Multi-stage Large Language Model Pipelines Can Outperform GPT-4o in Relevance Assessment

Julian A. Schnabel
julian.schnabel@hhu.de
Heinrich Heine University
Düsseldorf, Germany

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

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

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

Abstract

The effectiveness of search systems is evaluated using relevance labels that indicate the usefulness of documents for specific queries and users. While obtaining these relevance labels from real users is ideal, scaling such data collection is challenging. Consequently, third-party annotators are employed, but their inconsistent accuracy demands costly auditing, training, and monitoring. We propose an LLM-based modular classification pipeline that divides the relevance assessment task into multiple stages, each utilising different prompts and models of varying sizes and capabilities. Applied to TREC Deep Learning (TREC-DL), one of our approaches showed an 18.4% Krippendorff's α accuracy increase over OpenAI's GPT-4o mini while maintaining a cost of about 0.2 USD per million input tokens, offering a more efficient and scalable solution for relevance assessment. This approach beats the baseline performance of GPT-4o (5 USD). With a pipeline approach, even the accuracy of the GPT-4o flagship model, measured in α, could be improved by 9.7%.

CCS Concepts

• **Information systems** → **Information retrieval**.

Keywords

Relevance Judgements, LLMs, Evaluation

ACM Reference Format:
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 *Proceedings of The Web Conference (WebConf '25), April 28 – May 2, 2025, Sydney, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Accurate document relevance labels are essential for training and evaluating retrieval systems, as they determine the effectiveness of search results in meeting user needs. However, labelling documents is a time-consuming and costly task [2, 5]. Currently, trained human assessors or crowd workers are employed to evaluate query-document pairs, but these approaches are often resource-intensive and prone to biases. Recent research has proposed using LLMs for relevance assessment to reduce the dependence on time-consuming and costly human assessments while improving accuracy and alignment [4]. However, employing large complex LLMs, like the flagship GPT-4o, would still incur high costs. We propose a modular classification pipeline, which has the potential to reduce labelling costs further while achieving similar accuracies compared to costly LLMs. Our pipeline approach is divided into two main steps; the LLM performs a binary classification followed by a more detailed three-level relevance labelling. This structured approach streamlines the labelling process by combining an initial relevance filter with granular classification. We demonstrate that this pipeline delivers comparable labelling accuracy to state-of-the-art LLMs while significantly lowering costs, offering a scalable and efficient solution for high-quality data annotation.

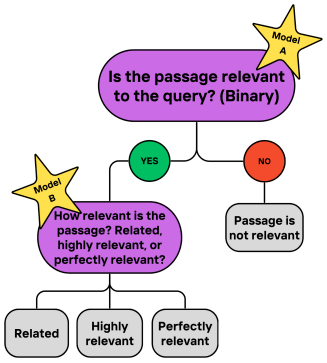

Figure 1: Visual overview of the pipeline approach where different models can judge at different stages of the relevance judgement labelling.

2 Related Work

Several works have demonstrated the feasibility of using LLMs for relevance judgement. Thomas et al. [7] explained the labelling process and prompts for using LLMs to judge relevance at Bing. Upadhyay et al. [8] provided an open-source reproduction of the Bing

WebConf '25, April 28 – May 2, 2025, Sydney, Australia

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

Relevance assessor [7] using GPT-4o. They provided the prompt and baseline accuracy score we used to validate our methods. Alaofi et al. [1] presented a comprehensive performance overview of different models, both commercial and open-source, and also tested different prompting techniques. Their findings about the impact of adversarial prompt- and keyword injection on relevance judgement suggest the value of using a complex model at the last stages of a labelling pipeline. Zendel et al. [10] conducted experiments with batch processing and labelling, suggesting that batch processing could further reduce the cost of our proposed pipeline.

3 Methodology

3.1 Dataset and Evaluation

We use the TREC 2023 Deep Learning Track (TREC-DL 23) [2], similar to Upadhyay et al. [8]. We use human relevance labels (i.e., *Qrels*) provided for queries and assess them using our proposed systems. These relevance labels, commonly called "gold labels", are the benchmark for evaluating our systems. The original judgments were made by NIST assessors, who, given a query, assigned relevance scores to passages based on the following scale [3]:

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

Although these relevance labels are considered authoritative, assessors do not have the actual information need and must infer the user's intent when assigning relevance. This reliance on subjective judgment can introduce biases or inconsistencies, which may affect the accuracy of the gold labels in reflecting user relevance. In the TREC-DL 23 dataset, the label distribution is as follows: Label 0 (13, 866), Label 1 (4, 372), Label 2 (2, 259) and Label 3 (1, 830).

