# Reasoning with Large Language Models for Relevance Judgements

Louis Geiger
louis.geiger@email.uni-freiburg.de
University of Freiburg
Freiburg, Germany

Danula Hettiachchi
danula.hettiachchi@rmit.edu.au
RMIT University
Melbourne, Australia

Falk Scholer
falk.scholer@rmit.edu.au
RMIT University
Melbourne, Australia

Johanne R. Trippas
j.trippas@rmit.edu.au
RMIT University
Melbourne, Australia

## Abstract

Relevance judgements are central to information retrieval (IR) evaluation, and there is much interest in using large language models (LLMs) for these judgements. This work presents a novel framework that operationalizes explicit reasoning as a control mechanism for LLM-based relevance judgements and systematically quantifies its effects across evaluation settings. We compare judgements with and without reasoning to assess the impact on accuracy and reliability across multiple datasets and models (e.g., gpt-oss, Gemini 2.5). Our results show that using reasoning often leads to lower-quality judgements and a higher rate of formatting failures, while incurring substantially higher token costs. These findings raise important questions about when and how reasoning should be incorporated into evaluation pipelines, particularly for passage-level relevance assessment, suggesting that its benefits depend on task context and complexity rather than being universally beneficial. This work also contributes a novel dataset of LLM reasoning outputs, augmenting two established test collections to support future research.

## CCS Concepts

• **Information systems → Users and interactive retrieval**.

## Keywords

Relevance judgements; Information Retrieval

**ACM Reference Format:**
Louis Geiger, Danula Hettiachchi, Falk Scholer, and Johanne R. Trippas. 2026. Reasoning with Large Language Models for Relevance Judgements. In *Proceedings of the 2026 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR) (ICTIR '26), July 25, 2026, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3805713.3820403

## 1 Introduction

Relevance judgements are a key component for information retrieval (IR) evaluation, providing the ground truth against which retrieval systems are trained, tuned, and assessed. Traditionally, such judgements have been created by human assessors following carefully defined guidelines to ensure consistency and reliability. However, with the increasing scale and cost of modern IR evaluation, recent years have seen a growing interest in using large language models (LLMs) as automated or semi-automated assessors of document relevance [1, 2, 17].

A common assumption in recent research is that prompting LLMs to provide explicit reasoning, such as asking them to explain or justify their decisions, will produce more accurate and reliable outputs [7, 19]. The motivation is that explicit explanations encourage transparency, consistency, and self-correction, which should improve performance compared to judgements made without reasoning. However, there is little empirical evidence on whether explicit reasoning can enhance LLM-based relevance assessment or introduce new forms of bias or error.

In this paper, we systematically address this knowledge gap by comparing relevance judgements produced with various reasoning levels across two representative state-of-the-art LLMs, *gpt-oss-120b* and *Gemini 2.5 Flash*. We conduct controlled experiments and quantitatively evaluate results against established human-relevance labels across two datasets, TREC DL 23 [6] and the more challenging DL-HARD [10], to determine whether reasoning improves the quality of LLM-generated judgements. Our findings indicate that adding reasoning does not consistently improve accuracy and, in some cases, even reduces alignment with human assessments, while substantially increasing token cost and failure rates. These results suggest that reasoning can alter model behaviour in unexpected ways, highlighting the importance of carefully designed tasks and evaluation methods in LLM-assisted IR research.

## 2 Related Work

Test collections are an important tool in IR for evaluating search systems. Traditionally, these collections include relevance judgements made by human annotators, often referred to as *gold labels*. However, judging the many documents required for these datasets can be time-consuming and expensive [9, 15, 16]. To address these limitations, the use of LLMs to automate or augment relevance judgements has become widespread, making this an active research area [2, 4, 8]. Various studies explored aspects such as the use of multi-model pipelines [14] and hybrid systems that combine human and machine assessments [16]. Alaofi et al. [1] conducted experiments with various prompts and models, showing that while LLMs



This work is licensed under a Creative Commons Attribution 4.0 International License.
*ICTIR '26, Melbourne, VIC, Australia*
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2600-2/2026/07
https://doi.org/10.1145/3805713.3820403

**Table 1: Distribution of relevance labels in the TREC DL-HARD and TREC DL 2023 datasets (without duplicates) [18].**

| Dataset | Non-relevant (0) | Partially relevant (1) | Relevant (2) | Highly relevant (3) | Total |
|---|---|---|---|---|---|
| DL23 | 11,618 | 3,774 | 1,942 | 1,544 | 18,878 |
| DL-HARD | 2,462 | 810 | 634 | 350 | 4,256 |

can perform well, they tend to be more lenient than human judges in general and can easily be tricked into changing their ratings.

