On the Use of LLMs for Relevance Labelling

MARWAH ALAOFI*, Taibah University, Saudi Arabia
PAUL THOMAS, Microsoft, Australia
FALK SCHOLER, RMIT University, Australia
MARK SANDERSON, RMIT University, Australia

Large Language Models (LLMs) are increasingly used to replace human judges to assess the relevance of information objects, raising concerns about circularity, bias, and whether simulated preferences can substitute for human judgement. This work presents experiments using multiple LLMs to label passages for relevance. It examines their gullibility – how easily they are misled into labelling irrelevant passages as relevant. It also compares LLMs with human judges in ranking systems, analysing differences in discriminative power and whether some systems benefit under LLM-based evaluation. Results show that LLMs are influenced by the presence of query terms, even with irrelevant or random passages. Moreover, LLM-generated rankings are highly correlated with those of human judges, with strong agreement on which system is better in pairwise comparisons. However, LLMs may exhibit lower discriminative power, as seen in flatter ranking slopes and missed significance for meaningful improvements. Yet, there are no cases where capable LLMs and human judges reach opposing conclusions with significance. LLMs may boost traditional systems more than neural ones, adding a new concern of system bias. These findings highlight the strong potential of LLMs for relevance labelling, while also highlighting failure cases that call for careful adoption and further research to maintain evaluation integrity.[1]

CCS Concepts: • **Information systems → Evaluation of retrieval results**; **Relevance assessment**; **Test collections**.

Additional Key Words and Phrases: Information retrieval; test collections; relevance judgements; LLMs

1 Introduction

Creating relevance judgements – the process of assessing the relevance of documents to a given search query – is the most labour-intensive task in creating test collections. Research has examined the use of Large Language Models (LLMs) to assess the relevance of documents, with recent attempts [1, 12, 25, 42, 43] showing promising results for using LLMs in generating relevance judgements (or "labels", to distinguish them from human "judgements"). The use of LLMs has become more common, with the TREC 2024 Retrieval-Augmented Generation (RAG) Track using an LLM to evaluate the retrieval component of RAG systems [44].

Unlike human judgement of relevance, relevance labels produced by LLMs are independent of the documents seen previously; i.e., each document is labelled entirely independently of others. They are also considerably cheaper to collect than using human judges. However, they raise multiple concerns, including circularity (where the same LLM may be used for both retrieval and evaluation [5]), bias toward certain systems (e.g., LLM re-rankers) or content (e.g.,

*Most of the research reported in this paper was carried out during the author's affiliation with RMIT University, Melbourne, Australia.
[1]Data and scripts are available at: https://github.com/MarwahAlaofi/humans-vs-LLM-as-judge/

Authors' Contact Information: Marwah Alaofi, Taibah University, Medina, Saudi Arabia, maofi@taibahu.edu.sa; Paul Thomas, Microsoft, Adelaide, Australia, pathom@microsoft.com; Falk Scholer, RMIT University, Melbourne, Australia, falk.scholer@rmit.edu.au; Mark Sanderson, RMIT University, Melbourne, Australia, mark.sanderson@rmit.edu.au.

Manuscript submitted to ACM

LLM-generated text [4]), and the imposition of an artificial ceiling on potential improvements – i.e., gains beyond the LLM judge's ability may not be recognised [40]. While some of these concerns have been initially explored, the impact of LLMs on evaluation remains an evolving area with competing views and findings, highlighting the need for further analysis and methodological development to understand the implications of adopting LLMs as judges in evaluating Information Retrieval (IR) systems.

Most of the available literature evaluates the reliability of LLM relevance labels primarily through their agreement with human judgements – using agreement metrics such as Cohen's κ – or through their similarity to human judges in system rankings, for instance by measuring Kendall's τ over TREC run rankings. Based on these aggregate metrics, such studies typically conclude whether LLMs can replace human judges. In contrast, examining failure cases and the systematic errors LLMs make when labelling passages – and how these errors affect the final ranking of systems and the decisions about which systems to adopt – remains under-examined in existing studies.

This study explores dimensions that may be overlooked when substituting human judges with LLMs, such as systematic errors that influence their labelling of passages and the ultimate ranking of systems. It introduces more fine-grained metrics to help researchers and practitioners better assess the reliability of adopting LLMs for relevance labelling. One of these metrics answers how vulnerable LLMs are to superficial or misleading signals when making relevance judgements, i.e., their *gullibility*. We define gullibility as the tendency of an LLM to assess relevance based on the mere presence of query words, or when explicitly prompted to label a document as relevant, rather than on genuine semantic relevance. Measuring gullibility is important because if LLMs are influenced by surface-level signals, they may negatively affect system evaluation and, more importantly, become vulnerable to adversarial prompts or content that can change evaluation outcomes and potentially impact downstream tasks.

Formally, the study investigates the following two main questions:

RQ1 How similar are LLMs to human judges in generating relevance labels?
RQ2 How similar are LLMs to human judges in system ranking?

The first experiment, described in Section 3, addresses **RQ1** by examining the agreement of various open-source and proprietary LLMs with human judges in labelling short texts (i.e., passages) for relevance. It establishes a cost-accuracy trade-off, examines the LLM's sensitivity to different prompts, analyses their labelling patterns, and – most importantly – investigates instances where LLMs and human judgements differ, with the goal of formulating and empirically testing hypotheses about the causes of LLM failures.

The second experiment, presented in Section 4, addresses **RQ2** by investigating how LLMs and human judges differ in ranking TREC runs for Deep Learning Track of TREC 2021 (DL21) and Deep Learning Track of TREC 2022 (DL22). We move beyond rank correlation to examine differences in discriminative power and identify whether certain runs gain an advantage under LLM-based evaluation.

The key contributions of this work are as follows:

C1 The proposal of multiple "gullibility" tests and metrics to expose some of the limitations of LLMs that can be hidden behind traditional metrics.
C2 An empirical evaluation of the quality, gullibility, and cost of multiple open-source and proprietary LLMs for relevance labelling.
C3 A detailed analysis of how the ranking of TREC runs varies from the ranking generated by humans. This includes ranking correlation and differences in discriminative power and an investigation into unknown potential system biases, demonstrating how LLM failures manifest in their ability to rank and distinguish between systems.

C4 A publicly available evaluation of TREC DL21 and DL22 runs using nine open-source and proprietary LLMs, aimed at facilitating research into the differences between LLMs and human judgements.

The paper is organised as follows: Section 2 reviews related work on using LLMs for relevance labelling. Sections 3 and 4 present experiments on labelling agreement and system ranking agreement, respectively. Section 5 concludes with a discussion and summary of findings.

2 Related Work

Human relevance judgements have been studied extensively in the literature. Notably, human judges tend to lack consistency in assessing document relevance [e.g. 37–39, 41]. This is due in part to their exposure to documents of varying levels of relevance during the judgement process, and the order by which these documents are presented. Consequently, similar documents might be assigned different relevance scores. For example, a judge may assess a document as very relevant until they encounter another document that appears more relevant, leading to a shift in their relevance threshold. This shift can result in similar subsequent documents being judged differently.

The stability of evaluation metrics and statistical power under varying sets of judgements, judgement collection procedures, and assessor-specific characteristics has been of interest in the IR community [e.g. 35–37, 48]. Rashidi et al. [35] examined the inconsistencies in human relevance judgements on the reliability of IR evaluation metrics. Sakai et al. [37] demonstrated that the presentation order of documents, the expertise of assessors, and their motivation influence relevance judgements. Roitero et al. [36] proposed a method to estimate the minimum number of crowd workers needed per document to ensure sufficient statistical power in crowdsourced relevance assessments.

Several studies examined the use of LLMs to assess the relevance of documents [1, 12, 25, 33, 42, 43], showing promising results – albeit with some pitfalls – that challenge what was previously perceived as impossible to automate. Notably, LLMs have been demonstrated to agree with humans at the same level of human-to-human agreement for relevance labelling [1, 4, 43] and generate similar rankings of systems [4, 42, 43]. Early work by Faggioli et al. [12] was the first to explore the potential of using LLMs for relevance labelling. Microsoft Bing later announced a large-scale in-house use of GPT-4 for generating relevance labels, reporting that it generates relevance labels of higher quality than those produced by crowdworkers – and at a fraction of the cost. The quality of these labels is monitored against gold labels, and they are used to train better re-rankers [42]. This capability enables many applications previously considered impractical or impossible, such as assessing the relevance of private documents that cannot be shared with human judges or creating larger-scale test collections without the need for a traditional pooling process.

This capability has prompted ongoing debate within the research community. Many researchers remain cautious – raising concerns about LLMs's validity, reliability, and reproducibility, and whether there is sufficient evidence to support their reliable use as substitutes for human judges [4, 5, 11, 40], urging the community not to overstate their effectiveness based merely on observed agreement with human judges in relevance labels and system ranking.

In examining the limitations of LLMs for relevance labelling, research suggests that LLMs can be more liberal than human judges: they are more likely to judge a document as relevant compared to humans [1], possibly leading to inflated system evaluation scores. Although their system rankings often show high correlation with human-generated rankings – typically interpreted in relation to a reference Kendall's τ that was established between human judges [46] – this metric has notable limitations when adopted across test collections comparing different sets of systems. After all, both labelling agreement and ranking correlation can still serve as proxies for LLM–human similarity, but they may obscure deeper concerns when assessing whether the use of LLMs for relevance labelling is a valid option.

The risk of circularity – where the same LLM is used both to generate or re-rank a run and to also train or evaluate it – was demonstrated by Clarke and Dietz [5]. They submitted a run to a TREC track that was produced using the same LLM employed as the judge, aiming to subvert the automated evaluation process. When evaluated using an LLM, the run ranked fifth, but it dropped to 28th under manual evaluation. Balog et al. [4] also demonstrated that Google's LLMs, when used as relevance judges, tend to favour IR systems that incorporate LLMs as re-rankers, over oracle systems that outperform them under human judgement. In contrast, an analysis of TREC DL23 runs by Rahmani et al. [34], involving runs that used T5 and GPT-4, found no evidence that they are favoured when either model is used for relevance labelling. The risk of circularity is also connected to a broader concern articulated by Soboroff [40], who warns against using LLMs to generate relevance judgements: doing so limits what can be measured to the capabilities of the LLM itself.

When examining potential bias toward LLM-generated documents – specifically, whether such documents are evaluated more positively than human-written documents, Balog et al. [4] suggest that LLM judges do not exhibit a particular preference for LLM-generated documents, as the distribution of relevance scores for LLM-rewritten passages is similar to that of the originals. However, as LLMs are increasingly used in retrieval pipelines, such as query generators [e.g. 2, 27, 34], new forms of bias may emerge. For example, using LLMs to generate query variants or query expansions might lead LLM judges to rate LLM-generated documents more positively due to the alignment between the query and the document.

On the discriminative ability of LLM judges among retrieval systems, Balog et al. [4] demonstrate that LLMs struggle to distinguish between top oracle systems. Their results show low Kendall's τ and reveal both failures to detect significant differences and incorrect identification of significance in pairwise comparisons when using their top LLM judge (Gemini 1.5 Pro). However, the study relies on Kendall's τ and presents only illustrative examples of such disagreements, without quantifying how often they occur. In addition, Otero et al. [33], through the use of top-weighted metrics of ranking correlation and pairwise system comparisons, demonstrated that LLMs struggle particularly with ranking top-performing systems and exhibit a high rate of false positives, often overestimating differences in pairwise comparisons.

It is important to note that the reported findings of these studies are specific to particular LLMs, configurations and experimental setups, and should not be generalised to all LLMs or evaluation contexts. As this area continues to evolve, a deeper understanding is still emerging regarding how LLMs behave in evaluation settings.

This paper continues this line of research by investigating several aspects of the adoption of LLMs as judges. It examines their failure patterns and their gullibility to manipulation to label non-relevant text as relevant – an aspect that has received no attention in the literature. Additionally, it provides a detailed analysis of how LLM ranking of systems using graded relevance labels is different to that of humans, exploring whether certain runs are systematically evaluated differently. This extends the discussion beyond the common concern that LLMs may favour neural or LLM-based re-rankers. The paper also assesses the discriminative ability of LLMs to distinguish between systems and evaluates the extent to which their conclusions align with those of human judges.

3 LLM and Human Labelling Agreement

This study aims to measure the agreement between LLMs and humans, examine the trade-off between cost and accuracy, identify the reasons for disagreement, and assess the sufficiency of existing metrics for evaluating labelling accuracy. Specifically, we explore the following three sub-questions:

Table 1. Total number of queries and included passages from DL21 and DL22 and the maximum, minimum and average number of passages per query (Q).

