-built models could potentially offer significant advantages in terms of efficiency and speed, while maintaining performance comparable to massive LLMs in terms of accuracy and reliability.
- *Internationalization.* Most related research focuses on English-language corpora. The issues discussed in this paper may be amplified in other languages, especially low resource ones, due to limitations in LLMs' multilingual capabilities. Further research is needed to evaluate the performance of LLM judges across a wider range of languages.
- *Training Models on LLM-generated Labels.* Training retrieval models on data labeled by LLMs introduces a significant risk of circularity and bias amplification. If done recursively, this might lead to model collapse [47]. Thomas et al. [54] acknowledge that parts of the Bing search engine are retrained using LLM-generated labels. Understanding the long-term effects of such training is an important research direction.

6 Conclusion

This paper has investigated the emerging and critical challenge of understanding the effect LLM-based rankers and AI-powered content creation may have on LLM-based judges' ability to accurately assess relevance. Through a synthesis of existing literature, we identified key concerns regarding the quality, validity, reliability, and potential biases of LLM judgments. Our experiments provided empirical evidence demonstrating how interactions between the various roles LLMs play can lead to inaccurate or biased assessments of retrieval effectiveness, particularly in scenarios involving LLM-based rankers. Finally, we presented guidelines for the use of LLMs as judges in IR and outlined a research agenda to address crucial open questions in this rapidly evolving field.

References

[1] Zahra Abbasiantaeb, Chuan Meng, Leif Azzopardi, and Mohammad Aliannejadi. 2024. Can We Use Large Language Models to Fill Relevance Judgment Holes?. In *Joint Proceedings of the 1st Workshop on Evaluation Methodologies, Testbeds and Community for Information Access Research (EMTCIR 2024) and the 1st Workshop on User Modelling in Conversational Information Retrieval (UM-CIR 2024) co-located with the 2nd International ACM SIGIR Conference on Information Retrieval in the Asia Pacific (SIGIR-AP 2024), Tokyo, Japan, December 12, 2024 (CEUR Workshop Proceedings, Vol. 3854)*.

[2] Marwah Alaofi, Paul Thomas, Falk Scholer, and Mark Sanderson. 2024. LLMs can be Fooled into Labelling a Document as Relevant. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP '24)*. 32–41.

[3] Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. 2022. InPars: Unsupervised Dataset Generation for Information Retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. 2387–2392.

[4] Lichang Chen, Chen Zhu, Jiuhai Chen, Davit Soselia, Tianyi Zhou, Tom Goldstein, Heng Huang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. ODIN: Disentangled Reward Mitigates Hacking in RLHF. In *Forty-first International Conference on Machine Learning (ICML '24)*.

[5] Nuo Chen, Jiqun Liu, Xiaoyu Dong, Qijiong Liu, Tetsuya Sakai, and Xiao-Ming Wu. 2024. AI Can Be Cognitively Biased: An Exploratory Study on Threshold Priming in LLM-Based Batch Relevance Assessment. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP '24)*. 54–63.

[6] Charles L. A. Clarke and Laura Dietz. 2024. LLM-based relevance assessment still can't replace human relevance assessment. arXiv:2412.17156 [cs.IR]

[7] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2020. Overview of the TREC 2020 Deep Learning Track. In *Proceedings of the Twenty-Ninth Text REtrieval Conference (TREC '20)*.

[8] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2019. Overview of the TREC 2019 Deep Learning track. In *Proceedings of the Twenty-Eighth Text REtrieval Conference (TREC '19)*.

[9] Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. 2024. Bias and Unfairness in Information Retrieval Systems: New Challenges in the LLM Era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24)*. 6437–6447.

[10] Sunhao Dai, Yuqi Zhou, Liang Pang, Weihao Liu, Xiaolin Hu, Yong Liu, Xiao Zhang, Gang Wang, and Jun Xu. 2024. Neural Retrievers are Biased Towards LLM-Generated Content. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24)*. 526–537.

[11] Zhuyun Dai, Vincent Y Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith Hall, and Ming-Wei Chang. 2023. Promptagator: Few-shot Dense Retrieval From 8 Examples. In *The Eleventh International Conference on Learning Representations (ICLR '23)*.

[12] Andrew Drozdov, Honglei Zhuang, Zhuyun Dai, Zhen Qin, Razieh Rahimi, Xuanhui Wang, Dana Alon, Mohit Iyyer, Andrew McCallum, Donald Metzler, and Kai Hui. 2023. PaRaDe: Passage Ranking using Demonstrations with LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2023 (EMNLP '23)*. 14242–14252.

[13] Yann Dubois, Percy Liang, and Tatsunori Hashimoto. 2024. Length-Controlled AlpacaEval: A Simple Debiasing of Automatic Evaluators. In *First Conference on Language Modeling (COLM '24)*.