Adding reasoning or "thinking" capabilities to LLMs is a recent development across providers. In contrast to earlier models, which used prompting strategies designed to evoke chain-of-thought reasoning [19], this new set of models separates reasoning from the actual response to the user's prompt. OpenAI's o1 model was the first to explicitly include such reasoning capabilities, with similar functionality now being incorporated into other models. The LLMs' reasoning aspect has been studied broadly [11]; however, its application in IR evaluation and relevance judgement remains an open question. Chen et al. [5] show that by monitoring the chain-of-thought reasoning, unwanted behaviours which would otherwise go unnoticed can be detected, albeit not reliably. Moreover, as reasoning increases both token usage and computation time, its potential benefits must be weighed against these drawbacks.

## 3 Methodology

We examine whether explicit reasoning affects the quality and reliability of LLM-based relevance judgements. By using models that expose their reasoning chains (i.e., *gpt-oss-120b* and *Gemini 2.5 Flash*) and allow us to control the level of reasoning, we can compare how reasoning steps contribute to the final relevance score. We evaluate each model across varying reasoning budgets to assess how the permitted reasoning budget influences retrieval effectiveness and explanation quality. This approach enables an assessment of the depth and transparency of reasoning in automated evaluation tasks.

### 3.1 Datasets and Models

**Datasets.** We use two existing datasets for our evaluation: The TREC Deep Learning 2023 (DL23) track dataset [6], which includes 22,327 human-annotated query-passage pairs, or *qrels*, and the Deep Learning Hard (DL-HARD) dataset [10], containing 4,256 qrels. In both datasets, each judgement is categorised into one of four relevance levels: non-relevant (0), partially relevant (1), relevant (2), and highly relevant (3). The label distributions of both datasets are shown in Table 1. For DL23, we use a cleaned version without duplicates to ensure data integrity, as the original dataset contains near-duplicates (multiple passage IDs with the same text) [13]. The DL23 dataset contains 82 unique queries across various topics, whereas DL-HARD contains 50 queries chosen specifically for being more challenging than traditional test collections.

**Models.** We evaluate two LLMs for which we adjust parameters: *gpt-oss-120b*, an open-weight model, and *Gemini 2.5 Flash*. We use OpenRouter and DeepInfra as hosts and send requests to the models using their respective APIs. Both models are prompted using the standardised "UMbrela" prompt [18] based on the DNA prompt [17] designed to elicit consistent reasoning behaviour across (the prompt is included in Appendix A). We use this prompt because it has been shown to be a robust and effective choice across different LLMs [3], while also closely incorporating the instructions given to human judges. For each model, we test multiple reasoning budgets that constrain the length or complexity of allowed reasoning traces to study how reasoning depth influences performance and output quality on the data.

A key distinction between the two LLMs is the level of access to their internal reasoning processes. The open-source *gpt-oss-120b* model provides access to intermediate outputs, which support examination of its reasoning structure, intermediate steps, and potential sources of error. In contrast, commercial models, such as *Gemini 2.5 Flash* or *GPT-5*, typically return only a summarised explanation or conceal the internal rationale. We therefore evaluate differences in retrieval performance, transparency, and interpretability between open and proprietary systems.

### 3.2 Controlling the Reasoning Process

Reasoning control varies across models and is managed through model-specific parameters. For OpenAI models, reasoning depth is configured via the `reasoning_budget` parameter, which accepts four discrete settings: minimal, low, medium, and high (minimal is not available for some models, including *gpt-oss-120b*). If no setting is specified, the parameter defaults to medium. For Google's Gemini 2.5 models, a similar mechanism is implemented via the `thinking_budget` parameter, which can take continuous values ranging from -1 (dynamic mode) to 0 (disabled reasoning) and up to a maximum of 24,576 tokens. Dynamic mode is the default, and, with our model setup, it averages 1,181 reasoning tokens per request on DL23. We did not observe a meaningful difference between the dynamic mode and a fixed budget of the same size, and in both cases, the budget primarily serves as an upper bound on the length of the reasoning trace; the actual reasoning content remains highly variable, depending on query complexity and model behaviour.