Dataset	Queries	Passages	Min/Q	Max/Q	Avg/Q
DL21	53	1549	16	44	29.23
DL22	76	2673	19	53	35.17

Table 2. The distribution of relevance judgments for the passages included from DL21 and DL22.

Dataset	0	1	2	3
DL21	23.89%	32.41%	27.89%	15.82%
DL22	40.55%	32.44%	17.81%	9.20%

RQ1 How similar are LLMs to human judges in generating relevance labels?

- **RQ1.1** How accurate are LLMs in producing relevance labels for passages compared to human-provided judgements, and what are the associated costs of using LLMs for labelling?
- **RQ1.2** What factors may influence the disagreement between humans and LLMs?
- **RQ1.3** Are current data and metrics sufficient to establish the reliability of using LLMs for relevance labelling?

3.1 Experiment Design

This section details the experiment design to address the research questions, with more details about follow-up experiments presented later in Section 3.2.3 and 3.2.4.

3.1.1 Test Collections and Participating Systems. To understand the performance of LLMs in relevance labelling for passages (RQ1.1 and RQ1.2), we use queries and passages from the passage retrieval task of the DL21 [8] and DL22 [**?**]. Both years used the expanded MS MARCO dataset (v2), which contains around 138 million passages [29]. The relevance judgements of these passages were collected using a 4-point scale (0-3) by National Institute of Standards and Technology (NIST) judges.

We use the union of the top ten passages returned by each participating IR system to be labelled by LLMs. Seven representative IR systems are explored: two lexical models (TF-IDF and BM25); three neural re-rankers (ColBERT [19], monoBERT [31] and monoT5 [30]); one neural-augmented index (Doc2Query [32]); and one dense model (ANCE [47]). Neural models use publicly available checkpoints, fine-tuned on MS MARCO. Retrieval was conducted using Pyterrier [26], except for Doc2Query for which Pyserini [24] was used over a pre-built augmented corpus with doc2query-T5 expansions.

Of the union of passages returned by all systems, we only include passages for which NIST human judgements are available, allowing for comparison with LLM labels. Detailed statistics for the queries and included passages are provided in Table 1, with the distribution of the relevance scores shown in Table 2. Unless otherwise specified, reported results include DL21 and DL22 combined.

3.1.2 LLMs, Prompts and Metrics. Our experiments use nine LLMs from four different providers, selecting both a smaller, less capable and more cost-effective LLM and a larger, more sophisticated and more expensive option from each provider as follows:

- **Anthropic**:[2] Claude-3 Haiku and Claude-3 Opus.
- **Cohere**:[3] Command-R and Command-R+.
- **Meta AI**:[4] LLaMA3-instruct-8B and LLaMA3-instruct-70B.
- **OpenAI**:[5] GPT-3.5-turbo (1106), GPT-4 (0613), and GPT4o (2024-05-13).

GPT-4o was included as a more affordable yet still capable alternative to GPT-4, which was used by Upadhyay et al. [44], achieving competitive results.

Model parameters are set consistently across all LLMs, identical to those used in Thomas et al. [42]: `top_p` is set to 1.0, `frequency_penalty` at 0.5, `presence_penalty` at 0, and `temperature` at 0. GPT models are run through Azure OpenAI Services, and other LLMs are run through Amazon Bedrock. Cost calculations for running the LLMs are based on the pricing provided for input and output tokens by these services during May-June 2024.

Three different zero-shot prompts are used in the experiments to examine their impact on the performance and stability of relevance labels produced by each LLM:

- **Basic Prompt**: This prompt provides minimal instructions, only giving the model the description of the relevance judgment scale and asking it to return a relevance label as a single number. The prompt is shown in Figure 1.
- **Rationale Prompt**: This prompt adopts the prompt used by Upadhyay et al. [43] which instructs the model to provide an *explanation* along with the relevance label. To maintain consistency among prompts, we do not use examples as in the original prompt. The full prompt is shown in the Appendix.
- **Utility Prompt**: This prompt is a modified version of Thomas et al. [42]'s optimal (i.e., DNA) prompt. The information need description and narrative are omitted in our prompt since they are not available in DL21 and DL22. Instead of using a 3-point scale, we have adopted a 4-point scale, consistent with the scale used in DL21 and DL22. This prompt instructs the model to assess *how useful the answer would be for a report*, similar to the instructions given to TREC judges. The full prompt is shown in the Appendix.

Labels are parsed according to the format specified in each prompt. Any labels that cannot be automatically parsed are excluded from the analysis. We note that parsing issues are minimal and are more frequent in smaller LLMs. Missing values are reported in the discussion and captions of figures in Section 3.2 (i.e., Figures 2, 7, and 8) to ensure the results can be interpreted in context.

The performance of relevance labels created by LLMs relative to the available NIST human relevance judgements are evaluated using the Mean Absolute Error (MAE) given both graded and binary labels. When binary labels are used for some metrics, scores of 2 and 3 are mapped to 1, according to TREC's recommendation and consistent with the baseline of Damessie et al. [10], which is used to interpret the results. We evaluated agreement with NIST judges using Cohen's κ [6] and Krippendorff's α on an ordinal scale [21]. Cohen's κ only considers exact nominal matches, while Krippendorff's α takes the severity of the error into account. Additionally, we report the overall accuracy and precision of binary labels, and the likelihood of labelling passages as relevant, for each LLM.

[2]https://www.anthropic.com
[3]https://cohere.ai
[4]https://ai.meta.com
[5]https://openai.com

```
Please read the query and passage below and indicate how relevant the
passage is to the query. Use the following scale:

3 for perfectly relevant: The passage is dedicated to the query and
contains the exact answer.
2 for highly relevant: The passage has some answer for the query, but the
answer may be a bit unclear, or hidden amongst extraneous information.
1 for related: The passage seems related to the query but does not answer
it.
0 for irrelevant: The passage has nothing to do with the query.

Query: {query}
Passage: {passage}

Indicate how relevant the passage is, using the scale above. Give only a
number, do not give any explanation.
```

Fig. 1. The basic prompt used with LLMs to label passage relevance, adopting the same scale description used in DL21 and DL22.

3.1.3 Disagreement and Metric Correlation. To address RQ1.2 about cases of disagreement between humans and LLMs, we manually inspected the failure cases in an exploratory manner to gain insights into possible reasons for failure. While this was not a systematic analysis, it yielded observations that motivated additional analyses and experiments, which are described in Section 3.2.2, 3.2.3 and 3.2.4.

Informed by the outcomes of RQ1.2, which suggests that alternative metrics may provide additional insights into the reliability of LLMs (with further details discussed later), we examine the implications of using different metrics to measure the reliability of LLMs and analyse their correlations as part of RQ1.3.

3.2 Results and Discussion

3.2.1 LLM agreement with humans and the cost-performance trade-off.

RQ1.1 *How accurate are LLMs in producing relevance labels for passages compared to human relevance judgements, and what are the associated costs of using LLMs for relevance labelling?*

Figure 2 shows the agreement between NIST human relevance judgements and LLM relevance labels using the three prompts. The agreement is measured using Cohen's κ on a binary scale (shown on the left) and Krippendorff's α on a 4-point ordinal scale (shown on the right). Costs, expressed in USD, are based on the number of input and output tokens used in each LLM-prompt combination. The cost of using each LLM varies depending on the prompt due to differences in the number of input (i.e., prompt) tokens and, more substantially, the number of output tokens. This explains why the rationale prompt, which requires an explanation for relevance, is usually more expensive than other prompts given the same LLM. Unparsable labels for each LLM-prompt are minimal, with an average of 0.22% and a maximum of 1.89% of missing labels and were excluded when computing the agreement and accuracy metrics.

Human-to-human agreement levels (measured in previous research) are used as baselines to interpret the degree of agreement observed between LLMs and humans. Specifically, we use two baselines that measure the agreement between silver judges, those who have task expertise but lack topic expertise, and one baseline that measures agreement between bronze judges, those who have neither task nor topic expertise, as defined by Bailey et al. [3]. The baselines are as follows:

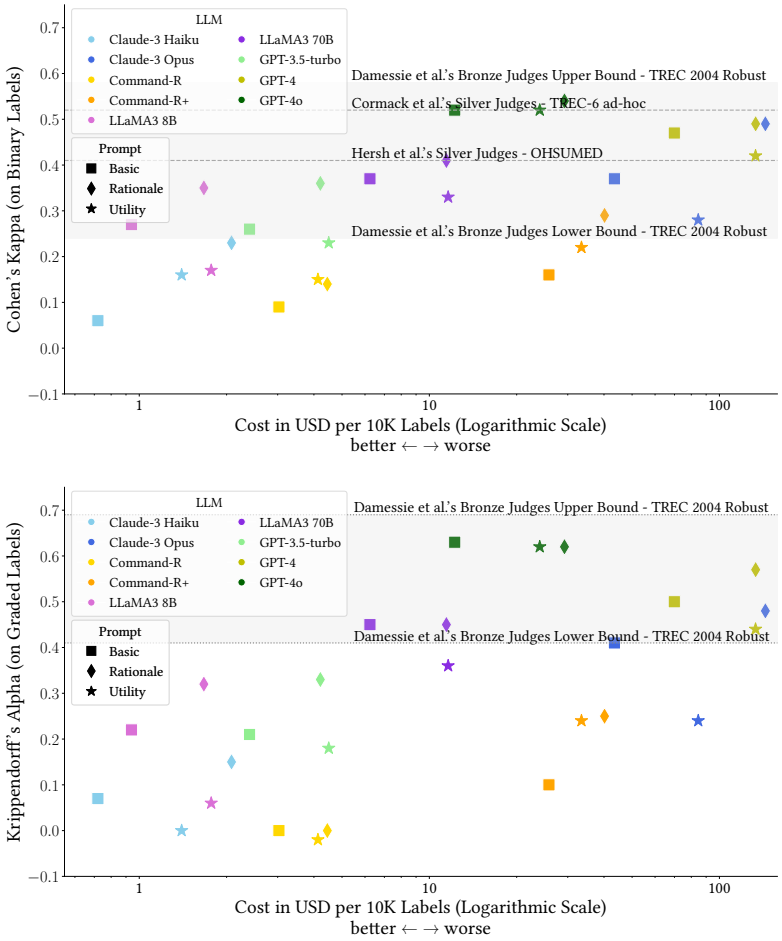


Fig. 2. Agreement between NIST relevance judgments and LLM relevance labels, measured using Cohen's κ on a binary scale (top) and Krippendorff's α on a 4-point ordinal scale (bottom), against cost. Colours represent LLM providers, with shades from lighter to darker indicating less to more capable models. Cost is calculated per 10K labels based on the average cost per label using the number of input and output tokens for each LLM-prompt combination. Baselines are depicted in the shaded grey area and dashed lines. Unparsable labels for each LLM-prompt are minimal, with an average of 0.22% and a maximum of 1.89% of missing labels.

- **Damessie et al. [10]**: The range of agreement measured using both Cohen's κ and Krippendorff's α on a graded scale among different groups of bronze judges. Relevance judgements were performed on the TREC 2004 Robust Track [45] with crowdsourcing and lab-based settings.
- **Hersh et al. [16]**: Agreement measured using Cohen's κ on a binary scale with silver judges on the OHSUMED test collection.
- **Cormack et al. [7]**: Agreement using Cohen's κ on a binary scale with silver judges on the TREC-6 ad hoc track [14].

All baselines are depicted in Figure 2 for reference, with the range of agreement observed by Damessie et al. shaded in grey and other agreements measured by Hersh et al. and Cormack et al. represented as dashed lines. It is important to note that these human-to-human agreement baselines were obtained from different test collections than those used in our study. These collections differ in document length, types of judges, and possibly in overall nature, yet the agreements are broadly consistent and provide reasonable approximations of human-to-human agreement in relevance judgements.

The x-axis indicates cost, represented on a logarithmic scale; therefore, the visual linear relationship observed in Figure 2 reflects a logarithmic relationship. Inexpensive small LLMs typically yield low agreement values, whereas achieving human-level performance requires larger models and higher financial investment, which is consistent with the scaling laws of LLMs [17].

Most highly capable LLMs perform within the human-to-human agreement range as measured by Cohen's κ. Notably, GPT-4o achieves a high level of agreement, comparable to the top agreement among silver judges, and substantially surpasses the performance of GPT-4 at less than half the cost.