[14] Guglielmo Faggioli, Laura Dietz, Charles L. A. Clarke, Gianluca Demartini, Matthias Hagen, Claudia Hauff, Noriko Kando, Evangelos Kanoulas, Martin Potthast, Benno Stein, and Henning Wachsmuth. 2023. Perspectives on Large Language Models for Relevance Judgment. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval (ICTIR '23)*. 39–50.

[15] Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. Bias and Fairness in Large Language Models: A Survey. *Computational Linguistics* 50, 3 (Sept. 2024), 1097–1179.

[16] Gemini Team Google. 2023. Gemini: A family of highly capable multimodal models. arXiv:2312.11805 [cs.CL]

[17] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences* 120, 30 (2023), e2305016120.

[18] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2024. A Survey on LLM-as-a-Judge. arXiv:2411.15594 [cs.CL]

[19] Alon Jacovi, Andrew Wang, Chris Alberti, Connie Tao, Jon Lipovetz, Kate Olszewska, Lukas Haas, Michelle Liu, Nate Keating, Adam Bloniarz, Carl Saroufim, Corey Fry, Dror Marcus, Doron Kukliansky, Gaurav Singh Tomar, James Swirhun, Jinwei Xing, Lily Wang, Madhu Gurumurthy, Michael Aaron, Moran Ambar, Rachana Fellinger, Rui Wang, Zizhao Zhang, Sasha Goldshtein, and Dipanjan Das. 2025. The FACTS Grounding Leaderboard: Benchmarking LLMs' Ability to Ground Responses to Long-Form Input. arXiv:2501.03200 [cs.CL]

[20] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2023. Holistic Evaluation of Language Models. *Transactions on Machine Learning Research* (2023).

[21] Paul Pu Liang, Chiyu Wu, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2021. Towards understanding and mitigating social biases in language models. In *International Conference on Machine Learning (ICML '21)*. 6565–6576.

[22] Weixin Liang, Yaohui Zhang, Mihai Codreanu, Jiayu Wang, Hancheng Cao, and James Zou. 2025. The Widespread Adoption of Large Language Model-Assisted Writing Across Society. arXiv:2502.09747 [cs.CL]

[23] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python Toolkit for Reproducible Information Retrieval Research with Sparse and Dense Representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21)*. 2356–2362.

[24] Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics* 12 (2024), 157–173.

[25] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP '23)*. 2511–2522.

[26] Yang Liu and Mirella Lapata. 2019. Text Summarization with Pretrained Encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3730–3740.

[27] Yiqi Liu, Nafise Moosavi, and Chenghua Lin. 2024. LLMs as Narcissistic Evaluators: When Ego Inflates Evaluation Scores. In *Findings of the Association for Computational Linguistics: ACL 2024*. 12688–12701.

[28] Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2024. Aligning with Human Judgement: The Role of Pairwise Preference in Large Language Model Evaluators. In *First Conference on Language Modeling (COLM '24)*.

[29] Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023. Zero-Shot Listwise Document Reranking with a Large Language Model. arXiv:2305.02156 [cs.IR]

[30] Sean MacAvaney and Luca Soldaini. 2023. One-Shot Labeling for Automatic Relevance Estimation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*. 2230–2235.

[31] J. S. McCarley, Rishav Chakravarti, and Avirup Sil. 2019. Structured Pruning of a BERT-based Question Answering Model. arXiv:1910.06360 [cs.CL]

[32] Navid Mehrdad, Hrushikesh Mohapatra, Mossaab Bagdouri, Prijith Chandran, Alessandro Magnani, Xunfan Cai, Ajit Puthenputhussery, Sachin Yadav, Tony Lee, ChengXiang Zhai, and Ciya Liao. 2024. Large Language Models for Relevance Judgment in Product Search. arXiv:2406.00247 [cs.IR]

[33] Roberto Navigli, Simone Conia, and Björn Ross. 2023. Biases in large language models: origins, inventory, and discussion. *ACM Journal of Data and Information Quality* 15, 2 (2023), 1–21.

[34] Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. arXiv:1901.04085 [cs.IR]

[35] Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. 2020. Document ranking with a pretrained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. 708–718.

[36] Harrie Oosterhuis, Rolf Jagerman, Zhen Qin, Xuanhui Wang, and Michael Bendersky. 2024. Reliable confidence intervals for information retrieval evaluation using generative ai. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24)*. 2307–2317.

[37] Arjun Panickssery, Samuel R. Bowman, and Shi Feng. 2024. LLM Evaluators Recognize and Favor Their Own Generations. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems (NeurIPS '24)*.

[38] Andrew Parry, Maik Fröbe, Sean MacAvaney, Martin Potthast, and Matthias Hagen. 2024. Analyzing Adversarial Attacks on Sequence-to-Sequence Relevance Models. In *Proceedings of the 46th European Conference on Information Retrieval (ECIR '24)*. 286–302.