We conducted our analysis at two reasoning levels for each model: low and high for *gpt-oss-120b*, and token budgets 0 and 500 for *Gemini 2.5 Flash*. This setup enables a controlled comparison of performance and interpretability across varying reasoning capacities, while keeping other evaluation parameters constant. Following the configuration proposed by Thomas et al. [17], we set the temperature to 0, frequency penalty to 0.5, presence penalty to 0, and top_p to 1.

### 3.3 Evaluating Judgments and Reasoning

To better understand the reasoning process behind relevance judgments, we conduct evaluations on four sets of generated relevance labels, corresponding to two models (*gpt-oss-120b* and *Gemini 2.5 Flash*) at two reasoning levels each (*high/low* and *0/500*). We then compare the labels produced by the models with those assigned by human annotators to assess accuracy and reliability. We record the number of tokens of each response to calculate the average token usage per judgement. This number includes both the reasoning and final response tokens, as the API does not allow access to them separately.

To further analyse the impact of reasoning levels within each model, we also split the relevance judgements for each dataset into four disjoint subsets: **Set 1** contains all judgements where only the high reasoning agreed with the human judge; **Set 2** contains all judgements where only the low reasoning agreed with the human;

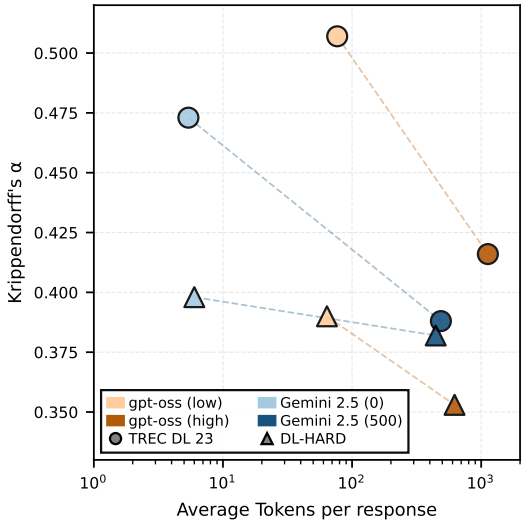



**Figure 1: Krippendorff's $\alpha$ values plotted against average number of output tokens per response for TREC DL 23 and DL-HARD and both LLMs at different reasoning budgets**

**Table 2: Results for passage-query pairs in TREC DL 23 and DL-HARD by different LLMs and reasoning amounts (Value in parentheses denotes `reasoning_budget`/`thinking_budget`)**

| Model | Binary $\kappa$ | 4-Stage $\kappa$ | Krippendorff's $\alpha$ | Invalid Results | Avg. Tokens |
|---|---|---|---|---|---|
| *TREC DL 23 (Near-duplicates removed)* | | | | | |
| gpt-oss (low) | 0.402 | 0.285 | 0.507 | 52 | 76.7 |
| gpt-oss (high) | 0.407 | 0.218 | 0.416 | 733 | 1126.5 |
| Gemini 2.5 (0) | 0.414 | 0.263 | 0.473 | 5 | 5.4 |
| Gemini 2.5 (500) | 0.418 | 0.196 | 0.388 | 203 | 487.5 |
| *DL-HARD* | | | | | |
| gpt-oss (low) | 0.282 | 0.239 | 0.390 | 0 | 63.9 |
| gpt-oss (high) | 0.328 | 0.215 | 0.353 | 1 | 622.01 |
| Gemini 2.5 (0) | 0.311 | 0.249 | 0.398 | 1 | 6.0 |
| Gemini 2.5 (500) | 0.307 | 0.224 | 0.382 | 11 | 444.37 |