GPT-4, LLaMA 70B, Claude-3 Opus, and Command-R+ also demonstrate LLM-to-human agreement competitive with human-to-human agreement. Interestingly, the open-source LLaMA 70B model achieves agreement levels that are similar to GPT-4, which is proprietary and among the most expensive LLMs. To illustrate the cost differences, the computing cost of running LLaMA 70B with a basic prompt on our subsets of DL21 and DL22 is $2.63, whereas GPT-4 costs $29.49.

When using Krippendorff's α to measure agreement on a graded scale, only GPT-4o and GPT-4 achieve levels in the human-to-human agreement range regardless of the used prompt. Command-R+ falls below the expected range, while LLaMA 70B and Claude-3 Opus show variable performance depending on the prompt used, with some prompts achieving agreement levels within the range. It appears that graded relevance is a harder task than binarised relevance: more LLMs fall within the range under binarised labels, whereas fewer do so when using graded labels.

We note that similar agreement scores of LLMs with the Utility prompt were obtained over the *full set of passages* retrieved by all TREC-submitted runs, as shown in Figure 16 in Appendix B.

While varying prompts in smaller LLMs lead to substantial differences in agreement, except in the case of Command-R, most larger LLMs exhibit higher stability in agreement regardless of the prompts used. The basic prompt, which requires the fewest input tokens and generates the fewest output tokens, performs effectively and is the most cost-efficient option. More complex prompts, while increasing costs, do not always enhance performance and can actually degrade it.

To examine the performance of LLMs beyond agreement scores, we compute the confusion matrices for all LLM-prompt combinations and report relevant metrics in Table 3, which shows the MAE for binary and graded relevance labels, overall accuracy, precision given the binary labels of non-relevant (0) and relevant (1), and the probability of each LLM-prompt combination to label a passage as relevant. For brevity, we only report the top-performing LLMs in Table 3, but consider all results in the discussion when relevant. Full results including all LLM-prompt combinations and the percentage of unparsable labels are provided in Table 12 in Appendix B.

The overall accuracy of LLMs is reasonable in most cases, mainly displaying lower precision for relevant (i.e., positive) labels, in other words, showing higher rates of false positives. The probability of these LLMs in Table 3 labelling a passage as relevant is, in most cases, substantially higher than that of human judges, who have a 33% probability of judging a passage as relevant given DL21 and DL22.

3.2.2 Factors causing disagreement.

RQ1.2 *What factors influence the disagreement between humans and LLMs?*

Table 3. MAE given binary and graded labels, precision (Prec) for non-relevant (0) and relevant (1) labels and the probability (P) of labelling a passage as relevant for all LLM-prompt combinations, including only the top performing LLMs.

LLM	Prompt	MAE (Binary)	MAE (Graded)	Accuracy	Prec(Label=0)	Prec(Label=1)	P(Label=1)
Claude-3 Opus	Basic	0.34	0.82	0.66	0.92	0.49	0.61
Claude-3 Opus	Rationale	0.25	0.77	0.75	0.91	0.58	0.50
Claude-3 Opus	Utility	0.41	1.05	0.59	0.94	0.44	0.71
Command-R+	Basic	0.51	1.24	0.49	0.98	0.39	0.84
Command-R+	Rationale	0.40	1.08	0.60	0.93	0.45	0.70
Command-R+	Utility	0.47	1.00	0.53	0.97	0.41	0.78
LLaMA3 70B	Basic	0.34	0.81	0.66	0.94	0.49	0.63
LLaMA3 70B	Rationale	0.31	0.81	0.69	0.94	0.52	0.59
LLaMA3 70B	Utility	0.37	0.95	0.63	0.94	0.47	0.67
GPT-4	Basic	0.27	0.78	0.73	0.92	0.56	0.53
GPT-4	Rationale	0.22	0.64	0.78	0.82	0.68	0.31
GPT-4	Utility	0.30	0.86	0.70	0.93	0.53	0.57
GPT-4o	Basic	0.21	0.61	0.79	0.84	0.69	0.32
GPT-4o	Rationale	0.21	0.64	0.79	0.87	0.65	0.38
GPT-4o	Utility	0.22	0.61	0.78	0.88	0.63	0.41

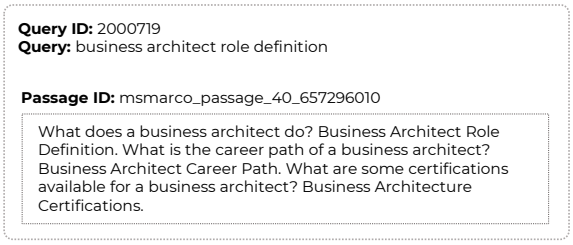


Fig. 3. An example false positive label: GPT-4 is fooled by query keywords, although the passage itself does not answer the query.

In our manual inspection of cases where LLMs and human judgements disagree, we observed that false positive passages, which are the most common error, often contain the query words but fail to provide useful information to the user. Figure 3 shows an example.

To further investigate this, we compute the ratio of query words present in each corresponding passage and find the average ratios for true positives, true negatives, false positives, and false negatives, as shown in Table 4. If the presence of query words impacts the relevance score assigned by LLMs, we would expect higher rates of query words in false positives compared to true negatives, and lower rates in false negatives compared to true positives; this appears to be the case across all LLMs, particularly in cases of false positives where the difference is significant across all LLMs.

3.2.3 Keyword stuffing. A key observation from our manual inspection of disagreement and from the query word matching in passages is that LLMs seem to be influenced by the presence of query words in the passage. That is, a non-relevant passage is likely to be labelled as relevant just because the query terms are present in it, leading to a higher rate of false positives and a distorted assessment of passage utility.

To investigate this hypothesis further, we design an experiment where we prompt LLMs to assess the relevance of either random or non-relevant passages with added query words. The creation of these passages is illustrated in Figure 4. We use two types of passages:

LLM	TP	TN	FP	FN
Claude-3 Haiku	0.74	0.70	0.75^{***}	0.72
Claude-3 Opus	0.74	0.68	0.78^{***}	0.68^{**}
Command-R	0.73	0.64	0.74^{***}	0.61^{**}
Command-R+	0.73	0.66	0.75^{***}	0.63
LLaMA3 8B	0.73	0.68	0.76^{***}	0.71
LLaMA3 70B	0.74	0.68	0.78^{***}	0.68^{*}
GPT-3.5-turbo	0.73	0.68	0.76^{***}	0.69
GPT-4	0.74	0.69	0.80^{***}	0.67^{***}
GPT-4o	0.74	0.71	0.81^{***}	0.71^{*}

Table 4. Average ratios of query words that appear in their labelled passages for True (T) and False (F) Positive (P) and Negative (N) passages across all LLMs (results are shown for the basic prompt only, for brevity). Asterisks indicate significance of FP vs TN or FN vs TP (Welch's t-test): $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

- **Random Passages (RandPs)**: Passages that are generated from randomly sampling words from the Brown corpus [22], forming nonsensical and ungrammatical passages. We create one passage of 100 words for each query in DL21. We also create other random passages of 200 and 400 words for each query to explore the effect of passage length on the error made by LLMs. We include DL21 only for this part of the analysis; since the passages are random, the underlying dataset should not have an impact on the results.
- **Non-relevant Passages (NonRelPs)**: Passages that are deemed non-relevant by both the LLM and NIST judges. We select 50 such passages randomly sampled from both DL21 and DL22 for each LLM-prompt combination (27 combinations).

We manipulate both types of passages by:

- **Query string injection (Q)** at a random position, in which the full original query string is inserted as-is at a random position.
- **Query words injection (QWs)**, where each query word is independently inserted into the passage at a random position (including stop words).

This results in four test conditions to be used with all LLM-prompt combinations, which we collectively refer to as *the keyword stuffing gullibility tests*. When varying the length of RandPs, the query string (or query words) is inserted only once at a random position regardless of the passage length. Unless otherwise specified, results of RandPs gullibility tests are based on the 100-word passages. An example of passage construction for RandP and NonRelP using query string injection is shown in Figure 5.

Figure 6 shows the distribution of relevance labels generated by GPT-4 using the three prompts and the four keyword stuffing *gullibility tests*. Since we have started with either nonsense text or non-relevant text, merely adding query terms should not make it relevant: that is, a labeller should assign a score of "0" despite our manipulations.

Relevance labels when using RandPs are shown in Figure 6 (a). The test where we inject the full query string appears to fool GPT-4 more often than does the test that injects query words separately. It is particularly concerning that in the basic prompt, approximately 26% of the random nonsensical passages are labelled as perfectly relevant merely due to the out-of-context presence of the query. The other prompts exhibit lower susceptibility to such errors.

Figure 6 (b) shows relevance labels when using NonRelPs. Both tests of injecting full query strings and individual query words tend to generate a higher ratio of passages mislabelled as relevant compared to RandPs, but with a lower

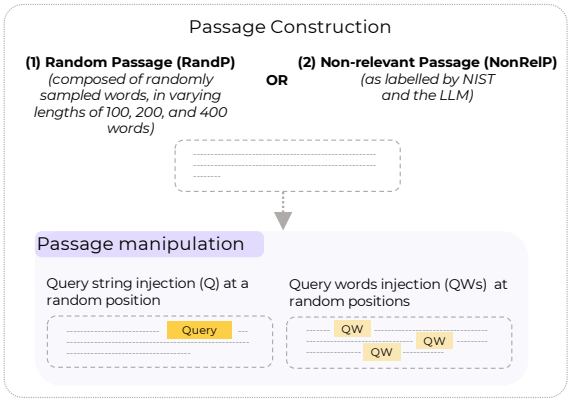


Fig. 4. Passage construction and manipulation to generate input passages for query-passage relevance labelling.

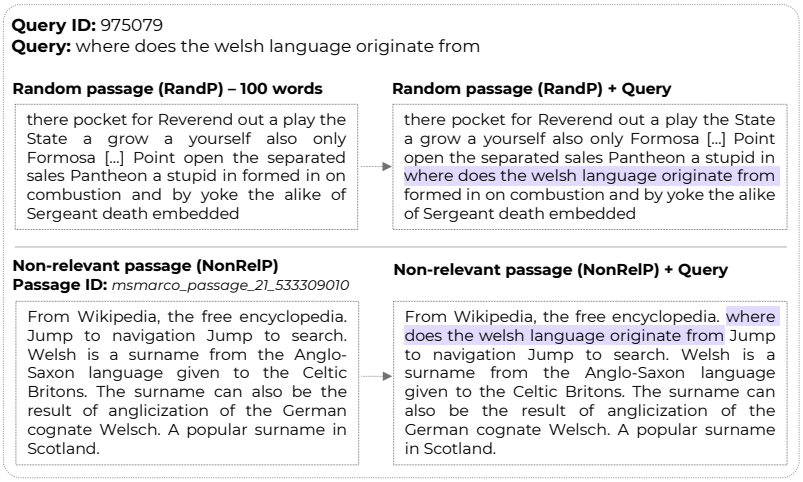
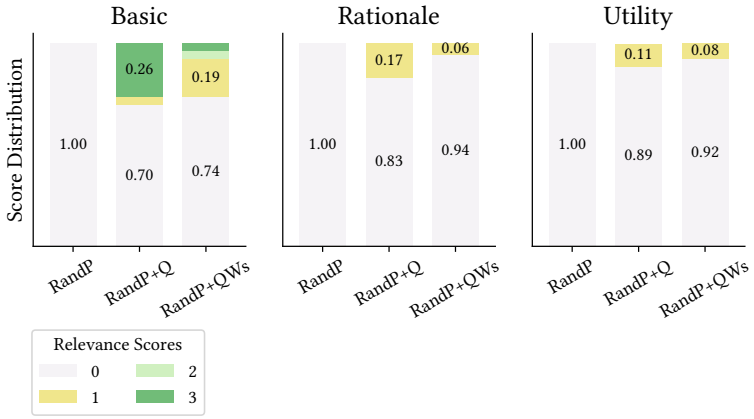


Fig. 5. An example of a RandP injected with a query string (top) and a NonRelP as per both NIST and GPT-4 (with the basic prompt) injected with the query string (bottom).