Models and Metrics. We used five models: *GPT-4o* and *GPT-4o mini* via Microsoft Azure, and *Llama 3.1 70B instruct*, *Llama 3.1 305B* and *Claude 3.5 Sonnet* via OpenRouter[1]. We did not perform any fine-tuning and used all the standard parameters. Only *GPT-4o* and *GPT-4o mini* were used to analyse the proposed pipeline approaches. The remaining models were only used for cost and accuracy comparison. When referencing binary accuracy, we use Cohen's κ; when referencing accuracy on a nominal scale, we use Krippendorff's α.

3.2 Reproducing Existing Baselines

3.2.1 Zero-shot baseline from UMBRELA [8]. For our baseline, we use a zero-shot prompting technique with the *description*, *narrative*, and *aspects* (DNA) method [7] with the GPT-4o model. This DNA prompt is divided into three structured sections where *description* and *narrative* clarify the user query and the passage the LLM needs to label, helping establish context, and *aspects* provides a step-by-step guide to structure the relevance labelling task into smaller, more manageable components, facilitating a more nuanced interpretation by the LLM. To verify and reproduce the results presented by Upadhyay et al. [8], we used their exact prompt (See Figure 6) and run the same test on GPT-4o. We refer to this prompt as the "*Normal*" prompt.

3.2.2 Zero-shot baseline from UMBRELA run with GPT-4o mini. Using the same dataset and the same *Normal* prompt [8], we reproduced the results with GPT-4o mini, OpenAI's budget LLM, with a cost per million input tokens of 0.15 USD instead of 5.00 USD for GPT-4o, the current flagship model.

3.3 Relevance Judgement Pipeline Approaches

We propose three novel relevance judgement pipeline approaches, categorised by single-stage vs. multi-stage and single-model vs. multi-model, as summarised in Table 1. UMBRELA represents the single-model, single-stage method (i.e., baseline), while our proposed method uses a multi-model, single-stage approach. We also test a multi-stage (i.e., starting with a binary decision and refining to three relevance levels) approach for both single and multi-model single-model methods.

Table 1: Relevance assessment methods, categorised by single-stage vs. multi-stage and single-model vs. multi-model.

	Single-stage	**Multi-stage**
Single-model	UMBRELA (Section 3.2)	Section 3.3.2
Multi-model	Section 3.3.1	Section 3.3.3

3.3.1 Multi-model Single-stage: Same prompt - different assessors. In the Multi-model Single-stage approach, the *Normal* prompt is used for two classification stages, each stage with a different model. First, we use the *Normal* prompt to classify all Qrels. All Qrels deemed irrelevant (i.e., score 0) are excluded from further classification. Next, we use the *Normal* prompt again, but only the relevant Qrels are judged again. This adjustment allows the second assessor (i.e., the second LLM) to classify a document that passed the initial irrelevance filter as irrelevant. Alaofi et al. [1] demonstrated that LLMs can be misled into labelling documents as relevant, a vulnerability particularly evident in smaller models. Assigning the second assessor – typically the larger and more robust model – the ability to override classifications made by smaller models could potentially mitigate misclassification caused by prompt manipulation or injection. However, this hypothesis remains untested, as Alaofi et al. [1] did not conduct additional experiments to validate it.

3.3.2 Single-model Multi-stage Judging (from Binary to Three Relevance Levels). This approach uses one LLM for a two-stage relevance evaluation framework. The first stage involves a *binary classification* to determine if a passage is relevant to a given query; see Figure 8. If deemed relevant, the passage is classified into one of three relevance levels (i.e., related (1), highly relevant (2), perfectly relevant (3), see Section 3.1), enhancing the assessment's precision and granularity, see Figure 7. This structured approach, guided by the *Normal* DNA prompt, ensures that the model provides a clear initial decision followed by a detailed categorisation of relevant passages.