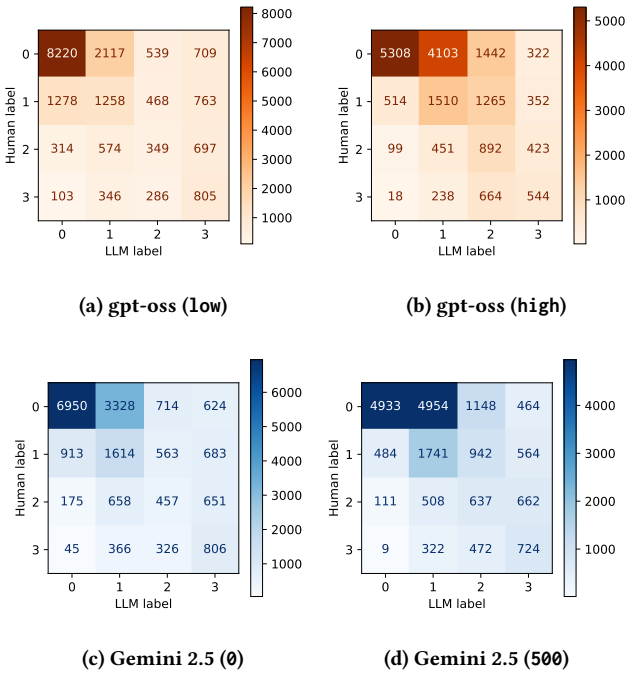
**Set 3** contains the judgements where both reasoning levels agreed with the human (and each other), and **Set 4** contains those where neither model chose the same label as the human. Formally, if $A$ is the set of "correct" judgements at the high reasoning level and $B$ the respective set at the low reasoning level, the categories represent $A \setminus B$, $B \setminus A$, $A \cap B$ and the complement of the union $(A \cup B)'$ respectively.

This separation allows us to compare characteristics such as query and passage length and word frequencies between low- and high-reasoning conditions. We can also qualitatively analyse each subset to identify common patterns and failure points in the reasoning, as we do in Sections 4.2 and 4.3.

# 4 Results

## 4.1 Overall LLM Performance

Table 2 summarises the results of the relevance assessment runs for all evaluated models. To facilitate analysis, we report results at the **binary** relevance judgements, where labels 0 and 1 are grouped as *non-relevant* and labels 2 and 3 as *relevant*, in line with the TREC




**(a) gpt-oss (`low`)** **(b) gpt-oss (`high`)**

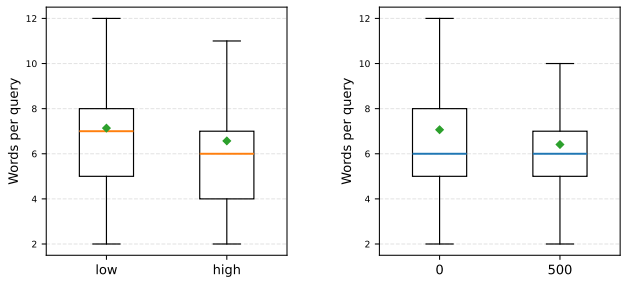
**(c) Gemini 2.5 (`0`)** **(d) Gemini 2.5 (`500`)**

**Figure 2: Comparison of human labels and those generated by two different LLMs with varying reasoning level for the TREC DL 23 dataset without near-duplicates**

DL 23 overview paper [6], as well as graded relevance on the **four-level** relevance scale (**0–3**). The binary mapping enables a direct comparison of agreement patterns, simplifying the interpretation of overall reliability. To quantify consistency between human and model-generated labels, we compute Cohen's $\kappa$ for both scales and also report Krippendorff's $\alpha$ as an additional measure of inter-rater agreement.

Both models produced agreement patterns consistent with previous work [1]. While binary $\kappa$ values slightly increased with higher reasoning, four-level relevance $\kappa$ and $\alpha$ values decreased, the latter of which is visualized in Figure 1. The decrease in four-level reliability appears to occur because the high-reasoning LLMs assign higher relevance scores more often, especially when the low-reasoning model assigned the score **0**. As shown in Figure 2, for DL23, *gpt-oss* assigned the non-relevant label (**0**) correctly 8,220 times on low reasoning and 5,308 times on high reasoning, while also incorrectly assigning scores of **1** instead of **0** 2,117 and 4,103 times, respectively. A similar upward shift occurs for *Gemini 2.5* and for both models with DL-HARD, while the latter dataset has lower reliability scores overall due to more challenging queries.

As anticipated, increasing the reasoning level increases token usage, with the average number of tokens used for each configuration reported in Table 2. For *gpt-oss*, changing from low to high reasoning increases the average output token number by a factor of up to 15. For Gemini models, the actual number of reasoning tokens is very close to the set value of `thinking_budget`, but there is some variance, and the parameter value is not a hard cut-off.