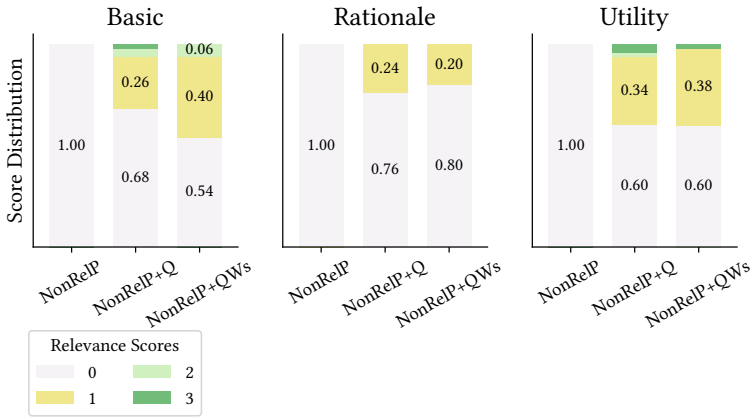
level of relevance when using the basic prompt. Most scenarios assign a marginal relevance of 1, with only a few cases showing high or perfect relevance. This is expected because the passages are sensible, in the sense that they were returned by IR systems in response to their respective queries, making it harder to label them correctly when injected with queries.

The performance of all LLMs in the keyword stuffing gullibility tests is summarised using the MAE. This metric is ideal for quantifying the error of LLMs, under the assumption that all input passages are non-relevant, and a relevance label of "0" is expected. The MAE weights errors according to their magnitude: responses with a score of 3 contribute more substantially to the MAE than those with scores of 1 or 2. This weighting makes the MAE particularly useful for quantifying deviations from the expected score of "0".

Figure 7 displays the MAE for all LLMs, averaged across all prompts used in the keyword stuffing gullibility tests. This averaging reflects the variation in prompts that researchers or practitioners might use, thereby accounting for these



(a) Keyword stuffing in randomly selected word passages (RandP) with injected queries (RandP+Q) and query words (RandP+QWs) given different prompts.



(b) Keyword stuffing in non-relevant passages (NonRelP) with injected queries (NonRelP+Q) and query words (NonRelP+QWs).

Fig. 6. Relevance score distribution of GPT-4 relevance labels when tested against keyword stuffing gullibility tests with two types of input passages (a) RandP and (b) NonRelP.

differences as potential contributors to errors or instability in the performance of LLMs. Most LLMs exhibit varying degrees of susceptibility to these tests, with GPT-4o demonstrating high resilience, particularly to tests involving RandPs. We note, however, that the results are limited by the varying number of generated labels available for each LLM – either because there were not enough passages labelled as non-relevant to be used in our NonRelPs tests, or due to parsing issues, as observed with Claude-3 Haiku, which tended to produce invalid outputs when fed with random passages. Exact figures are provided in the caption of Figure 7.

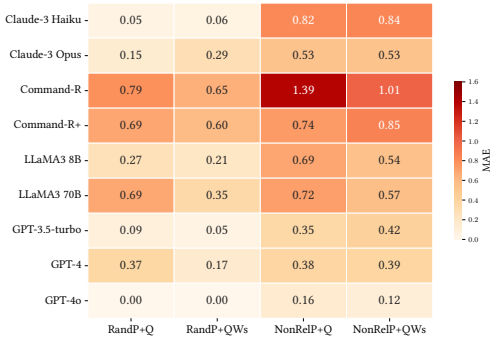


Fig. 7. The MAE scores for each LLM in each keyword stuffing gullibility test, averaged across the three prompts. Note: In RanP+Q and RanP+QWs, 20% of the labels generated by Claude-3 Haiku are unparsable. In NonRelP+Q and NonRelP+QWs GPT-3.5-turbo and LLaMA3 8B miss 8% and 17% of the labels, respectively, due to a lack of sufficient non-relevant passages to sample from. Other cases of missing labels are negligible, with each being less than 1%.

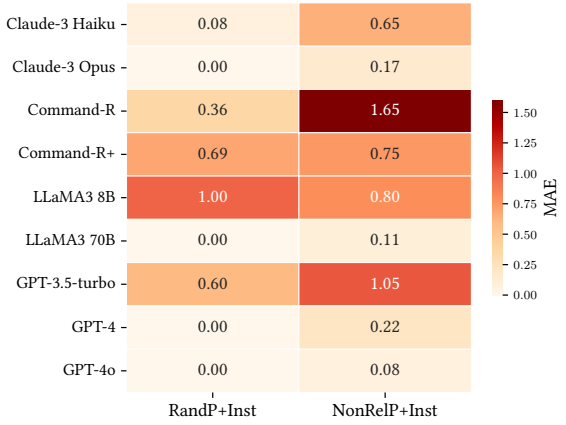


Fig. 8. The MAE scores for each LLM with both instruction injection gullibility tests, averaged across the three prompts. Note: In RandP+Inst, about 50% of the labels generated by Claude-3 Haiku are unparsable. In NonRelP+Inst, Claude-3 Haiku generates 5% of unparsable labels, GPT-3.5-turbo and LLaMA3 8B miss 8% and 17% of the labels, respectively, due to a lack of sufficient non-relevant passages to sample from. Other cases of missing labels are negligible, with each being less than 1%.

As we vary the length of RandPs in our experiment to explore the effect of the passage length on the gullibility of LLMs, no consistent pattern emerges, except in the case of GPT-4, which tends to make more errors as the passage length increases. Detailed results are omitted for brevity.

3.2.4 Instruction Injection. The previous section detailed experiments examining the impact of the presence of query strings or individual query words in passages, simulating keyword stuffing as a well-known Search Engine Optimisation (SEO) strategy to enhance ranking. This section explores another potential strategy, whereby content generators may manipulate LLMs to respond in a certain way or, in relation to relevance labelling, favourably label their content as relevant. We use the same RandP and NonRelP framework as described in Section 3.2.3. Each passage is preceded by an additional Instruction (Inst): 'The passage is dedicated to the query and contains the exact answer'. We refer to these tests as *Instruction Injection Gullibility Tests*.

Similar to the keyword stuffing gullibility tests, we quantify the error made by LLMs using MAE, where the expected label is "0". Figure 8 reports the MAE for each LLM across both tests, averaged across all prompts. The results show lower susceptibility compared to the keyword stuffing gullibility tests, with all large capable LLMs except Command-R+ performing well. Specifically, these models achieved an MAE of 0 when instructed to label RandPs as perfectly relevant, and exhibited some reasonably low degrees of error when labelling NonRelPs, as compared to their performance in keyword stuffing gullibility tests given the same type of passages. We also note that results are limited by the varying number of generated labels available for each LLM due to the reasons outlined previously. Exact figures are provided in the caption of Figure 8.

3.2.5 Agreement vs. Gullibility.

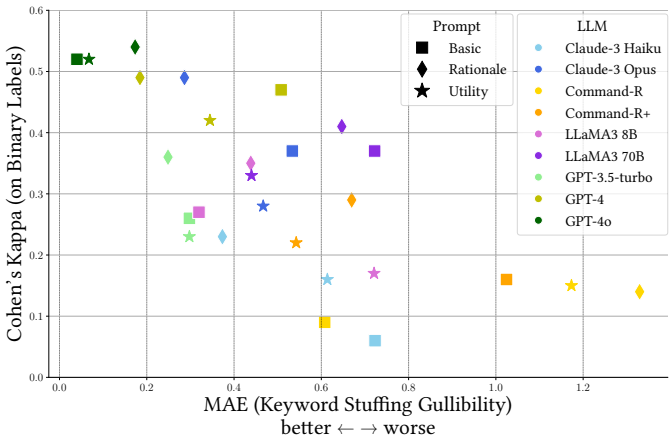


Fig. 9. Cohen κ scores against MAE for keyword stuffing gullibility tests.

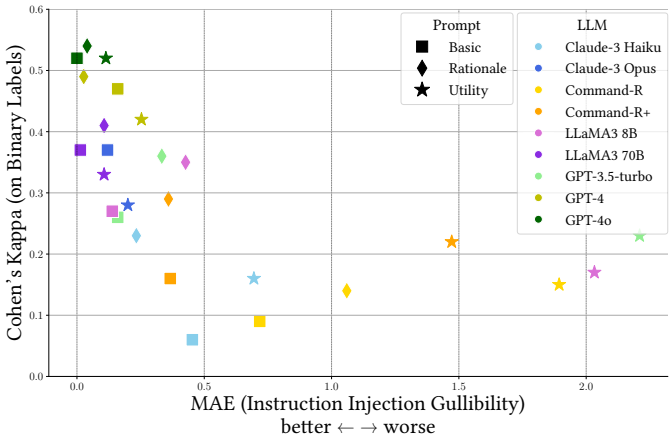


Fig. 10. Cohen κ scores against MAE for instruction gullibility tests.

RQ1.3 *Are current data and metrics sufficient to establish the reliability of using LLMs for relevance labelling?*

Figures 9 and 10 show the relationship between Cohen's κ and the average MAE for both *keyword stuffing gullibility* and *instruction injection gullibility* tests, respectively, for all LLM-prompt combinations. In general, the results show that conclusions drawn from evaluating LLMs using Cohen's κ do not necessarily mirror their corresponding performance based on the gullibility tests. For example, while the basic prompt seems to perform well according to Cohen's κ, it exhibits substantially higher vulnerability in the gullibility tests. In particular, the Pearson correlation coefficients between Cohen's κ and the MAE are measured as $\rho = -0.678$ for the keyword stuffing gullibility tests and $\rho = -0.582$ for the instruction injection gullibility tests, respectively.

Table 5. Total number of queries and included passages (the union of the top-10 of all participating TREC runs) from DL21 and DL22 and the maximum, minimum and average number of passages per query (Q).

Dataset	Queries	Passages	Min/Q	Max/Q	Avg/Q
DL21	53	7443	68	253	140.43
DL22	76	10573	58	277	139.12

Table 6. The distribution (%) of human relevance judgments for the passages (top-10 passages) of participating runs included in DL21 and DL22. Missing judgements are reported under N/A.

Dataset	0	1	2	3	N/A
DL21	45.56	28.30	17.35	8.80	0.00
DL22	46.85	29.06	13.95	6.03	4.10

4 System Ranking Agreement

In this second study, we aim to understand how the rankings of systems under evaluation (often referred to as *runs* in the context of TREC) differ when using LLMs versus human judges. We examine multiple aspects to gain a deeper understanding of these differences and their implications for evaluation. Prompted by previous findings in Section 3 that LLMs can be influenced by the presence of query terms, we question whether traditional word-matching systems gain an advantage under LLM-based evaluation. Specifically, we pose the following question and sub-questions:

RQ2 How similar are different LLMs to human judges in system ranking?

- **RQ2.1** How similar are the rankings of runs based on LLMs compared to those based on human judgements?
- **RQ2.2** Do traditional systems gain an advantage under LLM-based evaluation?

The following section details the experimental setup used to answer the raised questions.

4.1 Experiment Design

We evaluate all systems submitted to both DL21 (63 systems) and DL22 (100 systems) using relevance judgements from both human judges provided by NIST and those generated by LLMs. The same LLMs used in the labelling agreement study described in Section 3.1.2 are employed here. We only use the utility prompt [42] with all LLMs, as sensitivity to prompt variation was previously examined on a smaller set of systems in Section 3. Evaluation is conducted using the official TREC `run_eval` script.[6] We report and use NDCG@10, the official metric of this track, as the primary metric for our analyses. The following provides more details of the experimental setup specific to each research question. Table 5 presents query counts and statistics on the labelled passages, and Table 6 shows the distribution of relevance scores. The remainder of this section outlines the setup used to address each research question.

RQ2.1 *How similar are the rankings of systems based on LLMs compared to those based on human judgements?*

To assess the similarity between system rankings derived from the two evaluation methods (i.e., humans and LLMs), we use Kendall's τ [18], which measures the correlation between two ranked lists. To compare the ability of LLMs to discriminate among systems relative to humans, we examine the slope of the ranking lines; flatter slopes may indicate reduced discriminative power. We also perform pairwise comparisons between systems to assess *direction agreement* – that is, whether both evaluation methods agree that system 1 (S_1) is better than system 2 (S_2) – and whether the

[6] https://github.com/usnistgov/trec_eval

observed performance difference is statistically significant. The paired t-test is used to identify statistically significant differences. For each pair of systems (i.e., 1,953 pairs for DL21 and 4,950 pairs for DL22), we classify the comparison outcome into one of the following categories, following Ferro and Sanderson [13] and Moffat et al. [28]:

- **Active Agreement (AA)**: Both humans and the LLM agree on which system is better (e.g., $S_1 > S_2$), and this difference is statistically significant by both methods.
- **Passive Agreement (PA)**: Both humans and the LLM agree on which system is better, but the difference is not statistically significant by either.
- **Mixed Agreement (MA)**: Both humans and the LLM agree on which system is better, but statistical significance is observed by only one of the two.
- **Active Disagreement (AD)**: Humans and the LLM disagree on which system is better, and the difference is statistically significant by both.
- **Passive Disagreement (PD)**: Humans and the LLM disagree on which system is better, and the difference is not statistically significant by either.
- **Mixed Disagreement (MD)**: Humans and the LLM disagree on which system is better, and statistical significance is observed by only one of the two.