[1] https://openrouter.ai/

3.3.3 Multi-model Multi-stage Judging from Binary to Three Relevance Levels. Next, we use the approach explained in the previous section, but instead of using one model, different models are used for each of the stages (i.e., the binary and then three relevance level judgments). Thus, this pipeline approach variant involves dividing the judgment process by the level of importance and *assigning different models to each stage*. In practice, small and inexpensive models may perform well in making the binary decision of relevance versus irrelevance. Still, they may need assistance with the more complex task of distinguishing between documents rated as different relevance levels (i.e., highly relevant vs perfectly relevant). This approach has the potential to be cost-effective.

4 Results

Table 2 summarises the evaluation outcomes of the baselines and all proposed pipelines. We evaluate two models (GPT-4o and 4o-mini) under both homogeneous (4o–4o, mini–mini) and heterogeneous (mini–4o, 4o–mini) pairings, alongside two prompt types: Binary–Relevant and Binary–Normal.

Table 2: Accuracy for different GPT model/prompt combinations on TREC-DL23. Cost in USD per million input tokens.

Model		Prompt		Binary	4-scale		Cost
1	2	1	2	κ	κ	α	USD
4o	-	Normal	-	0.453	**0.296**	0.408	5.00
mini	-	Normal	-	0.400	0.254	0.359	**0.15**
mini	mini	Binary	Relevant	0.437	0.284	0.422	0.21
4o	4o	Binary	Relevant	0.428	0.280	0.450	6.57
mini	4o	Binary	Relevant	0.437	0.286	0.432	2.05
4o	mini	Binary	Relevant	0.428	0.279	0.443	5.05
mini	mini	Binary	Normal	0.439	0.281	0.425	0.21
4o	4o	Binary	Normal	0.429	0.280	**0.452**	6.57
mini	4o	Binary	Normal	0.450	0.295	0.446	2.05
4o	mini	Binary	Normal	0.430	0.276	0.445	5.05
mini	4o	Normal	Normal	0.400	0.260	0.367	2.87
4o	mini	Normal	Normal	**0.462**	0.294	0.411	5.05

4.1 Single-model Single-stage Judging Results

The reproduced UMBRELA baseline [8] obtained similar accuracy scores, as shown in Figure 2. The misjudgement "pattern" is also similar to UMBRELA's. GPT-4o seems to be slightly over-optimistic.

GPT-4o still performs best among all tested stand-alone models. This aligns with the findings of Alaofi et al. [1]. However, GPT-4o mini's accuracy is, regarding the cost per input tokens being only 3% of the cost of GPT-4o, satisfactory. As shown in Figure 4, the agreement between GPT4o-mini and its larger variant, GPT-4o, is high. Notably, the binary accuracy given in Table 2 increased for GPT-4o mini when using the custom binary prompt, whereas the accuracy for GPT-4o was reduced.

4.2 Multi-model Single-stage Judging Results

Using the same *Normal* prompt for both stages with different assessors (i.e., LLMs) produced the highest binary accuracy (See Table 2) but yielded below-average results for four-scale κ.

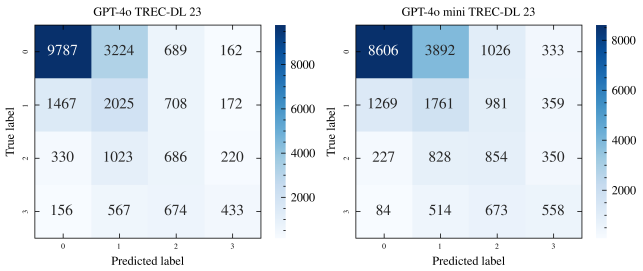

Figure 2: Reproduction of baseline (UMBRELA) with GPT-4o and GPT-4o mini.

4.3 Multi-model Multi-stage Judging Results

In the Multi-model Multi-stage approach using a modified binary prompt, the binary accuracy is identical to the binary accuracy of the respective Model 1. Four-scale Cohen's κ is higher in every multi-model approach than single-model single-stage with GPT-4o. However, the (4-scale) κ-Score for GPT-4o could not be exceeded (See Table 2). It is important to note that the GPT-4o mini/GPT-4o, binary/normal prompt combination reaches almost the same level of accuracy while significantly reducing the cost (See Figure 3).