(a) gpt-oss (b) Gemini 2.5

**Figure 3: Boxplots showing query length for low vs. high reasoning models for (a) *gpt-oss* and (b) *Gemini 2.5* in TREC DL 23. Both differences are statistically significant ($p < 0.001$).**

## 4.2 Analysing Reasoning Behaviour

Following the procedure outlined in Section 3.3, we analyse the LLM-generated reasoning to identify patterns across the four relevance categories. As shown in Figure 3, the word count of queries in DL23 correctly judged (that is, matching the human labels) only by the high-reasoning models is lower than those correctly judged only by the low-reasoning models. The average query length across the whole DL23 dataset is 6.84 words. For *gpt-oss*, queries correctly judged only by the high-reasoning model average 6.57 words, while those correctly judged only by the low-reasoning model average 7.14 words ($t(4932) = -8.83$, $p < 0.001$, $d = 0.20$). A similar pattern appears for *Gemini 2.5*, with corresponding averages of 6.41 and 7.06 words, respectively ($t(4838) = -9.96$, $p < 0.001$, $d = 0.24$).

Out of all queries used in the TREC DL 23 dataset (82 in total), some consistently performed better with high reasoning, while others usually worked better with low reasoning. For example, in the *gpt-oss* runs, the query "*when did blue sour patch come out*" was correctly judged only by the high-reasoning model 40.2% of the time it appeared, whereas the query "*what is the coastline of caribbean*" performed better with low reasoning, being correctly judged only by that model 45.3% of the time. However, we did not find any common patterns across the search topics that would explain why only one or the other reasoning amount would correctly judge them. Investigating this will require further studies and is an interesting avenue for future work.

## 4.3 When Reasoning Goes Wrong

Studying the raw reasoning output produced by the LLMs during relevance judgment, we notice several cases in which the model first assumes the correct relevance label but later changes it to an incorrect one. This effect, which we call "overthinking" in line with previous work [11], gets more noticeable with longer reasoning chains. For example, we observed a case where the high-reasoning model repeatedly calls 0 the "most appropriate score" (correctly in line with the human annotator), but ultimately changes its opinion to a score of 1.[1]

[1]Examples are included in Appendix B, and full reasoning outputs are available at https://github.com/loui013/reasoning-LLM-relevance

Furthermore, the number of invalid results, i.e., those where the model did not output a response in the desired format of `##final score: X` also increased, in the case of *gpt-oss* to almost 4% of total judgements, as seen in Table 2. Most of these invalid results stem from the model inserting additional Unicode characters or unwanted explanations in its answer, which cannot then be correctly parsed. In contrast, both low-reasoning variants adhered to the specified format, except for very few cases.

Another point of failure in reasoning is when the model incorrectly attempts to establish a connection between the query and the passage, even when there is none. This often occurred with the query "*what is considered poulty*" which includes a spelling mistake. Whereas human annotators most likely ignored this mistake and interpreted the last word as "poultry", the LLM sometimes argued that a misspelling of other words, for example "poultice", could have led to the word in the query. These far-fetched assumptions occur frequently in high-level reasoning, leading the model to frequently assign a relevance score of 1 or 2 when the correct label is actually 0.

## 5 Discussion

In this paper, we investigated how varying an LLM's reasoning effort influences its performance in relevance judgement tasks. Our results show that increasing a model's reasoning depth does not consistently improve alignment with human annotators and may even reduce it on graded relevance scales. Although binary scoring shows a very slight improvement, this comes at the cost of substantially higher token usage – up to 15 times in our results – leading to greater computational expense and longer processing times for large-scale evaluations. We observe that higher reasoning tends to perform better on shorter queries, while lower reasoning performs better on longer ones. This suggests that additional reasoning may help interpret low-context or abstract queries more effectively. However, higher reasoning also introduces new error patterns, including "overthinking" where initially correct assumptions are revised into incorrect labels, increased failure to follow formatting instructions, and a systematic upward shift in relevance scores with fewer assignments of label **0**.

Even though reasoning-enabled LLMs are supposed to provide transparency into how models form judgements, especially in tasks where outputs are minimal, our findings indicate that expanding "reasoning" can alter behaviour in unexpected and sometimes undesirable ways. For passage-level relevance assessment, simply enabling more reasoning is therefore not a reliable strategy for improving judgement quality.

**Limitations.** Our study focuses on English passage-level web collections and two contemporary models; results may differ for long documents or domain-specific tasks. The UMbrela prompt, while closely aligned with TREC guidelines, may not always be the best choice for reasoning-enabled relevance assessment. Reasoning budgets were constrained for cost reasons, particularly for *Gemini 2.5*, and longer chains might exhibit different scaling effects [12]. A further challenge is limited access to full reasoning traces, particularly for commercial models, which impedes reproducibility and the broader attribution of errors and biases.