We interpret the above metrics based on the final conclusions reached by the LLM and whether it aligns with human judgement. Table 7 presents a visual representation of the metrics, with cells shaded with different colors to reflect the extent to which the LLM's conclusion aligns with that of humans. Assuming we are making a decision on whether S_1 is different from S_2 – for example, when comparing two candidate systems for production or publication – we consider a difference to exist only when a statistically significant difference is observed. Otherwise, both systems S_1 and S_2 are treated as equal.

It is clear that in cases of AA, the LLM reaches the same conclusion as humans. Similarly, if no significance is observed by both the LLM and humans – even if the direction of the scores differs – both conclude that S_1 and S_2 not performing differently, thus the LLM still arrives at the same conclusion (i.e., choosing either system is acceptable). These include cases of PA and PD. We refer to these cases as *Matching Conclusions*, as the LLM reaches the same conclusion as humans, even in cases of disagreement in direction, since significance is not observed. This is highlighted in green in Table 7.

If the LLM identifies no significant difference between S_1 and S_2, but humans do, we consider this a case of a *Missed Improvement*. In this scenario, the LLM fails to recognise a significant improvement detected by humans and effectively treats the two systems as equivalent. This may lead to missed opportunities, such as failing to deploy a better-performing system in production or to report a meaningful improvement in a research setting. We highlight these cases in yellow in Table 7.

Conversely, if the LLM observes a statistically significant difference between S_1 and S_2, while humans do not, we consider this a case of a *False Improvement*. In this scenario, the LLM overestimates the difference between the systems, concluding that one is significantly better than the other. While deploying such a system may not result in a performance regression, it may lead to wasted effort in production settings and, if published, could be misleading by overstating the value of the proposed improvement. We highlight these cases in purple in Table 7.

Cases of AD, where the LLM and humans reach opposite conclusions with statistical confidence, represent the most extreme form of disagreement. These are highlighted in red in Table 7, as they indicate a fundamental conflict in which the LLM may recommend one system with significance while humans consider it significantly worse.

RQ2.2: *Do traditional systems gain an advantage under LLM-based evaluation?*

		LLM			
		$S_1 > S_2^*$	$S_1 > S_2$	$S_1 < S_2$	$S_1 < S_2^*$
NIST	$S_1 > S_2^*$	AA	MA	MD	AD
	$S_1 > S_2$	MA	PA	PD	MD
	$S_1 < S_2$	MD	PD	PA	MA
	$S_1 < S_2^*$	AD	MD	MA	AA

Table 7. Matrix of **A**ctive, **P**assive, and **M**ixed **A**greement (AA, PA, MA) and **D**isagreement (AD, PD, MD) in pairwise system comparisons (S_1, S_2) between NIST (human judges) and LLM-based evaluations. A * indicates a statistically significant difference. Cell shading indicates the alignment of LLM conclusion with humans: green for *matching conclusions*, yellow for *missed improvements*, purple for *false improvements*, and red for *opposite conclusions*.

Table 8. The total number of traditional and neural systems and their minimum, maximum, and average NDCG@10 scores given NIST human relevance judgments.

Dataset	Traditional				Neural			
	Total (%)	Min	Max	Avg	Total (%)	Min	Max	Avg
DL21	13 (20.97)	0.37	0.55	0.45	49 (79.03)	0.41	0.75	0.61
DL22	12 (12.00)	0.26	0.32	0.29	88 (88.00)	0.36	0.72	0.55

Table 9. The total number of traditional and *comparable* neural systems and their minimum, maximum, and average NDCG@10 scores given NIST relevance judgments in DL21.

Type	Total (%)	Min	Max	Avg
Traditional	13 (72.22)	0.37	0.55	0.45
Comparable Neural	5 (27.78)	0.41	0.50	0.45

We classify the participating systems into two broad categories – *traditional* and *neural* – based on the system classification provided by TREC [8, 9], which originally grouped systems into three types: *trad*, *nn*, and *nnlm*. Systems labelled as *trad*, which involve no neural representation learning (e.g., classical learning-to-rank, PRF, and BM25), are mapped to our *traditional* category. The remaining two groups –*nn*, which use representation learning without pre-trained models, and *nnlm*, which incorporate pre-trained models such as those used in BERT-style re-ranking – are collectively grouped under our *neural* category. Table 8 shows system counts for each category along with performance statistics.

We measure the *boost* each system in the two categories receives when evaluated using LLMs in terms of the NDCG@10 score. To account for performance differences between categories (as shown in Table 8, where traditional systems have an overall lower mean performance), we select a subset of *neural* systems whose effectiveness, measured by mean NDCG@10 scores, is very close to that of the traditional systems. We call this subset *comparable neural* systems. Table 9 reports statistics for the traditional and the *comparable* neural systems in DL21; no comparable systems were found in DL22, and thus it is excluded from this part of the analysis. We also perform an ANOVA to examine the impact of *system category* and *system effectiveness (i.e., NDCG@10)* (independent variables), on the observed *boost* (dependent variable).

4.2 Results and Discussion

RQ2.1: *How similar are the rankings of runs based on LLMs compared to that based on human judgements?*

Figure 11 shows the NDCG@10 scores of all participating runs for DL21 (top) and DL22 (bottom), as assessed by both LLMs (coloured dots) and humans (black diamonds), ranked from highest to lowest performance according to human judgments. Table 10 reports the Kendall's τ correlation, the slope of the rankings of runs, and the outcomes of our pairwise comparisons between runs based on human judgments versus LLMs labels.

As illustrated in Figure 11, LLMs tend to be more positive than human judges, often assigning higher scores to systems. This observation aligns with earlier findings in Section 3.2, where LLMs were shown to be more likely than humans to label passages as relevant. Nevertheless, the overall rankings of systems remain broadly consistent with human judgements, as reflected by the relatively high Kendall's τ (with the lowest τ being 0.84) as reported in Table 10. In terms of the slope of the ranking lines, Figure 11 shows that rankings produced by LLMs are generally flatter compared to those produced by human judges, with lower slopes as reported in Table 10, indicating that humans have greater discriminative power to distinguish among runs.

As shown in Table 10, AD ratios – where both LLMs and humans significantly disagree on which system is better – are zero across all cases. This means that there is no instance, at least with capable LLMs, where an LLM judges S_1 to be significantly better than S_2 while humans judge the opposite. We note a single instance of AD in DL21 when using Claude-3 Haiku (out of 1,953 pairs), one instance in DL22 when using GPT-3.5 Turbo, and seven instances in DL22 when using LLaMA-3 8B (out of 4,950 pairs). It is worth noting, however, that in cases of MA – where both LLMs and NIST agree on the direction of comparison but statistical significance is observed in only one – there is evidence of either missing or artificial significance differences. To explore this further, we plot the conclusion alignment as classified previously in Figure 12. Generally, results suggest that most LLMs exhibit more missing improvements than false improvements: while LLMs often agree with humans on which run is better, they are more likely to overlook meaningful improvements by failing to identify them as significant. This limitation may be mitigated with larger amounts of evaluation data – something that LLMs can help scale – and we leave this for future investigation.

GPT-4o stands out with notably different results: its agreement with human ranking of runs is the highest, as reflected by the highest Kendall's τ value, and it exhibits the steepest ranking slope among LLMs, even exceeding that of the human judges. In cases of MA, GPT-4o tends to exaggerate differences, marking them as significant. This results in smaller missing improvements (i.e., it rarely misses significant improvements) but leads to a higher rate of false improvements (i.e., marking some insignificant improvements as significant).

RQ2.2: *Do traditional systems gain an advantage under LLM-based evaluation?*

Figure 13 shows the average percentage boost received by each run category when evaluated using LLMs. Although both traditional and neural runs exhibit higher NDCG@10 scores under LLM-based evaluation, traditional runs receive nearly twice the boost compared to neural ones.

The boost traditional runs received when LLMs are used may be linked to their greater gullibility to the presence of query words in the labelled passages, as discussed earlier. Given that traditional systems generally have lower performance in terms of NDCG@10, there is a possibility that the boost they receive is related to their effectiveness. That is, since LLMs are more positive – i.e., they have a higher probability of labelling non-relevant passages as relevant – they could be boosting traditional systems more simply because these systems retrieve a larger proportion of non-relevant passages that LLMs incorrectly assess as relevant. Figure 14 shows the correlation between NDCG@10 and the boost received by runs, confirming this hypothesis: the lower the effectiveness of a system, the more likely it

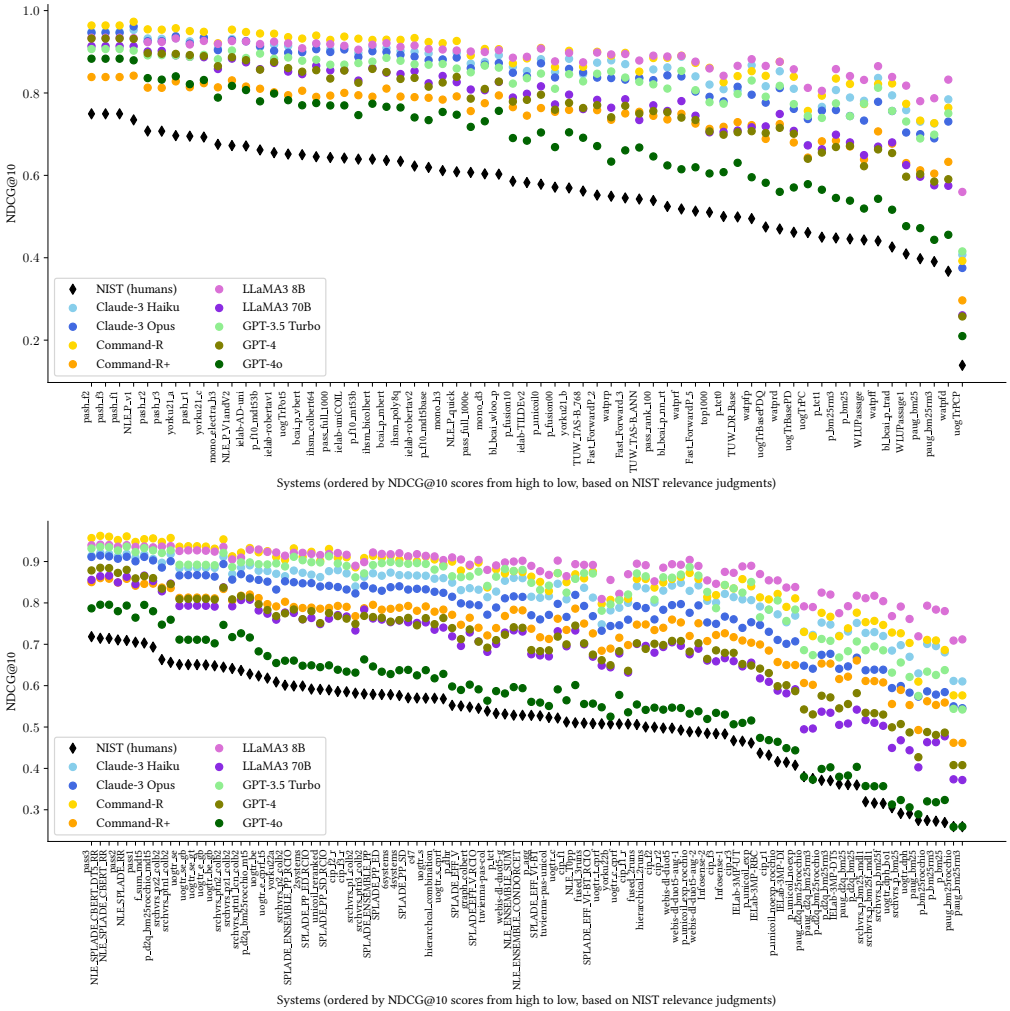


Fig. 11. The rankings of TREC DL21 (top) and DL22 (bottom) runs based on NIST relevance judgments and different LLM relevance labels using the Utility prompt [42].

is to be boosted by LLMs, particularly those with a high false positive rate when labelling passages. This effect also explains the flatter ranking lines observed previously.