[39] Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. 2021. The Expando-Mono-Duo Design Pattern for Text Ranking with Pretrained Sequence-to-Sequence Models. arXiv:2101.05667 [cs.IR]

[40] Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2024. Large Language Models are Effective Text Rankers with Pairwise Ranking Prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*. 1504–1518.

[41] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* 21, 140 (2020), 1–67.

[42] Hossein A. Rahmani, Clemencia Siro, Mohammad Aliannejadi, Nick Craswell, Charles L. A. Clarke, Guglielmo Faggioli, Bhaskar Mitra, Paul Thomas, and Emine Yilmaz. 2024. Report on the 1st Workshop on Large Language Model for Evaluation in Information Retrieval (LLM4Eval 2024) at SIGIR 2024. arXiv:2408.05388 [cs.IR]

[43] Hossein A. Rahmani, Xi Wang, Emine Yilmaz, Nick Craswell, Bhaskar Mitra, and Paul Thomas. 2025. SynDL: A Large-Scale Synthetic Test Collection for Passage Retrieval. arXiv:2408.16312 [cs.IR]

[44] Devendra Sachan, Mike Lewis, Mandar Joshi, Armen Aghajanyan, Wen-tau Yih, Joelle Pineau, and Luke Zettlemoyer. 2022. Improving Passage Retrieval with Zero-Shot Question Generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP '22)*. 3781–3797.

[45] Jayant Sachdev, Sean D. Rosario, Abhijeet Phatak, He Wen, Swati Kirti, and Chittaranjan Tripathy. 2025. Automated Query-Product Relevance Labeling using Large Language Models for E-commerce Search. arXiv:2502.15990 [cs.IR]

[46] Tefko Saracevic. 1996. Relevance reconsidered. In *Proceedings of the second conference on conceptions of library and information science (CoLIS 2)*. 201–218.

[47] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross J. Anderson, and Yarin Gal. 2024. AI models collapse when trained on recursively generated data. *Nature* 631, 8022 (July 2024), 755–759.

[48] Ian Soboroff. 2025. Don't Use LLMs to Make Relevance Judgments. *Information Retrieval Research* 1 (Mar. 2025), 29–46.

[49] Heydar Soudani, Roxana Petcu, Evangelos Kanoulas, and Faegheh Hasibi. 2024. A Survey on Recent Advances in Conversational Data Generation. arXiv:2405.13003 [cs.CL]

[50] Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large Language Models are Inconsistent and Biased Evaluators. arXiv:2405.01724 [cs.CL]

[51] 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 '23)*. 14918–14937.

[52] Manveer Singh Tamber and Jimmy Lin. 2025. Illusions of Relevance: Using Content Injection Attacks to Deceive Retrievers, Rerankers, and LLM Judges. arXiv:2501.18536 [cs.IR]

[53] Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022. Transformer memory as a differentiable search index. In *Proceedings of the 36th International Conference on Neural Information Processing Systems (NeurIPS '22)*. 21831–21843.

[54] Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2024. Large Language Models can Accurately Predict Searcher Preferences. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24)*. 1930–1940.

[55] Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Daniel Campos, Nick Craswell, Ian Soboroff, Hoa Trang Dang, and Jimmy Lin. 2024. A Large-Scale Study of Relevance Assessments with Large Language Models: An Initial Look. arXiv:2411.08275 [cs.IR]

[56] Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Nick Craswell, and Jimmy Lin. 2024. UMBRELA: UMbrela is the (Open-Source Reproduction of the) Bing RELevance Assessor. arXiv:2406.06519 [cs.IR]

[57] Tu Vu, Kalpesh Krishna, Salaheddin Alzubi, Chris Tar, Manaal Faruqui, and Yun-Hsuan Sung. 2024. Foundational Autoraters: Taming Large Language Models for Better Automatic Evaluation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP '24)*. 17086–17105.

[58] Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023. How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NeurIPS '23)*. 74764–74786.

[59] Wenda Xu, Guanglei Zhu, Xuandong Zhao, Liangming Pan, Lei Li, and William Wang. 2024. Pride and Prejudice: LLM Amplifies Self-Bias in Self-Refinement. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL '24)*. 15474–15492.

[60] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. In *8th International Conference on Learning Representations (ICLR '20)*.

[61] Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text Generation Evaluating with Contextualized Embeddings and Earth Mover Distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP '19)*. 563–578.

[62] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and Chatbot Arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NeurIPS '23)*.

[63] Jinhua Zhu, Yingce Xia, Lijun Wu, Di He, Tao Qin, Wengang Zhou, Houqiang Li, and Tieyan Liu. 2020. Incorporating BERT into Neural Machine Translation. In *International Conference on Learning Representations (ICLR '20)*.

[64] Honglei Zhuang, Zhen Qin, Rolf Jagerman, Kai Hui, Ji Ma, Jing Lu, Jianmo Ni, Xuanhui Wang, and Michael Bendersky. 2023. RankT5: Fine-Tuning T5 for Text Ranking with Ranking Losses. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*. 2308–2313.