Krippendorff's α-score weights misjudgements on the difference to the actual score. Every Model/Prompt combination outperformed the baseline single model results. The highest α-score for Multi-model Multi-stage was generated by utilizing the binary prompt with GPT-4o mini and the normal 4-scale prompt with GPT-4o as the relevance classifier. Similar accuracy was achieved by reversed model roles.

4.4 Single-model Multi-stage Judging Results

Krippendorff's α increased for every version of the two-stage judgement for both GPT-4o and GPT-4o mini. Approaches, where both models were GPT-4o, performed slightly better than the small model pipelines. However, using GPT-4o mini for both stages and the binary and normal prompt results in a 0.425 or 0.422 α when using the modified relevance prompt (Confusion matrix in Figure 5). This exceeded the GPT-4o mini accuracy and even slightly outperformed the GPT-4o stand-alone. As shown in Figure 3, these results highlight that our pipeline approach can gain accuracy while significantly reducing costs.

4.5 Cost Effectiveness

For the cost calculation, we only consider the cost per million input tokens because only one character is generated as output. Thus, the output cost is negligible but would scale similarly to the input token cost. The following formula was used for the pipeline approach cost calculation, and the comparison of cost vs accuracy is shown in Figure 3.

$$\text{Cost} = \text{cost}_{\text{M1}} + \text{cost}_{\text{M2}} \cdot (1 - \text{rate}_{\text{M1:0}})$$

5 Conclusions

Of the proposed pipelines, except the Multi-model Single-stage approach, all combinations increase Krippendorff's α compared to the baseline. The largest increase in accuracy was achieved when

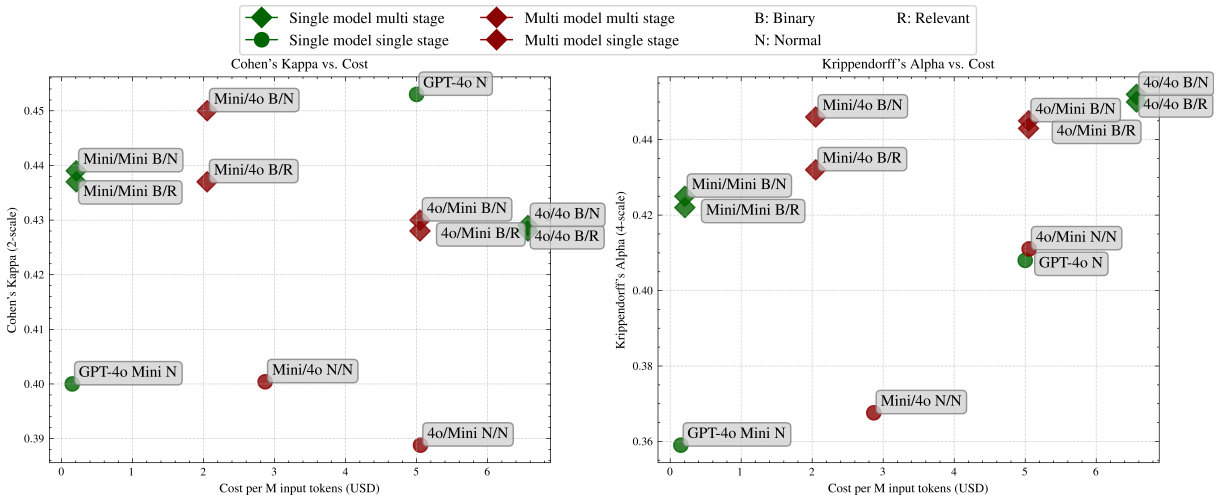

Figure 3: Cost vs Cohen's Kappa (κ) and Krippendorff's Alpha (α).

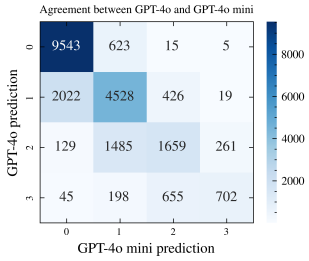

Figure 4: GPT-4o prediction vs GPT-4o mini predictions on TREC-DL 23.