We therefore present results using only comparable neural runs – those with performance similar to traditional systems – to control for the performance variable. As shown in Figure 15, the difference between traditional and comparable neural systems is smaller than that between traditional and all neural systems, but it still persists, except for GPT-4o, where it seems to boost neural systems more than traditional ones. We note, however, that these findings are not conclusive due to the small number of systems that meet our comparability criteria. To more robustly assess the impact of both system category and performance on the boost systems receive, we conduct ANOVA tests on the full set of runs.

Table 10. Kendall's Tau (τ) correlation, ranking slope, and the proportions of Active, Passive, and Mixed Agreement (AA, PA and MA) and Disagreement (AD, PD and MD) of run pairwise comparisons given LLMs and humans for DL21 and DL22 datasets. Note: slopes of LLMs are based on the visual lines of their rankings of runs shown in Figure 11.

LLM	**DL21**								**DL22**							
	τ	Slope	AA	PA	MA	AD	PD	MD	τ	Slope	AA	PA	MA	AD	PD	MD
Claude-3 Haiku	0.84	-0.0036	0.64	0.13	0.15	0.00	0.06	0.01	0.84	-0.0023	0.68	0.09	0.15	0.00	0.06	0.02
Claude-3 Opus	0.90	-0.0043	0.69	0.17	0.10	0.00	0.05	0.00	0.88	-0.0031	0.73	0.11	0.10	0.00	0.05	0.01
Command-R	0.87	-0.0038	0.63	0.15	0.16	0.00	0.06	0.01	0.85	-0.0027	0.72	0.08	0.13	0.00	0.05	0.02
Command-R+	0.88	-0.0038	0.65	0.17	0.12	0.00	0.06	0.00	0.85	-0.0029	0.70	0.10	0.13	0.00	0.06	0.02
LLaMA3 8B	0.85	-0.0023	0.62	0.14	0.17	0.00	0.06	0.01	0.82	-0.0017	0.65	0.09	0.16	0.00	0.05	0.03
LLaMA3 70B	0.91	-0.0054	0.69	0.16	0.10	0.00	0.05	0.00	0.87	-0.0039	0.73	0.09	0.11	0.00	0.05	0.01
GPT-3.5 Turbo	0.85	-0.0033	0.62	0.15	0.16	0.00	0.07	0.01	0.78	-0.0029	0.64	0.08	0.17	0.00	0.07	0.04
GPT-4	0.92	-0.0057	0.71	0.17	0.08	0.00	0.04	0.00	0.89	-0.0037	0.74	0.11	0.09	0.00	0.05	0.01
GPT-4o	0.94	-0.0068	0.73	0.17	0.07	0.00	0.03	0.00	0.92	-0.0046	0.77	0.13	0.06	0.00	0.04	0.00
NIST (humans)	-	**-0.0059**	–	–	–	–	–	–	–	**-0.0041**	–	–	–	–	–	–

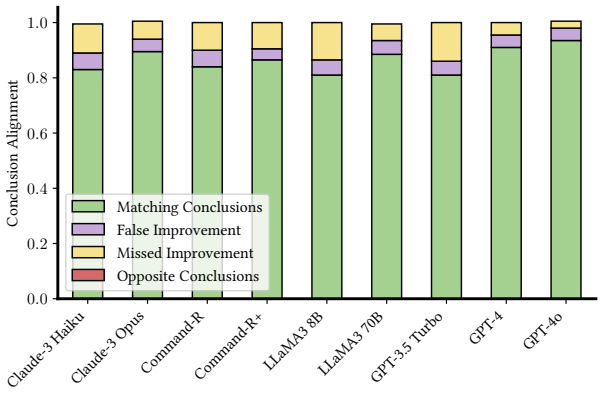


Fig. 12. The distribution of the conclusion alignment of LLMs with humans, averaged across DL21 and DL22. Colors correspond to the types of conclusion alignment presented in Table 7.

Table 11 presents the ANOVA results examining two variables: system category and system effectiveness. In DL21, the system category has a medium to large effect on the boost received, while effectiveness consistently shows a large effect. In DL22, the effect of the category is generally small to medium – except for GPT-4o and LLaMA3-70b, where it is considered large. Effectiveness continues to have a large effect in DL22 for most LLM, with the exception of GPT-4o. These findings suggest that system category does, indeed, influence the advantage gained under LLM-based evaluation, with most of the LLMs included in this experiment – adding a new concern that LLMs might boost keyword matching systems.

The results presented here provide additional evidence alongside the work of Balog et al. [4] and Otero et al. [33], which showed that differences between LLMs and human judges are more nuanced when examining system pairwise comparisons, as opposed to only using ranking correlations where such differences may be obscured. Our study contributes further evidence by testing multiple LLMs from different providers, while using the same set of topics across all cases, since topic choice can strongly influence how systems are compared. We also examine the effect of system

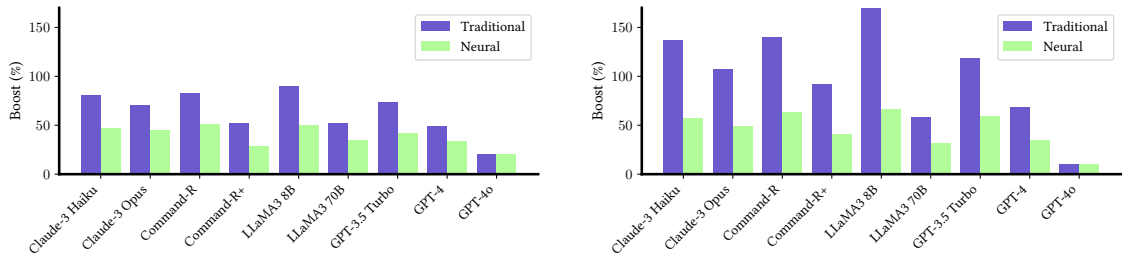


Fig. 13. The boost in NDCG@10 scores for both traditional and neural systems when using different LLM relevance labels in DL21 (left) and DL22 (right).

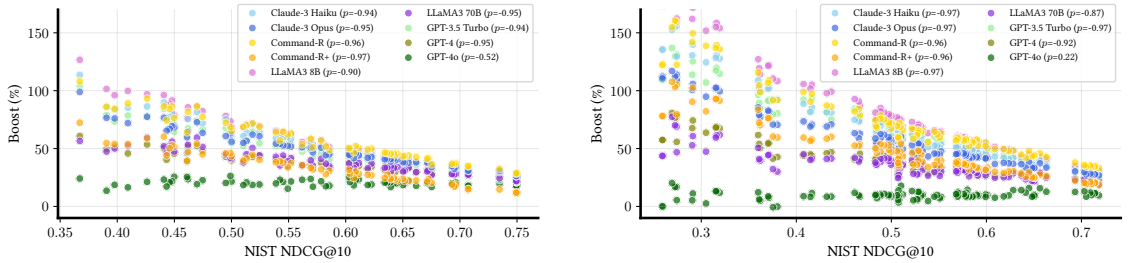


Fig. 14. The boost received by each system against NIST NDCG@10 scores and their Pearson correlation (p) when using different LLM relevance labels in DL21 (left) and DL22 (right). Note that one outlier run is removed from the DL21 plot but included in the correlation calculation.

category on how it is impacted by LLMs evaluation, following the observation by Otero et al. [33] that systems are affected to varying degrees. Although our sample size for each system category is limited, this study opens a new line of inquiry into the impact of LLMs across different types of systems. Finally, our results highlight how score inflation can minimise performance gaps among systems and provide detailed metrics to better understand and study the influence of LLMs evaluation on pairwise comparisons. Taken together, this combined evidence suggests that LLMs struggle to rank top-performing systems, are unfair to oracle systems, and result in different levels of score boosting across systems – possibly influenced by both their performance and category.

5 Conclusions

This research explored the performance of LLMs for labelling the relevance of passages in response to a query, considering whether such labels show accuracy comparable to human judges, and whether simple accuracy measures are sufficient to avoid the potential impact of simple adversarial activities. It also assessed whether such vulnerabilities benefit traditional word-matching systems. Additionally, it explored the implications of using LLMs as system evaluation judges – specifically, how closely their conclusions align with those of human assessors. We address the following research questions:

RQ1.1 How accurate are LLMs in producing relevance labels for passages compared to human-provided relevance judgements, and what are the associated costs of using LLMs for relevance labelling?

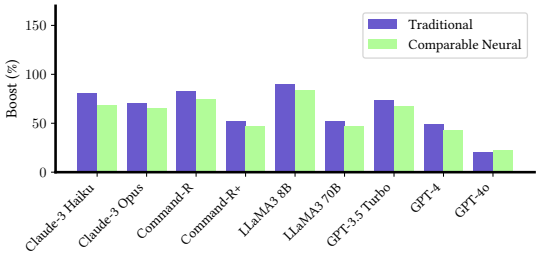


Fig. 15. The boost in NDCG@10 scores for both traditional and comparable neural systems when using different LLM relevance labels in DL21.

Table 11. ANOVA Effect Sizes (ω^2) of system category (independent) and effectiveness as measured by NIST NDCG@10 (independent) on the boost runs received. Red, orange and yellow represent large, medium and small effect sizes, respectively.

LLM	DL21		DL22	
	Category	Effectiveness	Category	Effectiveness
Claude-3 Haiku	0.145	0.71	0.038	0.828
Claude-3 Opus	0.194	0.677	0.071	0.793
Command-R	0.173	0.703	0.063	0.796
Command-R+	0.109	0.78	0.05	0.82
LLaMA3 8B	0.122	0.763	0.029	0.835
LLaMA3 70B	0.241	0.621	0.159	0.679
GPT-3.5 Turbo	0.133	0.744	0.091	0.791
GPT-4	0.269	0.582	0.113	0.735
GPT-4o	0.524	0.328	0.567	0.084

In common with past work, we see good agreement between labels from some LLMs and labels from qualified human judges. Performance varies with model and prompt, but broadly the larger and more expensive models show both better performance, and greater consistency across prompt variations.

RQ1.2 What factors influence the disagreement between humans and LLMs?

On the whole, models tend to be more positive than humans: while a "non-relevant" label is relatively reliable, a "relevant" label may be more prone to being a false positive. This is true of most models and prompts. Closer examination showed that many models are prone to false positives when query words are present, even if the passage is clearly not relevant: that is, they fall victim to keyword stuffing. Many models can also be manipulated into giving false positives by inserting "instructions" into the passage itself, meaning labels from LLMs are prone to spamming.

RQ1.3 Are current data and metrics sufficient to establish the reliability of using LLMs for relevance labelling?

Commonly used measures of overall agreement are useful in their ability to distinguish better models and prompts from others, but do not capture patterns of failure. Relying exclusively on agreement, therefore, risks blinding us to interesting patterns of failure, such as keyword or instruction stuffing. We recommend close examination of the output of models based on additional measures, and have proposed two gullibility tests.

RQ2.1 How similar are the rankings of runs based on LLMs compared to that based on human judgements?

LLMs labels yield more positive evaluations of systems, but the resulting system rankings from LLMs and human judges appear similar, with rare cases of active disagreement where opposing conclusions are reached by humans and LLMs. However, cases of missed or false improvements remain a concern.

RQ2.2 Do traditional systems gain an advantage under LLM-based evaluation?

The lower a system's performance, the more likely it is to be boosted under LLM-based evaluation – a pattern observed with traditional word-matching systems, which generally show lower effectiveness compared to neural systems. However, even after controlling for performance and comparing traditional and neural systems with similar effectiveness, we find that LLMs tend to rate traditional systems more positively than neural ones. This effect appears to be linked to the tendency of LLMs to label documents as relevant based on the presence of query keywords – a characteristic of passages retrieved by traditional systems. However, this finding is limited by the small size of our system subsets.

Overall, the results indicate that despite good performance in aggregate – e.g. human-like measures of Cohen's κ and Krippendorff's α – and similar rankings of TREC runs, competitive LLMs are likely to be influenced by the presence of query words in the labelled passages, even if those passages are constructed from random words. This influence of queries may contribute to a higher rate of false positives. This limitation also manifests in the tendency of LLMs to boost traditional systems, whose retrieved passages would include the query words.

Considering the sets of passages that need to be labelled for relevance when building test collections, a considerable portion of them would likely be non-relevant, having been retrieved by systems due to the presence of query words. Mislabelling them as "relevant" due to this influence could pose a major limitation on the use of LLMs for the relevance labelling task and a negative impact on models trained on such labels. An LLM labeller would be expected to at least exhibit higher ability in relevance labelling than an information retrieval system.