## 6 Conclusions

Through a systematic analysis of model-internal reasoning in LLM-assisted IR evaluation, conducted on the DL23 and DL-HARD benchmarks, we identified previously unreported metric- and task-dependent trade-offs: **graded agreement decreases despite minor gains in binary accuracy, while cost and failure rates rise sharply**. These outcomes suggest that reasoning should be applied selectively rather than as a default in LLM-assisted IR evaluation.

**Practical implications.** *(i)* For short-passage judging, disabling reasoning may yield more stable and cheaper assessments; *(ii)* if reasoning is used, additional validation of output format and label distribution is essential; *(iii)* reasoning traces appear more valuable for auditing model behaviour than for directly improving scores.

In future work we intend to analyse the reasoning traces more systematically, evaluate document-level collections, and explore hybrid pipelines where reasoning is used for critique or quality control rather than as the sole basis for final labels.

## Acknowledgments

This work was supported by a fellowship of the German Academic Exchange Service (DAAD), and the Australian Research Council Centre of Excellence for Automated Decision-Making and Society (CE200100005).

## References

[1] Marwah Alaofi, Paul Thomas, Falk Scholer, and Mark Sanderson. 2024. LLMs can be Fooled into Labelling a Document as Relevant. In *Proc. ACM SIGIR-AP*.
[2] Negar Arabzadeh and Charles LA Clarke. 2025. Benchmarking LLM-based relevance judgment methods. In *Proc. ACM SIGIR*.
[3] Negar Arabzadeh and Charles LA Clarke. 2025. A human-ai comparative analysis of prompt sensitivity in llm-based relevance judgment. In *Proc. ACM SIGIR*. 2784–2788.
[4] Krisztian Balog, Don Metzler, and Zhen Qin. 2025. Rankers, judges, and assistants: Towards understanding the interplay of llms in information retrieval evaluation. In *Proc. ACM SIGIR*.
[5] Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, et al. 2025. Reasoning Models Don't Always Say What They Think. *arXiv preprint arXiv:2505.05410* (2025).
[6] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Hossein A. Rahmani, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2025. Overview of the TREC 2023 deep learning track. *arXiv preprint arXiv:2507.08890* (2025).
[7] Jie Huang and Kevin Chen-Chuan Chang. 2022. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403* (2022).
[8] David La Barbera, Riccardo Lunardi, Mengdie Zhuang, and Kevin Roitero. 2025. Impersonating the Crowd: Evaluating LLMs' Ability to Replicate Human Judgment in Misinformation Assessment. In *Proc. ACM ICTIR*.
[9] David E Losada, Javier Parapar, and Alvaro Barreiro. 2019. When to stop making relevance judgments? A study of stopping methods for building information retrieval test collections. *JASIST* 70, 1 (2019), 49–60.
[10] Iain Mackie, Jeffrey Dalton, and Andrew Yates. 2021. How deep is your learning: The DL-HARD annotated deep learning dataset. In *Proc. ACM SIGIR*. 2335–2341.
[11] Sara Vera Marjanović, Arkil Patel, Vaibhav Adlakha, Milad Aghajohari, Parishad BehnamGhader, Mehar Bhatia, Aditi Khandelwal, Austin Kraft, Benno Krojer, Xing Han Lù, et al. 2025. DeepSeek-R1 Thoughtology: Let's think about LLM Reasoning. *arXiv preprint arXiv:2504.07128* (2025).
[12] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. 2025. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 20286–20332.
[13] Mandeep Rathee, Sean MacAvaney, and Avishek Anand. 2025. Guiding retrieval using llm-based listwise rankers. In *European Conference on Information Retrieval*. Springer, 230–246.
[14] Julian A Schnabel, Johanne R Trippas, Falk Scholer, and Danula Hettiachchi. 2025. Multi-stage large language model pipelines can outperform gpt-4o in relevance assessment. In *Companion Proc. ACM WebConf*.
[15] Ian Soboroff. 2025. Don't Use LLMs to Make Relevance Judgments. *Information Retrieval Research* 1, 1 (2025), 29–46.
[16] Rikiya Takehi, Ellen M. Voorhees, Tetsuya Sakai, and Ian Soboroff. 2025. LLM-Assisted Relevance Assessments: When Should We Ask LLMs for Help?. In *Proc. ACM SIGIR*.
[17] Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2024. Large Language Models can Accurately Predict Searcher Preferences.
[18] 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 preprint arXiv:2406.06519* (2024).
[19] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.