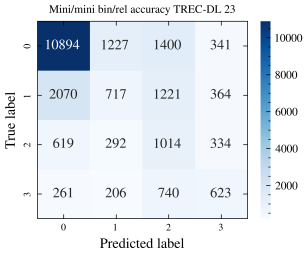

Figure 5: GPT-4o mini / GPT-4o mini Bin/R predictions on TREC-DL 23.

using GPT-4o mini. Most notably, the Single-model Multi-stage approach achieved high accuracy while maintaining an extremely low cost. This accuracy could not be achieved just using GPT-4o, the flagship model, although it is more than 20 times more expensive. Regarding cost, the affordability of GPT-4o mini leaves room for more elaborate pipelines, incorporating even more stages. Possible approaches could include a step for each relevance label or more specialised prompts after a "pre-classification".

Using a specialised binary classification prompts increased accuracy for the smaller model. Especially in the multi-stage approaches, dividing the relevance judgement task into a binary relevance decision and a relevance classification was beneficial for overall accuracy. We note limitations in our work. For example, Upadhyay et al. [8] states that near duplicates are contained in each label category in TREC-DL23. Since filtering these out is a complex process, we did not filter duplicates. In addition, to enhance the generalisability of findings, the tests could be conducted on more datasets, such as TREC-DL from different years and broader ranges of tasks. In addition, even though we reported in Section 3.1 that we used additional models to OpenAI, none of the open-source models demonstrated competitive performance compared to OpenAI's solutions. For this reason, we chose not to include their results in our analysis. Further research could improve our prompt and/or fine-tune models for an even higher irrelevance-detection rate. For example, the prompt could be optimised with techniques such as chain-of-thought [9] or narratives [6]. The TREC-DL datasets are heavily zero-weighted (ca. 75%). This fact raises the importance of "spam-filtering" in relevance judgment tasks. Given that the current state of knowledge is that larger models perform better, a small (and affordable) "spam-filter" will significantly reduce overall assessment costs.

Acknowledgments

This research is partially supported by the Australian Research Council (CE200100005) and RMIT AWS Cloud Supercomputing Hub.

References

[1] Marwah Alaofi, Paul Thomas, Falk Scholer, and Mark Sanderson. 2024. LLMs can be Fooled into Labelling a Document as Relevant. In *Proc. SIGIR-AP.*
[2] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2023. Overview of the TREC 2022 Deep Learning Track. In *Proc. TREC.* NIST, TREC.
[3] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M Voorhees. 2020. Overview of the TREC 2019 Deep Learning Track. (2020).
[4] Zheng Liu, Yujia Zhou, Yutao Zhu, Jianxun Lian, Chaozhuo Li, Zhicheng Dou, Defu Lian, and Jian-Yun Nie. 2024. Information Retrieval Meets Large Language Models. In *Proc. WebConf.*
[5] Kevin Roitero, Alessandro Checco, Stefano Mizzaro, and Gianluca Demartini. 2022. Preferences on a Budget: Prioritizing Document Pairs when Crowdsourcing Relevance Judgments. In *Proc. WebConf.*
[6] Vahid Sadiri Javadi, Johanne R Trippas, and Lucie Flek. 2024. Unveiling Information Through Narrative In Conversational Information Seeking. In *Proc. CUI.*
[7] Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2024. Large Language Models can Accurately Predict Searcher Preferences. In *Proc. SIGIR.*
[8] 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).
[9] 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.
[10] Oleg Zendel, J. Shane Culpepper, Falk Scholer, and Paul Thomas. 2024. Enhancing Human Annotation: Leveraging Large Language Models and Efficient Batch Processing. In *Proc. CHIIR.*

A Prompts

```
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 6: Baseline UMBRELA prompt as used by Upadhyay et al. [8].

```
Given a query and a passage, you must provide a score on an
integer scale of 1 to 3 with the following meanings:
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.
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 7: Modified 3-scale classification prompt (Relevant).

```
Given a query and a passage, you must provide a score on an
integer scale of 0 to 1 with the following meanings:
0 = represent that the passage has nothing to do with the query,
1 = represents that the passage has something to do with the
query.
Important Instruction: Assign category 1 if the passage is
relevant to the topic. If it is not relevant to the topic,
assign 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 (0). The final score must be an
integer value only.
Do not provide any code in the result. Provide each score in the
format of: ##final score: score without providing any reasoning.
```

Figure 8: Modified binary classification prompt (Binary).