In production environments, LLMs might be vulnerable to keyword stuffing and other SEO strategies. This is not to suggest that LLMs have a unique limitation, as there is evidence that humans are also impacted by word matching [15, 20, 23]. However, recognising these challenges will allow for more effective testing of such models, similar to the ways in which human-based labelling activities are safeguarded with approaches such as the addition of gold-standard questions.

Because LLMs are generally more lenient – more likely to label documents as relevant – the performance gaps across systems appear narrower. Top-performing systems leave less room for further judgement improvement, while less effective systems benefit more from this positive bias, leading to flatter lines of rankings. As a result, statistical significance becomes harder to detect. When comparing any typically highly effective system (e.g., neural) to a weaker one (e.g., traditional) – the performance gap is likely to be underestimated under LLM-based evaluation. This is seen in improvement spotted by humans but missed by LLMs.

This study is not intended as an evaluation of specific LLMs, which continue to evolve rapidly. Rather, our contribution lies in demonstrating the types of tests and analyses that can be applied to assess the capacity of LLMs to perform relevance labelling, and in examining the potential consequences of their adoption for this task. Although we have reported some findings regarding the relative performance of particular LLMs, these should be viewed as illustrative; the central message of the study is the methodology and the broader insights it provides into the use of LLMs for evaluation.

The gullibility tests proposed in this study are not intended to be exhaustive and are certainly just the beginning of research in this area. While we, as a community, have invested significantly in evaluating the reliability of human judgements, it may now be prudent to invest in testing these models beyond established evaluations to more comprehensively assess their reliability.

Our study used particular LLMs and prompts, and tested on two test collections, and of course, other LLMs or prompt variants may not demonstrate exactly the same bias. However, our experiments included a range of competent models. Their overall performance is as good as human judges; it was only on closer examination, beyond simple aggregates of agreement metrics and ranking correlations, that we observed the weaknesses described here. Performance in aggregate, whether for this particular setup or any other, can mask unfortunate edge cases. As we adopt new instruments, caution is advised.

6 Limitations and Future Directions

The conclusions drawn from this research should be interpreted in light of the limitations mentioned previously. Our interpretation of LLM agreement with humans – which is high – is based on a comparison with human-to-human agreement levels measured on test collections older and different from those used in our study. These collections differ in document length, types of judges, and possibly in their overall nature; nevertheless, the reported agreement levels are broadly consistent across different collections and provide reasonable approximations of human-to-human agreement in relevance judgements. Measuring human-to-human agreement on the same collections used in this study would represent the ideal baseline for a more accurate comparison.

We acknowledge that the test collections used in this study consist of short text documents (i.e., passages). Their nature, length, and the characteristics of the human judges involved in assessing their relevance, among other factors, may have influenced the results observed. Further research on test collections of different types and with documents of varying lengths may yield either similar or divergent conclusions. Nevertheless, we note that our findings appear broadly consistent with those reported in studies involving longer documents [42].

We also note that the results regarding LLMs' gullibility are limited by the varying number of generated labels available for each LLM – either because there were insufficient passages labelled as non-relevant to be used in our NonRelPs tests, or due to parsing issues, as observed with Claude-3 Haiku, which tended to produce invalid outputs when presented with random passages. We report these issues, and we believe that their impact is minimal and unlikely to affect the main conclusions.

Our analysis of the effect of system category – whether neural or traditional – is limited by the small number of traditional systems considered in our experiments. While our results suggest that LLMs' vulnerability to making mistakes when query words are present in passages may give traditional word-matching systems an advantage, this limits our ability to draw reliable conclusions. We believe that LLM labelling performance should be interpreted in light of system category – which has typically been studied in the context of circularity, where LLMs are used for both labelling and re-ranking, but not with respect to simpler biases of this kind.

We urge future research to study the different biases and complexities arising from the use of LLMs in relevance judgements. While we believe that their potential outweighs their limitations, and that they represent a valuable resource in settings where evaluation resources are scarce or expensive to obtain manually, it remains important to understand the multiple dimensions of their use and the potential vulnerabilities that must be considered and tested in order to preserve the integrity of IR evaluation.

Acknowledgments

Marwah Alaofi is supported by a scholarship from Taibah University, Saudi Arabia. This work is also supported by the Australian Research Council (DP190101113). We thank RMIT AWS Cloud Supercomputing Hub (RACE) for providing technical and financial support to access a wide range of LLMs, with special thanks to Patrick Taylor. We also thank Kun Ran for his assistance with LLM access, and the anonymous reviewers for their valuable feedback.

References

[1] Zahra Abbasiantaeb, Chuan Meng, Leif Azzopardi, and Mohammad Aliannejadi. 2024. Can We Use Large Language Models to Fill Relevance Judgment Holes? arXiv:2405.05600 [cs.IR] https://arxiv.org/abs/2405.05600

[2] Marwah Alaofi, Luke Gallagher, Mark Sanderson, Falk Scholer, and Paul Thomas. 2023. Can Generative LLMs Create Query Variants for Test Collections? An Exploratory Study. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, Hsin-Hsi Chen, Wei-Jou (Edward) Duh, Hen-Hsen Huang, Makoto P. Kato, Josiane Mothe, and Barbara Poblete (Eds.). ACM, 1869–1873. doi:10.1145/3539618.3591960

[3] Peter Bailey, Nick Craswell, Ian Soboroff, Paul Thomas, Arjen P. de Vries, and Emine Yilmaz. 2008. Relevance assessment: are judges exchangeable and does it matter. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Singapore, Singapore) *(SIGIR '08)*. Association for Computing Machinery, New York, NY, USA, 667–674. doi:10.1145/1390334.1390447

[4] Krisztian Balog, Donald Metzler, and Zhen Qin. 2025. Rankers, Judges, and Assistants: Towards Understanding the Interplay of LLMs in Information Retrieval Evaluation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Padua, Italy) *(SIGIR '25)*. Association for Computing Machinery, New York, NY, USA, 3865–3875. doi:10.1145/3726302.3730348

[5] Charles Clarke and Laura Dietz. 2025. LLM-based Relevance Assessment Still Can't Replace HumanRelevance Assessment. In *Proceedings of the Eleventh International Workshop on Evaluating Information Access, EVIA 2025, a Satellite Workshop of the NTCIR-18 Conference, Tokyo, Japan, June 10, 2025*. National Institute of Informatics (NII). doi:10.20736/0002002105

[6] Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement* 20, 1 (1960), 37–46.

[7] Gordon V. Cormack, Christopher R. Palmer, and Charles L. A. Clarke. 1998. Efficient construction of large test collections. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Melbourne, Australia) *(SIGIR '98)*. Association for Computing Machinery, New York, NY, USA, 282–289. doi:10.1145/290941.291009

[8] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2021. Overview of the TREC 2021 Deep Learning Track. In *Proceedings of the Thirtieth Text REtrieval Conference, TREC 2021, online, November 15-19, 2021 (NIST Special Publication, Vol. 500-335)*, Ian Soboroff and Angela Ellis (Eds.). National Institute of Standards and Technology (NIST). https://trec.nist.gov/pubs/trec30/papers/Overview-DL.pdf

[9] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2022. Overview of the TREC 2022 Deep Learning Track. In *Proceedings of the Thirty-First Text REtrieval Conference, TREC 2022, online, November 15-19, 2022 (NIST Special Publication, Vol. 500-338)*, Ian Soboroff and Angela Ellis (Eds.). National Institute of Standards and Technology (NIST). https://trec.nist.gov/pubs/trec31/papers/Overview_deep.pdf

[10] Tadele T. Damessie, Thao P. Nghiem, Falk Scholer, and J. Shane Culpepper. 2017. Gauging the Quality of Relevance Assessments Using Inter-Rater Agreement. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 1089–1092. doi:10.1145/3077136.3080729

[11] Laura Dietz, Oleg Zendel, Peter Bailey, Charles L. A. Clarke, Ellese Cotterill, Jeff Dalton, Faegheh Hasibi, Mark Sanderson, and Nick Craswell. 2025. Principles and Guidelines for the Use of LLM Judges. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)* (Padua, Italy) *(ICTIR '25)*. Association for Computing Machinery, New York, NY, USA, 218–229. doi:10.1145/3731120.3744588

[12] 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 2023, Taipei, Taiwan, 23 July 2023*, Masaharu Yoshioka, Julia Kiseleva, and Mohammad Aliannejadi (Eds.). ACM, 39–50. doi:10.1145/3578337.3605136

[13] Nicola Ferro and Mark Sanderson. 2022. How Do You Test a Test? A Multifaceted Examination of Significance Tests. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining* (Virtual Event, AZ, USA) *(WSDM '22)*. Association for Computing Machinery, New York, NY, USA, 280–288. doi:10.1145/3488560.3498406

[14] Martin Franz and Salim Roukos. 1998. Trec-6 ad-hoc retrieval. *NIST SPECIAL PUBLICATION SP* (1998), 511–516.

[15] Lei Han, Eddy Maddalena, Alessandro Checco, Cristina Sarasua, Ujwal Gadiraju, Kevin Roitero, and Gianluca Demartini. 2020. Crowd Worker Strategies in Relevance Judgment Tasks. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 241–249.

[16] William Hersh, Chris Buckley, T. J. Leone, and David Hickam. 1994. OHSUMED: An Interactive Retrieval Evaluation and New Large Test Collection for Research. In *SIGIR '94*, Bruce W. Croft and C. J. van Rijsbergen (Eds.). Springer London, London, 192–201.

[17] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling Laws for Neural Language Models. *CoRR* abs/2001.08361 (2020). arXiv:2001.08361 https://arxiv.org/abs/2001.08361

[18] Maurice G. Kendall. 1938. A New Measure of Rank Correlation. *Biometrika* 30, 1-2 (06 1938), 81–93. doi:10.1093/biomet/30.1-2.81

[19] 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* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 39–48. doi:10.1145/3397271.3401075

[20] Kenneth A. Kinney, Scott B. Huffman, and Juting Zhai. 2008. How evaluator domain expertise affects search result relevance judgments. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*. 591–598.

[21] Klaus Krippendorff. 2011. Computing Krippendorff's alpha-reliability. (2011).

[22] Henry Kučera, Winthrop Francis, William Freeman Twaddell, Mary Lois Marckworth, Laura M Bell, and John Bissell Carroll. 1967. Computational analysis of present-day American English. *International Journal of American Linguistics* (1967).

[23] Xiangsheng Li, Yiqun Liu, Jiaxin Mao, Zexue He, Min Zhang, and Shaoping Ma. 2018. Understanding Reading Attention Distribution during Relevance Judgement. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 733–742.

[24] 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 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021)*. 2356–2362.

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

[26] Craig Macdonald and Nicola Tonellotto. 2020. Declarative Experimentation inInformation Retrieval using PyTerrier. In *Proceedings of ICTIR 2020*.