## A Prompt

Given a query and a passage, you must provide a score on an integer scale of 0 to 3 with the following meanings:
0 = represent 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.
Important Instruction: Assign category 1 if the passage is somewhat related to the topic but not completely, category 2 if passage presents something very important related to the entire topic but also has some extra information and category 3 if the passage only and entirely refers to the topic. If none of the above satisfies give it category 0.
Query: {query}
Passage: {passage}
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 passage is (T).
Consider the aspects above and the relative importance of each, and decide on a final score (O). Final score must be an integer value only.
Do not provide any code in result. Provide each score in the format of: ##final score: score without providing any reasoning.

**Figure 4:** The prompt used for relevance judgment, as described by Uphadyay et al. [18].

## B Reasoning Examples

**Query**: how do you spell with asl
**Passage**: Submit Definition.Close. Note. The ASL fingerspelling provided here is most commonly used for proper names of people and places; it is also used in some languages for concepts for which no sign is available at that moment.
**Human relevance score**: 0
**Reasoning**: **Evaluating Relevance to Query**I'm currently focused on the specific scoring criteria for relevance. I'm taking the query "how do you spell with asl" and comparing it to the passage "Submit Definition. Close. Note. The ASL fingers". I'm trying to figure out which score, 0-3, best fits the degree of relevance, according to the specified definitions, and **I believe it's going to be a 0**.**Refining the Relevance Assessment**I've been meticulously considering the user's intent: understanding *how* to spell using ASL. The passage discusses ASL fingerspelling, but it does so in a tangential way. I'm focusing on the specific scoring definitions, which emphasizes understanding the method of spelling. Based on my analysis, **I still believe the most appropriate score is a 0,** given the lack of direct instruction.**Revisiting Scoring Criteria**I'm now deep-diving into the nuances of scoring. The passage, while related to ASL fingerspelling, fails to address the "how" aspect the user seeks. Focusing on the scoring definitions, **a score of 1 seems the most appropriate**. The passage is demonstrably connected to the query's topic, but it fails to provide the instructional content the user needs.
**LLM relevance score**: 1

**Figure 5:** Example reasoning output by *Gemini 2.5-Flash* with `thinking_budget` set to 500, where initially correct reasoning turns into an incorrect final label.

**Query**: what is considered poulty
**Passage**: Poultices. A poultice is a moist paste usually made from ground herbs or foods mixed with water, oil or another liquid. A wide variety of poultices have been used historically as home remedies for boils and other skin infections 1. These pastes are usually applied to the affected area and covered.
**Human relevance score**: 0
**Reasoning `low`**: We need to output final score. Query: "what is considered poulty". Likely typo "poultry"? But passage about poultices. Not related. So score 0.
**LLM relevance score (`low`)**: 0
**Reasoning `high`**: We need to parse the problem. The user says: "Given a query and a passage, you must provide a score on an integer scale of 0 to 3 with the following meanings: ..." The query: "what is considered poulty". Actually it's likely "what is considered poultice"? **The query says "what is considered poulty". Possibly a typo for poultice?** The user wants us to evaluate how well this passage answers that question. The passage defines poultices: A poultice is a moist paste usually made from ground herbs or foods mixed with water... used historically as home remedies... applied... Thus this directly answers what poultice is - i.e., defines it. So it's relevant and provides exact answer. So likely category 3 because it's dedicated entirely to describing poultice definition. But we need also consider underlying intent - likely user wants definition/meaning of poultice/poulties? The phrase "what is considered poulty" might refer incorrectly spelled term; maybe they want meaning/definition. The passage gives definition exactly. Thus M= high match (maybe 3). T= trustworthiness moderate-high because it's factual description from presumably reliable source (though no citation beyond footnote). But trustworthiness not needed for scoring beyond internal weighting perhaps but final O should reflect overall rating based on both M & T relative importance. Given high match & decent trustworthiness => final O=3. Thus output format required: ##final score: <score> Only integer value after colon no reasoning. Thus produce: ##final score: 3
**LLM relevance score (`high`)**: 3

**Figure 6:** A query-passage pair judged by *gpt-oss 120b* with both low and high reasoning shown.