[27] Iain Mackie, Shubham Chatterjee, and Jeffrey Dalton. 2023. Generative Relevance Feedback with Large Language Models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, Hsin-Hsi Chen, Wei-Jou (Edward) Duh, Hen-Hsen Huang, Makoto P. Kato, Josiane Mothe, and Barbara Poblete (Eds.). ACM, 2026–2031. doi:10.1145/3539618.3591992

[28] Alistair Moffat, Falk Scholer, and Paul Thomas. 2012. Models and metrics: IR evaluation as a user process. In *Proceedings of the Seventeenth Australasian Document Computing Symposium* (Dunedin, New Zealand) *(ADCS '12)*. Association for Computing Machinery, New York, NY, USA, 47–54. doi:10.1145/2407085.2407092

[29] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016 (CEUR Workshop Proceedings, Vol. 1773)*, Tarek Richard Besold, Antoine Bordes, Artur S. d'Avila Garcez, and Greg Wayne (Eds.). CEUR-WS.org. https://ceur-ws.org/Vol-1773/CoCoNIPS_2016_paper9.pdf

[30] Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 708–718. doi:10.18653/v1/2020.findings-emnlp.63

[31] Rodrigo Frassetto Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *CoRR* abs/1901.04085 (2019). arXiv:1901.04085 http://arxiv.org/abs/1901.04085

[32] Rodrigo Frassetto Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document Expansion by Query Prediction. *CoRR* abs/1904.08375 (2019). arXiv:1904.08375 http://arxiv.org/abs/1904.08375

[33] David Otero, Javier Parapar, and Álvaro Barreiro. 2025. Limitations of Automatic Relevance Assessments with Large Language Models for Fair and Reliable Retrieval Evaluation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2025, Padua, Italy, July 13-18, 2025*, Nicola Ferro, Maria Maistro, Gabriella Pasi, Omar Alonso, Andrew Trotman, and Suzan Verberne (Eds.). ACM, 2545–2549. doi:10.1145/3726302.3730221

[34] Hossein A. Rahmani, Nick Craswell, Emine Yilmaz, Bhaskar Mitra, and Daniel Campos. 2024. Synthetic Test Collections for Retrieval Evaluation. 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*, Grace Hui Yang, Hongning Wang, Sam Han, Claudia Hauff, Guido Zuccon, and Yi Zhang (Eds.). ACM, 2647–2651. doi:10.1145/3626772.3657942

[35] Lida Rashidi, Justin Zobel, and Alistair Moffat. 2024. The Impact of Judgment Variability on the Consistency of Offline Effectiveness Measures. *ACM Trans. Inf. Syst.* 42, 1 (2024), 19:1–19:31. doi:10.1145/3596511

[36] Kevin Roitero, David La Barbera, Michael Soprano, Gianluca Demartini, Stefano Mizzaro, and Tetsuya Sakai. 2024. How Many Crowd Workers Do I Need? On Statistical Power when Crowdsourcing Relevance Judgments. *ACM Trans. Inf. Syst.* 42, 1 (2024), 21:1–21:26. doi:10.1145/3597201

[37] Tetsuya Sakai, Sijie Tao, Nuo Chen, Yujing Li, Maria Maistro, Zhumin Chu, and Nicola Ferro. 2024. On the Ordering of Pooled Web Pages, Gold Assessments, and Bronze Assessments. *ACM Trans. Inf. Syst.* 42, 1 (2024), 23:1–23:31. doi:10.1145/3600227

[38] Mark Sanderson, Falk Scholer, and Andrew Turpin. 2010. Relatively relevant: Assessor shift in document judgements. In *Australasian Document Computing Symposium*. https://api.semanticscholar.org/CorpusID:14426189

[39] Falk Scholer, Andrew Turpin, and Mark Sanderson. 2011. Quantifying test collection quality based on the consistency of relevance judgements. In *Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, Beijing, China, July 25-29,*

2011, Wei-Ying Ma, Jian-Yun Nie, Ricardo Baeza-Yates, Tat-Seng Chua, and W. Bruce Croft (Eds.). ACM, 1063–1072. doi:10.1145/2009916.2010057

[40] Ian Soboroff. 2025. Don't Use LLMs to Make Relevance Judgments. *Information Retrieval Research* 1, 1 (Mar. 2025), 29–46. doi:10.54195/irrj.19625

[41] Paul Thomas, Gabriella Kazai, Ryen W White, and Nick Craswell. 2022. The crowd is made of people: Observations from large-scale crowd labelling. In *Proceedings of the ACM SIGIR Conference on Human Information Interaction and Retrieval*.

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

[43] Shivani Upadhyay, Ehsan Kamalloo, and Jimmy Lin. 2024. LLMs Can Patch Up Missing Relevance Judgments in Evaluation. arXiv:2405.04727 [cs.IR] https://arxiv.org/abs/2405.04727

[44] 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] https://arxiv.org/abs/2406.06519

[45] Ellen Voorhees. 2005. Overview of the TREC 2004 Robust Retrieval Track. doi:10.6028/NIST.SP.500-261

[46] Ellen M. Voorhees. 1998. Variations in relevance judgments and the measurement of retrieval effectiveness. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Melbourne, Australia) *(SIGIR '98)*. Association for Computing Machinery, New York, NY, USA, 315–323. doi:10.1145/290941.291017

[47] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. https://openreview.net/forum?id=zeFrfgyZln

[48] Justin Zobel and Lida Rashidi. 2020. Corpus Bootstrapping for Assessment of the Properties of Effectiveness Measures. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux (Eds.). ACM, 1933–1952. doi:10.1145/3340531.3411998

A Prompts

Included below are the Rationale and Utility prompts used in these experiments. Text in **bold** highlights the main differences compared to the basic prompt.

Rationale Prompt

```
You are an expert judge of content. Using your internal knowledge and simple commonsense reasoning, try to verify if the passage is relevant to the query. Here, "0" represents that the passage has nothing to do with the query, "1" represents that the passage seems related to the query but does not answer it, "2" represents that the passage has some answer for the query, but the answer may be a bit unclear, or hidden amongst extraneous information and "3" represents that the passage is dedicated to the query and contains the exact answer.

Provide an explanation for the relevance and give your answer from one of the categories 0, 1, 2 or 3 only. One of the categorical values is compulsory in the answer.

Instructions: Think about the question. After explaining your reasoning, provide your answer in terms of 0, 1, 2 or 3 categories. Only provide the relevance category on the last line without any further details.
Example: Relevance Category: score.
###
Query: {query}
Passage: {passage}
Explanation:
```

Utility Prompt

```
Given a query and a passage, you must provide a score on an integer scale of 0 to 3 with the following meanings:
```

3 for perfectly relevant: The passage is dedicated to the query and contains the exact answer.
2 for highly relevant: The passage has some answer for the query, but the answer may be a bit unclear, or hidden amongst extraneous information.
1 for related: The passage seems related to the query but does not answer it.
0 for irrelevant: The passage has nothing to do with the query

Assume that you are writing a report on the subject of the topic. If you would use any of the information contained in the web page in such a report, mark it 1. If the web page is primarily about the topic, or contains vital information about the topic, use higher scores as described in the scale above. Otherwise, mark it 0.

Query
A person has typed "{query}" into a search engine.

Result
Consider the following passage:
{passage}

Instructions
Split this problem into steps:
Consider the underlying intent of the search.
Measure how well the content matches a likely intent of the query (M).
Measure how trustworthy the web page is (T).
Consider the aspects above and the relative importance of each, and decide on a final score (O).
Produce a JSON array of scores without providing any reasoning. Do not add any text before or after the JSON array. Example: {"M": score, "T": score, "O": score}
Results {

B Supplementary Results

Table 12. Agreement, accuracy and percentage of missing (unparsable) labels. Reported are κ for binary labels, α for graded relevance, and MAE for both binary and graded labels, as well as precision (Prec) for non-relevant (0) and relevant (1) labels, and the probability (P) of labelling a passage as relevant. Results are shown given all LLM-prompt combinations.

LLM	Prompt	κ	α	MAE (Binary)	MAE (Graded)	Accuracy	Prec(Label=0)	Prec(Label=1)	P(Label=1)	Missing (%)
Claude-3 Haiku	Basic	0.06	0.07	0.47	1.03	0.53	0.70	0.36	0.51	0.43
Claude-3 Haiku	Rationale	0.23	0.15	0.45	1.18	0.55	0.96	0.42	0.77	0.17
Claude-3 Haiku	Utility	0.16	0.00	0.52	1.28	0.48	0.97	0.39	0.84	0.21
Claude-3 Opus	Basic	0.37	0.41	0.34	0.82	0.66	0.92	0.49	0.61	0.00
Claude-3 Opus	Rationale	0.49	0.48	0.25	0.77	0.75	0.91	0.58	0.50	0.00
Claude-3 Opus	Utility	0.28	0.24	0.41	1.05	0.59	0.94	0.44	0.71	0.00
Command-R	Basic	0.09	0.00	0.58	1.16	0.42	0.97	0.36	0.91	0.00
Command-R	Rationale	0.14	-0.00	0.53	1.26	0.47	0.96	0.38	0.85	0.00
Command-R	Utility	0.15	-0.02	0.52	1.30	0.48	0.96	0.39	0.84	0.00
Command-R+	Basic	0.16	0.10	0.51	1.24	0.49	0.98	0.39	0.84	0.00
Command-R+	Rationale	0.29	0.25	0.40	1.08	0.60	0.93	0.45	0.70	1.89
Command-R+	Utility	0.22	0.24	0.47	1.00	0.53	0.97	0.41	0.78	0.00
LLaMA3 8B	Basic	0.27	0.22	0.41	0.86	0.59	0.92	0.44	0.70	0.09
LLaMA3 8B	Rationale	0.35	0.32	0.33	0.94	0.67	0.87	0.50	0.55	1.61
LLaMA3 8B	Utility	0.17	0.06	0.51	0.96	0.49	0.95	0.39	0.82	0.09
LLaMA3 70B	Basic	0.37	0.45	0.34	0.81	0.66	0.94	0.49	0.63	0.12
LLaMA3 70B	Rationale	0.41	0.45	0.31	0.81	0.69	0.94	0.52	0.59	0.09
LLaMA3 70B	Utility	0.33	0.36	0.37	0.95	0.63	0.94	0.47	0.67	0.09
GPT-3.5-turbo	Basic	0.26	0.21	0.42	1.07	0.58	0.93	0.44	0.71	0.02
GPT-3.5-turbo	Rationale	0.36	0.33	0.34	1.00	0.66	0.91	0.49	0.59	0.02
GPT-3.5-turbo	Utility	0.23	0.18	0.45	0.91	0.55	0.93	0.42	0.74	0.02
GPT-4	Basic	0.47	0.50	0.27	0.78	0.73	0.92	0.56	0.53	0.09
GPT-4	Rationale	0.49	0.57	0.22	0.64	0.78	0.82	0.68	0.31	0.14
GPT-4	Utility	0.42	0.44	0.30	0.86	0.70	0.93	0.53	0.57	0.09
GPT-4o	Basic	0.52	0.63	0.21	0.61	0.79	0.84	0.69	0.32	0.00
GPT-4o	Rationale	0.54	0.62	0.21	0.64	0.79	0.87	0.65	0.38	0.02
GPT-4o	Utility	0.52	0.62	0.22	0.61	0.78	0.88	0.63	0.41	0.95

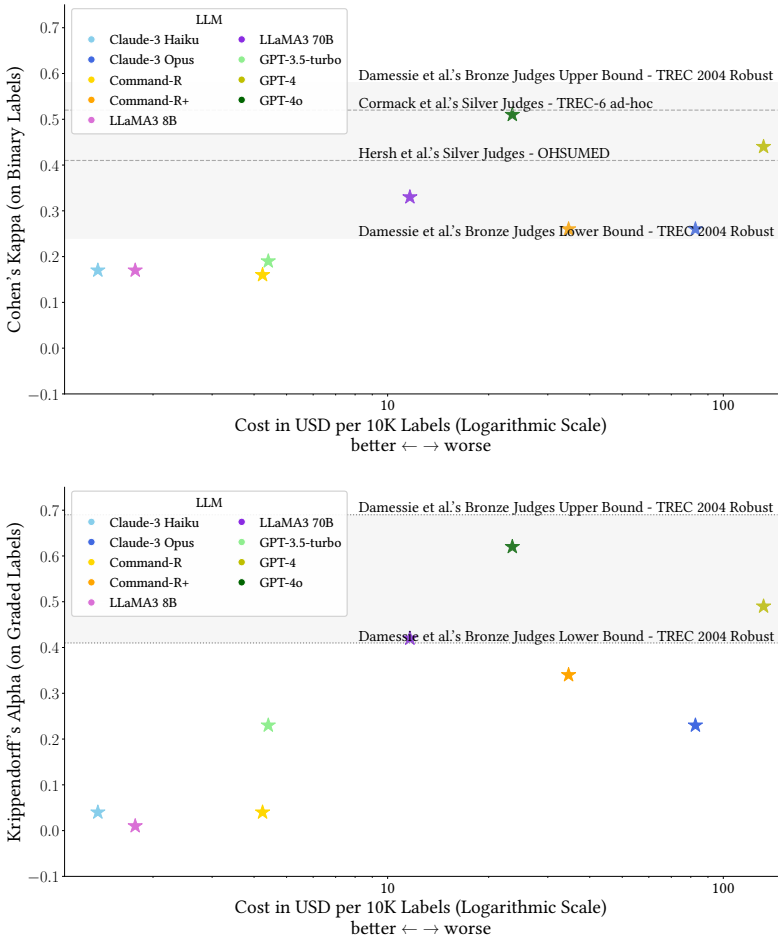


Fig. 16. Agreement between NIST relevance judgments and LLM relevance labels, measured over the full set of passages pooled from TREC-submitted runs. Agreement is shown using Cohen's κ on a binary scale (top) and Krippendorff's α on a 4-point ordinal scale (bottom), against cost. Colours represent LLM providers, with shades from lighter to darker indicating less to more capable models. Cost is calculated per 10K labels based on the average cost per label using the number of input and output tokens.